Universidad de Valladolid

ESCUELA DE INGENIERÍAS INDUSTRIALES

UNIVERSIDAD DE VALLADOLID

ESCUELA DE INGENIERÍAS INDUSTRIALES

Grado en Ingeniería en Electrónica Industrial y Automática

Control de una caldera con procesador ARM Cortex M4

ANEJOS

Autor:

Cano Ortega, Pablo

Tutor:

Plaza Pérez, Francisco

Departamento de Tecnología Electrónica

Valladolid, 11 de febrero de 2023

ÍNDICE

Código C kit STM32F429I-DISC1.....5
Código Kotlin aplicación Android HomeT 7
Datasheets.....9

Código C kit STM32F429I-DISC1

```c
/**
  ******************************************************************************
  * File Name          : main.c
  * Description        : Main program body
  ******************************************************************************
  ** This notice applies to any and all portions of this file
  * that are not between comment pairs USER CODE BEGIN and
  * USER CODE END. Other portions of this file, whether
  * inserted by the user or by software development tools
  * are owned by their respective copyright owners.
  *
  * COPYRIGHT(c) 2022 STMicroelectronics
  *
  * Redistribution and use in source and binary forms, with or without modification,
  * are permitted provided that the following conditions are met:
  *   1. Redistributions of source code must retain the above copyright notice,
  *      this list of conditions and the following disclaimer.
  *   2. Redistributions in binary form must reproduce the above copyright notice,
  *      this list of conditions and the following disclaimer in the documentation
  *      and/or other materials provided with the distribution.
  *   3. Neither the name of STMicroelectronics nor the names of its contributors
  *      may be used to endorse or promote products derived from this software
  *      without specific prior written permission.
  *
  * THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
  * AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
  * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
  * DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE
  * FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
  * DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
  * SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER
  * CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
  * OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
  * OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
  *
  ******************************************************************************
  *
  * Universidad de Valladolid
  * Escuela de Ingenierias Industriales
  * Grado en Ingenieria en Electronica Industrial y Automatica
  *
  * Asignatura: Trabajo fin de grado
  * Año: 2022
  * Autor: Pablo Cano Ortega
  * Descripcion: Programa para el control de una caldera en una vivienda usando un procesador ARM Cortex M4.
  *
  ******************************************************************************
  */

/* Includes ------------------------------------------------------------------*/
#include "main.h"
#include "stm32f4xx_hal.h"

/* USER CODE BEGIN Includes */
#include "stm32f429i_discovery_lcd.h" // LCD
#include "stm32f429i_discovery_ts.h"  // pantalla tactil
#include <stdbool.h> // datos booleanos
#include <stdio.h> // sizeof(), strlen(), sprintf()...
#include <string.h> // funcion strstr(),
#include <stdlib.h> // funcion atof()

/* USER CODE END Includes */

/* Private variables ---------------------------------------------------------*/
// VARIABLES GENERADAS A PARTIR DE LA CONFIGURACION DE PINES Y RELOJES //
ADC_HandleTypeDef hadc1; // manejador del convertidor analogico/digital

// Variables del reloj de tiempo real
RTC_HandleTypeDef hrtc; // manejador del RTC
RTC_TimeTypeDef sTime; // tiempo: horas, minutos y segundos
RTC_DateTypeDef sDate; // fecha

UART_HandleTypeDef huart5; // manejador del puerto UART5

/* USER CODE BEGIN PV */
/* Private variables ---------------------------------------------------------*/
```

```
 77
 78  // DEFINES DE LAS POSICIONES DE SIMBOLOS Y BOTONES //
 79  // Simbolo Celsius (circulo) en la temperatura medida
 80  #define celsiusX 182
 81  #define celsiusY 76
 82  #define celsiusR 2
 83
 84  // Separacion superior (linea horizontal)
 85  #define separaSupY 110
 86
 87  // Botones (circulos)
 88  #define botonR 24 // Radio
 89  // Posicion botones, centros de los circulos
 90  #define botonIzqX 50
 91  #define botonDerX 190
 92  #define botonSupY 170
 93  #define botonInfY 270
 94
 95  // Separacion inferior (linea horizontal)
 96  #define separaInfY 210
 97
 98  // Simbolo reloj
 99  #define simbRelR 16 // Radio
100
101  // OTRAS VARIABLES //
102  static TS_StateTypeDef  TS_State; // Variable para guardar la informacion al pulsar sobre la pantalla
103
104  /* USER CODE END PV */
105
106  /* Private function prototypes -----------------------------------------------*/
107  // FUNCIONES GENERADAS A PARTIR DE LA CONFIGURACION DE PINES Y RELOJES //
108  void SystemClock_Config(void); // Relojes
109  static void MX_GPIO_Init(void); // Inicializacion de pines GPIO. Pin PG14, controla el LED L4
110  static void MX_ADC1_Init(void); // Inicializacion del convertidor analogico/digital 1
111  static void MX_RTC_Init(void); // Inicializacion del reloj de tiempo real
112  static void MX_UART5_Init(void); // Inicializacion del puerto UART5
113
114  /* USER CODE BEGIN PFP */
115  /* Private function prototypes -----------------------------------------------*/
116  // FUNCIONES DESARROLLADAS //
117  static void PantallaInicial(float, bool); // Pantalla inicial del dispositivo
118  static void PantallaRTC(int, int); // Pantalla del RTC, se configura la hora y fecha del dispositivo
119  static void PantallaManual(float); // Pantalla del modo manual
120  static void PantallaProgTemp(float, int); // Pantalla eleccion de temperaturas del modo programa
121  static void PantallaProgDias(void); // Pantalla eleccion de dias del modo programa
122  static void PantallaProgHoras(int, int, int, bool[24][7]); // Pantalla de horas del modo programa
123  static void ActualizarMatriz(int, int, int, bool[24][7]); // Funcion para actualizar la matriz de
     temperaturas, horas y dias del modo programa
124  static void ControlCaldera(float, int, int, float, float, bool[24][7]); // Funcion para controlar la
     caldera, en modo programa, a traves del rele
125  static void ESP_Init(void); // Inicializacion del modulo ESP
126  static void ResponderHTTP(bool *, float, float *, float *, float *, bool[24][7]); // Funcion que
     procesa las peticiones HTTP de la app
127  static void ActualizarMatrizHTTP(int, char[25], bool[24][7]); // Actualiza la matriz del modo
     programa segun la peticion desde la app
128  static void ProgramaBoolAChar(int, bool[24][7], char[24]); // Funcion que pasa el programa de un dia
     de Bool a Char
129
130  /* USER CODE END PFP */
131
132  /* USER CODE BEGIN 0 */
133
134  /* USER CODE END 0 */
135
136  int main(void)
137  {
138
139    /* USER CODE BEGIN 1 */
140
141    /* USER CODE END 1 */
142
143    /* MCU Configuration--------------------------------------------------------*/
144
145    /* Reset of all peripherals, Initializes the Flash interface and the Systick. */
146    HAL_Init();
147
148    /* USER CODE BEGIN Init */
```

```
149
150     /* USER CODE END Init */
151
152     /* Configure the system clock */
153     SystemClock_Config();
154
155     /* USER CODE BEGIN SysInit */
156
157     /* USER CODE END SysInit */
158
159     /* Initialize all configured peripherals */
160     MX_GPIO_Init();
161     MX_ADC1_Init();
162     MX_RTC_Init();
163     MX_UART5_Init();
164
165     /* USER CODE BEGIN 2 */
166     // VARIABLES GENERALES //
167     uint8_t estado; // Variable para guardar el estado (OK o ERROR) al inicializar la pantalla tactil
168     uint16_t x, y; // Variables para guardar la posicion donde se ha pulsado la pantalla
169     bool modo = 0; // Modo del termostato, manual = 0 y programa = 1
170     float tempSensor = 18.0; // Variable para guardar la temperatura del sensor
171     float tempManual = 18.0; // Variable para guardar la temperatura manual
172     float tempPrograma1 = 18.0; // Variable para guardar la temperatura 1 del modo programa
173     float tempPrograma2 = 18.0; // Variable para guardar la temperatura 2 del modo programa
174
175     // Matriz de temperaturas de 24 filas (horas) y 7 columnas (dias)
176     // Booleanos: un 0 indica que es T1, un 1 que es T2
177     bool matriz_temp[24][7] = {0}; // Se inicializa todo a 0
178
179     // Se prepara la pantalla LCD
180     BSP_LCD_Init(); // Inicializa el LCD de 240 x 320 pixels
181     BSP_LCD_LayerDefaultInit(1, SDRAM_DEVICE_ADDR); // Definicion de la direccion del buffer de la SDRAM
182     BSP_LCD_SelectLayer(1); // Seleccion de la capa donde esccribimos
183     BSP_LCD_DisplayOn(); // Se enciende el LCD
184     BSP_LCD_Clear(LCD_COLOR_WHITE); // Se borra y se pone el LCD en un color
185     BSP_LCD_SetBackColor(LCD_COLOR_WHITE); // Se pone el color del background
186     BSP_LCD_SetTextColor(LCD_COLOR_BLACK); // Se pone el color del texto
187
188     // Se inicializa el ESP
189     ESP_Init();
190
191     // Se fija la hora inicial del RTC
192     sTime.Hours = 12; // Hora
193     sTime.Minutes = 30; // Minutos
194     sTime.Seconds = 0; // Segundos
195     HAL_RTC_SetTime(&hrtc, &sTime, RTC_FORMAT_BIN); // Se guarda la hora
196     // Se fija la fecha inicial del RTC
197     sDate.Date = 15; // Dia del mes
198     sDate.Month = 1; // Mes: 1 = RTC_MONTH_JANUARY, 2 = RTC_MONTH_FEBRUARY, etc.
199     sDate.WeekDay = 1; // Dia de la semana: 1 = RTC_WeekDay_MONDAY, 2 = RTC_WeekDay_TUESDAY, etc.
200     sDate.Year = 22; // Año: 2000 + Year
201     HAL_RTC_SetDate(&hrtc, &sDate, RTC_FORMAT_BIN); // Se guarda la fecha
202
203     // Se muestra la pantalla inicial
204     PantallaInicial(tempSensor, modo);
205
206     // Se inicializa la pantalla tactil y la funcion devuelve si el proceso se ha relizado
    correctamente
207     estado = BSP_TS_Init(BSP_LCD_GetXSize(), BSP_LCD_GetYSize());
208
209     // Si ocurre algun error al inicar la pantalla tactil, se avisa del mismo por el LCD
210     if (estado != TS_OK)
211     {
212       // Se muestra que hay un error con la pantalla tactil
213       BSP_LCD_Clear(LCD_COLOR_WHITE);
214       BSP_LCD_SetBackColor(LCD_COLOR_WHITE);
215       BSP_LCD_SetTextColor(LCD_COLOR_RED);
216       BSP_LCD_DisplayStringAtLine(5, "ERROR DE TS");
217     }
218     /* USER CODE END 2 */
219
220     /* Infinite loop */
221     /* USER CODE BEGIN WHILE */
222     while (1)
223     {
224     /* USER CODE END WHILE */
```

```
225
226     /* USER CODE BEGIN 3 */
227
228       // Se manejan las peticiones HTTP que llegan desde la aplicacion Android
229       ResponderHTTP(&modo, tempSensor, &tempManual, &tempPrograma1, &tempPrograma2, matriz_temp);
230
231       // Se obtiene el tiempo del RTC, para que funcione correctamente hay que coger Time y Date al
    mismo tiempo
232       HAL_RTC_GetTime(&hrtc, &sTime, RTC_FORMAT_BIN);
233       HAL_RTC_GetDate(&hrtc, &sDate, RTC_FORMAT_BIN);
234
235       // Se obtiene el valor de la temperatura actual del sensor TMP36
236       if (sTime.Seconds == 2 || sTime.Seconds == 32)  // Se mide la temperatura actual cada 30 segundos
237       {
238         float suma_temp = 0.0; // Variable para la media
239         // Se toman 50 mediciones del sensor en 1 segundo y se calcula la media
240         for (int i = 0; i < 50; i++)
241         {
242           // Se realiza la conversion analogico/digital
243           // Las funciones HAL_ADC hacen uso del manejador hadc1
244           HAL_ADC_Start(&hadc1);
245           HAL_ADC_PollForConversion(&hadc1, 10); // Se realiza la conversion con un timeout de 10
    milisegundos
246           int valor_pin = HAL_ADC_GetValue(&hadc1);
247           HAL_ADC_Stop(&hadc1);
248
249           // Conversion usando 12 bits --> 2 ^ (12) = 4096 niveles en el ADC
250           float voltaje = valor_pin * (2.93 / 4096.0); // Se convierte el nivel obtenido a voltaje
251           float temp = (voltaje - 0.5) * 100.0; // Se pasa el voltaje a temperatura
252           suma_temp += temp;
253           HAL_Delay(10); // 10 milisegundos de delay para hacer 50 mediciones durante 1 segundo
254         }
255         tempSensor = suma_temp / 50.0; // Se calcula la media como la temperatura medida como el
    sensor
256         PantallaInicial(tempSensor, modo); // Se muestra la nueva temperatura en la pantalla
257       }
258
259       // Control de la caldera mediante relé cada minuto, con una histeresis de medio grado
260       if (sTime.Seconds == 40)
261       {
262         if (modo) // Modo programa
263         {
264           int diaControl = sDate.WeekDay - 1; // Se resta 1 porque el sDate.WeekDay va de 1 a 7 y las
    columnas de la matriz de 0 a 6
265           ControlCaldera(tempSensor, diaControl, sTime.Hours, tempPrograma1, tempPrograma2,
    matriz_temp); // Control de la caldera en modo programa
266         }
267         else // Modo manual
268         {
269           // Temperatura inferior a la deseada, histeresis de medio grado
270           if (tempSensor <= (tempManual - 0.5)) HAL_GPIO_WritePin(LED_GPIO_Port, LED_Pin,
    GPIO_PIN_SET); // Se enciende el LED lo que indica que se cierra el rele y se activa la caldera
271           // Temperatura superior a la deseada, histeresis de medio grado
272           else if (tempSensor >= (tempManual + 0.5)) HAL_GPIO_WritePin(LED_GPIO_Port, LED_Pin,
    GPIO_PIN_RESET); // Se apaga el LED lo que indica que se abre el rele y se desactiva la caldera
273         }
274       }
275
276       // CONTROL DE LA PANTALLA TACTIL //
277       if (estado == TS_OK) // Si la pantalla tactil se inicio correctamente
278       {
279         BSP_TS_GetState(&TS_State); // Rutina para obtener la informacion o el estado de la pantalla
    tactil
280
281         if (TS_State.TouchDetected) // Se detecta una pulsacion sobre la pantalla
282         {
283           // Se guardan las coordenadas al pulsar sobre la pantalla
284           x = TS_State.X;
285           y = TS_State.Y;
286
287           // Se chequean si las coordenadas entran dentro del Centro boton +- Radio boton (24)
288           if (x >= 96 && x <= 144 && y >= 246 && y <= 294) // Se pulsa modo programa. Centro: x = 120,
    y = 270
289           {
290             int paso_programar = 1; //  Variable para guardar el paso en el se esta cuando estamos en
    modo programacion
291             PantallaProgTemp(tempPrograma1, paso_programar);
```

```
292
293             while (1)
294             {
295               BSP_TS_GetState(&TS_State);
296
297               if (TS_State.TouchDetected)
298               {
299                 x = TS_State.X;
300                 y = TS_State.Y;
301
302                 if (x >= 26 && x <= 74 && y >= 146 && y <= 194) // Se disminuye 0.5 la temperatura (T1
    o T2)
303                 {
304                   // En funcion del paso se disminuye la temperatura de programa 1 o 2
305                   if (paso_programar == 1)
306                   {
307                     tempPrograma1 -= 0.5;
308                     PantallaProgTemp(tempPrograma1, paso_programar); // Se muestra el cambio
309                   }
310                   else if (paso_programar == 2)
311                   {
312                     tempPrograma2 -= 0.5;
313                     PantallaProgTemp(tempPrograma2, paso_programar); // Se muestra el cambio
314                   }
315                 }
316                 else if (x >= 156 && x <= 204 && y >= 146 && y <= 194) // Se aumenta 0.5 la
    temperatura (T1 o T2)
317                 {
318                   // En funcion del paso se aumenta la temperatura de programa 1 o 2
319                   if (paso_programar == 1)
320                   {
321                     tempPrograma1 += 0.5;
322                     PantallaProgTemp(tempPrograma1, paso_programar); // Se muestra el cambio
323                   }
324                   else if (paso_programar == 2)
325                   {
326                     tempPrograma2 += 0.5;
327                     PantallaProgTemp(tempPrograma2, paso_programar); // Se muestra el cambio
328                   }
329                 }
330                 else if (x >= 26 && x <= 74 && y >= 246 && y <= 294) // Se pulsa en OK
331                 {
332                   // En funcion del paso actual se pasa a configurar parametro pertinente
333                   if (paso_programar == 1) // Se pasa a configurar T2
334                   {
335                     paso_programar = 2;
336                     PantallaProgTemp(tempPrograma2, paso_programar); // Se muestra el cambio a T2
337                   }
338                   else if (paso_programar == 2) // Se pasa a configurar los programas de los dias de
    la semana
339                   {
340                     modo = 1; // Se pasa a modo programa
341                     paso_programar = 3;
342                     PantallaProgDias(); // Se muestra la pantalla de los dias
343
344                     while (1)
345                     {
346                       BSP_TS_GetState(&TS_State);
347
348                       if (TS_State.TouchDetected)
349                       {
350                         x = TS_State.X;
351                         y = TS_State.Y;
352                         int dia;
353                         // Se empieza a configurar el programa empezando de 12 de la noche (0) a 1 de
    la mañana (1)
354                         int hora_inicio = 0;
355                         int hora_fin = 1;
356
357                         if (x >= 96 && x <= 144 && y >= 246 && y <= 294) // Se pulsa en Lunes
358                         {
359                           dia = 0;
360                           PantallaProgHoras(dia, hora_inicio, hora_fin, matriz_temp); // Se muestra la
    pantalla de las horas de 0 a 1
361                           ActualizarMatriz(dia, hora_inicio, hora_fin, matriz_temp); // Se actualiza
    la matriz de temperaturas
362                         }
```

```
363                       else if (x >= 166 && x <= 214 && y >= 246 && y <= 294) // Se pulsa en Martes
364                       {
365                         dia = 1;
366                         PantallaProgHoras(dia, hora_inicio, hora_fin, matriz_temp); // Se muestra la pantalla de las horas de 0 a 1
367                         ActualizarMatriz(dia, hora_inicio, hora_fin, matriz_temp); // Se actualiza la matriz de temperaturas
368                       }
369                       else if (x >= 26 && x <= 74 && y >= 146 && y <= 194) // Se pulsa en Miercoles
370                       {
371                         dia = 2;
372                         PantallaProgHoras(dia, hora_inicio, hora_fin, matriz_temp); // Se muestra la pantalla de las horas de 0 a 1
373                         ActualizarMatriz(dia, hora_inicio, hora_fin, matriz_temp); // Se actualiza la matriz de temperaturas
374                       }
375                       else if (x >= 96 && x <= 144 && y >= 146 && y <= 194) // Se pulsa en Jueves
376                       {
377                         dia = 3;
378                         PantallaProgHoras(dia, hora_inicio, hora_fin, matriz_temp); // Se muestra la pantalla de las horas de 0 a 1
379                         ActualizarMatriz(dia, hora_inicio, hora_fin, matriz_temp); // Se actualiza la matriz de temperaturas
380                       }
381                       else if (x >= 166 && x <= 214 && y >= 146 && y <= 194) // Se pulsa en Viernes
382                       {
383                         dia = 4;
384                         PantallaProgHoras(dia, hora_inicio, hora_fin, matriz_temp); // Se muestra la pantalla de las horas de 0 a 1
385                         ActualizarMatriz(dia, hora_inicio, hora_fin, matriz_temp); // Se actualiza la matriz de temperaturas
386                       }
387                       else if (x >= 26 && x <= 74 && y >= 46 && y <= 94) // Se pulsa en Sabado
388                       {
389                         dia = 5;
390                         PantallaProgHoras(dia, hora_inicio, hora_fin, matriz_temp); // Se muestra la pantalla de las horas de 0 a 1
391                         ActualizarMatriz(dia, hora_inicio, hora_fin, matriz_temp); // Se actualiza la matriz de temperaturas
392                       }
393                       else if (x >= 96 && x <= 144 && y >= 46 && y <= 94) // Se pulsa en Domingo
394                       {
395                         dia = 6;
396                         PantallaProgHoras(dia, hora_inicio, hora_fin, matriz_temp); // Se muestra la pantalla de las horas de 0 a 1
397                         ActualizarMatriz(dia, hora_inicio, hora_fin, matriz_temp); // Se actualiza la matriz de temperaturas
398                       }
399                       else if (x >= 166 && x <= 214 && y >= 46 && y <= 94) // Se pulsa en Salir
400                       {
401                         // Se vuelve al inicio
402                         paso_programar = 0;
403                         PantallaInicial(tempSensor, modo);
404                         break;
405                       }
406                     }
407                     HAL_Delay(200);
408                   }
409                 }
410               }
411               else if (x >= 166 && x <= 214 && y >= 246 && y <= 294) // Se pulsa en BACK
412               {
413                 // En funcion del paso se vuelve al inicio o a la temperatura 1
414                 if (paso_programar == 1) // Se vuelve al inicio
415                 {
416                   PantallaInicial(tempSensor, modo);
417                   break;
418                 }
419                 else // Se vuelve a la temperatura 1
420                 {
421                   paso_programar = 1;
422                   PantallaProgTemp(tempPrograma1, paso_programar);
423                 }
424               }
425
426               if (paso_programar == 0) break; // Si se ha pulsado Salir en la pantalla de los dias, se romple el while (1) exterior
```

```
427               }
428               HAL_Delay(200);
429             }
430           }
431           else if (x >= 166 && x <= 214 && y >= 146 && y <= 194) // Se pulsa modo manual. Centro: x =
      190, y = 170
432           {
433             modo = 0; // Se pasa a modo manual
434             PantallaManual(tempManual); // Se muestra la pantalla manual
435             HAL_Delay(200); // Delay para evitar una pulsacion no deseada
436
437             while (1)
438             {
439               BSP_TS_GetState(&TS_State);
440
441               if (TS_State.TouchDetected)
442               {
443                 x = TS_State.X;
444                 y = TS_State.Y;
445
446                 if (x >= 26 && x <= 74 && y >= 146 && y <= 194) // Se disminuye 0.5 la temperatura
      deseada
447                 {
448                   tempManual -= 0.5;
449                   PantallaManual(tempManual); // Se muestra el cambio
450                 }
451                 else if (x >= 166 && x <= 214 && y >= 146 && y <= 194) // Se aumenta 0.5 la
      temperatura deseada
452                 {
453                   tempManual += 0.5;
454                   PantallaManual(tempManual); // Se muestra el cambio
455                 }
456                 else if (x >= 96 && x <= 144 && y >= 246 && y <= 294) // Se guarda la temperatura
      deseada del modo manual y se vuelve al inicio
457                 {
458                   // Se vuelve al inicio
459                   PantallaInicial(tempSensor, modo);
460                   HAL_Delay(100); // Delay para evitar una pulsacion no deseada
461                   break;
462                 }
463               }
464               HAL_Delay(200);
465             }
466           }
467           else if (x >= 26 && x <= 74 && y >= 146 && y <= 194) // Se pulsa configurar RTC. Centro: x =
      50, y = 170
468           {
469             int pasoRTC = 1; // Paso en la configuracion del RTC para saber que dato se esta
      configurando
470
471             // Se obtiene el valor actual del tiempo
472             HAL_RTC_GetTime(&hrtc, &sTime, RTC_FORMAT_BIN);
473             // Se obtiene el valor actual de la fecha
474             HAL_RTC_GetDate(&hrtc, &sDate, RTC_FORMAT_BIN);
475             // Se muestra la pantalla de configuracion del RTC
476             PantallaRTC(sTime.Hours, pasoRTC);
477
478             HAL_Delay(200); // Delay para evitar una pulsacion no deseada
479
480             while (1)
481             {
482               BSP_TS_GetState(&TS_State); // Rutina para obtener la informacion de la pantalla al pulsar
483
484               if (TS_State.TouchDetected) // Se detecta una pulsacion sobre la pantalla
485               {
486                 // Se guardan las coordenadas al pulsar sobre la pantalla
487                 x = TS_State.X;
488                 y = TS_State.Y;
489
490                 if (x >= 166 && x <= 214 && y >= 146 && y <= 194) // Se aumenta el dato del RTC en
      funcion del paso en el que se este
491                 {
492                   if (pasoRTC == 1) // Se aumenta la hora
493                   {
494                     sTime.Hours++;
495
496                     if (sTime.Hours > 23) sTime.Hours = 0;
```

```
497                     else if (sTime.Hours < 0) sTime.Hours = 23;
498
499                     PantallaRTC(sTime.Hours, pasoRTC); // Se muestra el cambio
500                   }
501                   else if (pasoRTC == 2) // Se aumentan los minutos
502                   {
503                     sTime.Minutes++;
504
505                     if (sTime.Minutes > 59) sTime.Minutes = 0;
506                     else if (sTime.Minutes < 0) sTime.Minutes = 59;
507
508                     PantallaRTC(sTime.Minutes, pasoRTC); // Se muestra el cambio
509                   }
510                   else if (pasoRTC == 3) // Se aumenta el dia de la semana
511                   {
512                     sDate.WeekDay++;
513
514                     if (sDate.WeekDay == 8) sDate.WeekDay = 1;
515                     else if (sDate.WeekDay == 0) sDate.WeekDay = 7;
516
517                     PantallaRTC(sDate.WeekDay, pasoRTC); // Se muestra el cambio
518                   }
519                   else if (pasoRTC == 4) // Se aumenta el dia
520                   {
521                     sDate.Date++;
522
523                     if (sDate.Date == 32) sDate.Date = 1;
524                     else if (sDate.Date == 0) sDate.Date = 31;
525
526                     PantallaRTC(sDate.Date, pasoRTC); // Se muestra el cambio
527                   }
528                   else if (pasoRTC == 5) // Se aumenta el mes
529                   {
530                     sDate.Month++;
531
532                     if (sDate.Month == 13) sDate.Month = 1;
533                     else if (sDate.Month == 0) sDate.Month = 12;
534
535                     PantallaRTC(sDate.Month, pasoRTC); // Se muestra el cambio
536                   }
537                   else if (pasoRTC == 6) // Se aumenta el año
538                   {
539                     sDate.Year++;
540                     PantallaRTC(sDate.Year, pasoRTC); // Se muestra el cambio
541                   }
542                 }
543                 else if (x >= 26 && x <= 74 && y >= 146 && y <= 194) // Se disminuye el dato del RTC
    en funcion del paso en el que se esta
544                 {
545                   if (pasoRTC == 1) // Se disminuye la hora
546                   {
547                     sTime.Hours--;
548
549                     if (sTime.Hours > 23) sTime.Hours = 0;
550                     else if (sTime.Hours < 0) sTime.Hours = 23;
551
552                     PantallaRTC(sTime.Hours, pasoRTC); // Se muestra el cambio
553                   }
554                   else if (pasoRTC == 2) // Se disminuyen los minutos
555                   {
556                     sTime.Minutes--;
557
558                     if (sTime.Minutes > 59) sTime.Minutes = 0;
559                     else if (sTime.Minutes < 0) sTime.Minutes = 59;
560
561                     PantallaRTC(sTime.Minutes, pasoRTC); // Se muestra el cambio
562                   }
563                   else if (pasoRTC == 3) // Se disminuye el dia de la semana
564                   {
565                     sDate.WeekDay--;
566
567                     if (sDate.WeekDay == 8) sDate.WeekDay = 1;
568                     else if (sDate.WeekDay == 0) sDate.WeekDay = 7;
569
570                     PantallaRTC(sDate.WeekDay, pasoRTC); // Se muestra el cambio
571                   }
572                   else if (pasoRTC == 4) // Se disminuye el dia
```

```
573                   {
574                     sDate.Date--;
575 
576                     if (sDate.Date == 32) sDate.Date = 1;
577                     else if (sDate.Date == 0) sDate.Date = 31;
578 
579                     PantallaRTC(sDate.Date, pasoRTC); // Se muestra el cambio
580                   }
581                   else if (pasoRTC == 5) // Se disminuye el mes
582                   {
583                     sDate.Month--;
584                     if (sDate.Month == 13) sDate.Month = 1;
585                     else if (sDate.Month == 0) sDate.Month = 12;
586 
587                     PantallaRTC(sDate.Month, pasoRTC); // Se muestra el cambio
588                   }
589                   else if (pasoRTC == 6) // Se diminuye el año
590                   {
591                     sDate.Year--;
592                     PantallaRTC(sDate.Year, pasoRTC);
593                   }
594                 }
595                 else if (x >= 26 && x <= 74 && y >= 246 && y <= 294) // Se pulsa OK y se pasa al
    siguiente dato o se guarda la configuracion
596                 {
597                   pasoRTC++; // Se aumenta el paso (se avanza a otro dato RTC)
598 
599                   // Se actualiza la pantalla en funcion del paso
600                   if (pasoRTC == 2) PantallaRTC(sTime.Minutes, pasoRTC);
601                   else if (pasoRTC == 3) PantallaRTC(sDate.WeekDay, pasoRTC);
602                   else if (pasoRTC == 4) PantallaRTC(sDate.Date, pasoRTC);
603                   else if (pasoRTC == 5) PantallaRTC(sDate.Month, pasoRTC);
604                   else if (pasoRTC == 6) PantallaRTC(sDate.Year, pasoRTC);
605                   else // Fin de la configuracion
606                   {
607                     // Se guardan las elecciones y se sale del bucle while(1) al inicio
608                     // Fijar la hora del RTC
609                     sTime.Seconds = 0;
610                     HAL_RTC_SetTime(&hrtc, &sTime, RTC_FORMAT_BIN);
611                     // Fijar fechar del RTC
612                     HAL_RTC_SetDate(&hrtc, &sDate, RTC_FORMAT_BIN);
613                     PantallaInicial(tempSensor, modo);
614                     break;
615                   }
616                 }
617                 else if (x >= 166 && x <= 214 && y >= 246 && y <= 294) // Se pulsa BACK y se vuelve al
    dato RTC anterior o se guarda la configuracion
618                 {
619                   pasoRTC--; // Se disminuye el paso (se retrocede a otro dato RTC)
620 
621                   // Se actualiza la pantalla en funcion del paso
622                   if (pasoRTC == 1) PantallaRTC(sTime.Hours, pasoRTC);
623                   else if (pasoRTC == 2) PantallaRTC(sTime.Minutes, pasoRTC);
624                   else if (pasoRTC == 3) PantallaRTC(sDate.WeekDay, pasoRTC);
625                   else if (pasoRTC == 4) PantallaRTC(sDate.Date, pasoRTC);
626                   else if (pasoRTC == 5) PantallaRTC(sDate.Month, pasoRTC);
627                   else // Fin de la configuracion
628                   {
629                     // Se guardan las elecciones y se sale del bucle while(1) al inicio
630                     // Fijar la hora del RTC
631                     sTime.Seconds = 0;
632                     HAL_RTC_SetTime(&hrtc, &sTime, RTC_FORMAT_BIN);
633                     // Fijar fecha del RTC
634                     HAL_RTC_SetDate(&hrtc, &sDate, RTC_FORMAT_BIN);
635                     PantallaInicial(tempSensor, modo);
636                     break;
637                   }
638 
639                 }
640               }
641               HAL_Delay(200);
642             }
643           }
644         }
645       }
646       HAL_Delay(100);
647     }
```

```
  /* USER CODE END 3 */

}

/** System Clock Configuration
*/
void SystemClock_Config(void)
{

  RCC_OscInitTypeDef RCC_OscInitStruct;
  RCC_ClkInitTypeDef RCC_ClkInitStruct;
  RCC_PeriphCLKInitTypeDef PeriphClkInitStruct;

    /**Configure the main internal regulator output voltage
    */
  __HAL_RCC_PWR_CLK_ENABLE();

  __HAL_PWR_VOLTAGESCALING_CONFIG(PWR_REGULATOR_VOLTAGE_SCALE1);

    /**Initializes the CPU, AHB and APB busses clocks
    */
  RCC_OscInitStruct.OscillatorType = RCC_OSCILLATORTYPE_LSI|RCC_OSCILLATORTYPE_HSE;
  RCC_OscInitStruct.HSEState = RCC_HSE_ON;
  RCC_OscInitStruct.LSIState = RCC_LSI_ON;
  RCC_OscInitStruct.PLL.PLLState = RCC_PLL_ON;
  RCC_OscInitStruct.PLL.PLLSource = RCC_PLLSOURCE_HSE;
  RCC_OscInitStruct.PLL.PLLM = 8;
  RCC_OscInitStruct.PLL.PLLN = 360;
  RCC_OscInitStruct.PLL.PLLP = RCC_PLLP_DIV2;
  RCC_OscInitStruct.PLL.PLLQ = 4;
  if (HAL_RCC_OscConfig(&RCC_OscInitStruct) != HAL_OK)
  {
    _Error_Handler(__FILE__, __LINE__);
  }

    /**Activate the Over-Drive mode
    */
  if (HAL_PWREx_EnableOverDrive() != HAL_OK)
  {
    _Error_Handler(__FILE__, __LINE__);
  }

    /**Initializes the CPU, AHB and APB busses clocks
    */
  RCC_ClkInitStruct.ClockType = RCC_CLOCKTYPE_HCLK|RCC_CLOCKTYPE_SYSCLK
                              |RCC_CLOCKTYPE_PCLK1|RCC_CLOCKTYPE_PCLK2;
  RCC_ClkInitStruct.SYSCLKSource = RCC_SYSCLKSOURCE_PLLCLK;
  RCC_ClkInitStruct.AHBCLKDivider = RCC_SYSCLK_DIV1;
  RCC_ClkInitStruct.APB1CLKDivider = RCC_HCLK_DIV4;
  RCC_ClkInitStruct.APB2CLKDivider = RCC_HCLK_DIV2;

  if (HAL_RCC_ClockConfig(&RCC_ClkInitStruct, FLASH_LATENCY_5) != HAL_OK)
  {
    _Error_Handler(__FILE__, __LINE__);
  }

  PeriphClkInitStruct.PeriphClockSelection = RCC_PERIPHCLK_RTC;
  PeriphClkInitStruct.RTCClockSelection = RCC_RTCCLKSOURCE_LSI;
  if (HAL_RCCEx_PeriphCLKConfig(&PeriphClkInitStruct) != HAL_OK)
  {
    _Error_Handler(__FILE__, __LINE__);
  }

    /**Configure the Systick interrupt time
    */
  HAL_SYSTICK_Config(HAL_RCC_GetHCLKFreq()/1000);

    /**Configure the Systick
    */
  HAL_SYSTICK_CLKSourceConfig(SYSTICK_CLKSOURCE_HCLK);

  /* SysTick_IRQn interrupt configuration */
  HAL_NVIC_SetPriority(SysTick_IRQn, 0, 0);
}

/* ADC1 init function */
static void MX_ADC1_Init(void)
```

```
725   {
726
727     ADC_ChannelConfTypeDef sConfig;
728
729       /**Configure the global features of the ADC (Clock, Resolution, Data Alignment and number of
      conversion)
730       */
731     hadc1.Instance = ADC1;
732     hadc1.Init.ClockPrescaler = ADC_CLOCK_SYNC_PCLK_DIV4;
733     hadc1.Init.Resolution = ADC_RESOLUTION_12B;
734     hadc1.Init.ScanConvMode = DISABLE;
735     hadc1.Init.ContinuousConvMode = DISABLE;
736     hadc1.Init.DiscontinuousConvMode = DISABLE;
737     hadc1.Init.ExternalTrigConvEdge = ADC_EXTERNALTRIGCONVEDGE_NONE;
738     hadc1.Init.ExternalTrigConv = ADC_SOFTWARE_START;
739     hadc1.Init.DataAlign = ADC_DATAALIGN_RIGHT;
740     hadc1.Init.NbrOfConversion = 1;
741     hadc1.Init.DMAContinuousRequests = DISABLE;
742     hadc1.Init.EOCSelection = ADC_EOC_SINGLE_CONV;
743     if (HAL_ADC_Init(&hadc1) != HAL_OK)
744     {
745       _Error_Handler(__FILE__, __LINE__);
746     }
747
748       /**Configure for the selected ADC regular channel its corresponding rank in the sequencer and
      its sample time.
749       */
750     sConfig.Channel = ADC_CHANNEL_13;
751     sConfig.Rank = 1;
752     sConfig.SamplingTime = ADC_SAMPLETIME_28CYCLES;
753     if (HAL_ADC_ConfigChannel(&hadc1, &sConfig) != HAL_OK)
754     {
755       _Error_Handler(__FILE__, __LINE__);
756     }
757
758   }
759
760   /* RTC init function */
761   static void MX_RTC_Init(void)
762   {
763
764       /**Initialize RTC Only
765       */
766     hrtc.Instance = RTC;
767     hrtc.Init.HourFormat = RTC_HOURFORMAT_24;
768     hrtc.Init.AsynchPrediv = 127;
769     hrtc.Init.SynchPrediv = 255;
770     hrtc.Init.OutPut = RTC_OUTPUT_DISABLE;
771     hrtc.Init.OutPutPolarity = RTC_OUTPUT_POLARITY_HIGH;
772     hrtc.Init.OutPutType = RTC_OUTPUT_TYPE_OPENDRAIN;
773     if (HAL_RTC_Init(&hrtc) != HAL_OK)
774     {
775       _Error_Handler(__FILE__, __LINE__);
776     }
777
778       /**Initialize RTC and set the Time and Date
779       */
780     if(HAL_RTCEx_BKUPRead(&hrtc, RTC_BKP_DR0) != 0x32F2){
781     if (HAL_RTC_SetTime(&hrtc, &sTime, RTC_FORMAT_BIN) != HAL_OK)
782     {
783       _Error_Handler(__FILE__, __LINE__);
784     }
785
786     if (HAL_RTC_SetDate(&hrtc, &sDate, RTC_FORMAT_BIN) != HAL_OK)
787     {
788       _Error_Handler(__FILE__, __LINE__);
789     }
790
791       HAL_RTCEx_BKUPWrite(&hrtc,RTC_BKP_DR0,0x32F2);
792     }
793
794   }
795
796   /* UART5 init function */
797   static void MX_UART5_Init(void)
798   {
799
```

```
800     huart5.Instance = UART5;
801     huart5.Init.BaudRate = 9600;
802     huart5.Init.WordLength = UART_WORDLENGTH_8B;
803     huart5.Init.StopBits = UART_STOPBITS_1;
804     huart5.Init.Parity = UART_PARITY_NONE;
805     huart5.Init.Mode = UART_MODE_TX_RX;
806     huart5.Init.HwFlowCtl = UART_HWCONTROL_NONE;
807     huart5.Init.OverSampling = UART_OVERSAMPLING_16;
808     if (HAL_UART_Init(&huart5) != HAL_OK)
809     {
810       _Error_Handler(__FILE__, __LINE__);
811     }
812
813   }
814
815   /** Configure pins as
816           * Analog
817           * Input
818           * Output
819           * EVENT_OUT
820           * EXTI
821   */
822   static void MX_GPIO_Init(void)
823   {
824
825     GPIO_InitTypeDef GPIO_InitStruct;
826
827     /* GPIO Ports Clock Enable */
828     __HAL_RCC_GPIOH_CLK_ENABLE();
829     __HAL_RCC_GPIOC_CLK_ENABLE();
830     __HAL_RCC_GPIOA_CLK_ENABLE();
831     __HAL_RCC_GPIOD_CLK_ENABLE();
832     __HAL_RCC_GPIOG_CLK_ENABLE();
833
834     /*Configure GPIO pin Output Level */
835     HAL_GPIO_WritePin(LED_GPIO_Port, LED_Pin, GPIO_PIN_RESET);
836
837     /*Configure GPIO pin : LED_Pin */
838     GPIO_InitStruct.Pin = LED_Pin;
839     GPIO_InitStruct.Mode = GPIO_MODE_OUTPUT_PP;
840     GPIO_InitStruct.Pull = GPIO_NOPULL;
841     GPIO_InitStruct.Speed = GPIO_SPEED_FREQ_HIGH;
842     HAL_GPIO_Init(LED_GPIO_Port, &GPIO_InitStruct);
843
844   }
845
846   /* USER CODE BEGIN 4 */
847   static void PantallaInicial(float temperatura, bool modo)
848   {
849     // Se limpia la pantalla
850     BSP_LCD_Clear(LCD_COLOR_WHITE); // Se borra y se pone el LCD en un color
851     BSP_LCD_SetBackColor(LCD_COLOR_WHITE); // Se pone el color del background
852     BSP_LCD_SetTextColor(LCD_COLOR_BLACK); // Se pone el color del texto
853
854     // Se muestra el modo actual
855     // Se muestran los Strings en el LCD usando la funcion BSP_LCD_DisplayStringAt(X, Y, String, Modo)
856     if (modo) BSP_LCD_DisplayStringAt(0, BSP_LCD_GetYSize()/2 - 130, "Modo:Programa", LEFT_MODE);
857     else BSP_LCD_DisplayStringAt(0, BSP_LCD_GetYSize()/2 - 130, "Modo:Manual", LEFT_MODE);
858
859     // Se muestra la temperatura actual
860     char buffer [14]; // Variable para guardar el String de la temperatura
861     // Se convierte el float en String. Formato %05.2f: 5 caracteres en total (el punto cuenta), 2
    cifras decimales y rellenar con 0 para llegar a 5
862     sprintf(buffer, "Temp:%05.2f C", temperatura);
863     // Se muestra la temperatura en la pantalla
864     BSP_LCD_DisplayStringAt(0, BSP_LCD_GetYSize()/2 - 90, buffer, LEFT_MODE);
865     // Se dibuja el simbolo de celsius(°) con BSP_LCD_DrawCircle(X, Y, Radio)
866     BSP_LCD_DrawCircle(celsiusX, celsiusY, celsiusR);
867     BSP_LCD_DrawCircle(celsiusX, celsiusY, celsiusR + 1);
868
869     // Separacion
870     // Se dibuja una linea horizontal con BSP_LCD_DrawHLine(X, Y, Longitud)
871     BSP_LCD_DrawHLine(0, separaSupY, BSP_LCD_GetXSize());
872     BSP_LCD_DrawHLine(0, separaSupY + 1, BSP_LCD_GetXSize());
873
874     // Se muestra el modo manual y la configuracion del RTC
875     BSP_LCD_DisplayStringAt(BSP_LCD_GetXSize()/2 - 110, BSP_LCD_GetYSize()/2 - 40, "RELOJ", LEFT_MODE);
```

```
876     BSP_LCD_DisplayStringAt(BSP_LCD_GetXSize()/2 + 20, BSP_LCD_GetYSize()/2 - 40, "MANUAL", LEFT_MODE);
877     // Boton Reloj de tiempo real
878     BSP_LCD_SetTextColor(LCD_COLOR_CYAN);
879     // Se dibuja un circulo relleno de un color con BSP_LCD_FillCircle(X, Y, Radio)
880     BSP_LCD_FillCircle(botonIzqX, botonSupY, botonR);
881     // Boton Modo manual
882     BSP_LCD_SetTextColor(LCD_COLOR_ORANGE);
883     BSP_LCD_FillCircle(botonDerX, botonSupY, botonR);
884     // Simbolo Calendario
885     BSP_LCD_SetTextColor(LCD_COLOR_BLACK);
886     // Se dibuja un rectangulo con BSP_LCD_DrawRect(X, Y, Ancho, Alto)
887     BSP_LCD_DrawRect(botonIzqX - (botonR/2) - 1, botonSupY - (botonR/2) - 1, botonR + 2, botonR + 2);
888     BSP_LCD_DrawRect(botonIzqX - (botonR/2), botonSupY - (botonR/2), botonR, botonR);
889     BSP_LCD_DrawRect(botonIzqX - (botonR/2) - 2, botonSupY - (botonR/2) - 2, botonR + 4, botonR + 4);
890     // Simbolo lapiz y linea
891     // Se dibuja una linea con BSP_LCD_DrawLine(X1, Y1, X2, Y2)
892     BSP_LCD_DrawLine(botonDerX + (botonR/2) - 1, botonSupY - (botonR/4), botonDerX - (botonR/2) - 1,
    botonSupY + (botonR/2));
893     BSP_LCD_DrawLine(botonDerX + (botonR/2), botonSupY - (botonR/4), botonDerX - (botonR/2), botonSupY
    + (botonR/2));
894     BSP_LCD_DrawLine(botonDerX + (botonR/2) + 1, botonSupY - (botonR/4), botonDerX - (botonR/2) + 1,
    botonSupY + (botonR/2));
895     BSP_LCD_DrawHLine(botonDerX - (botonR/2), botonSupY + 14, botonR);
896     BSP_LCD_DrawHLine(botonDerX - (botonR/2), botonSupY + 15, botonR);
897
898     // Separacion
899     BSP_LCD_DrawHLine(0, separaInfY, BSP_LCD_GetXSize());
900     BSP_LCD_DrawHLine(0, separaInfY + 1, BSP_LCD_GetXSize());
901
902     // Se muestra el modo programa
903     BSP_LCD_DisplayStringAt(BSP_LCD_GetXSize()/2 - 65, BSP_LCD_GetYSize()/2 + 60, "PROGRAMA",
    LEFT_MODE);
904     // Boton Modo programa
905     BSP_LCD_SetTextColor(LCD_COLOR_MAGENTA);
906     BSP_LCD_FillCircle(BSP_LCD_GetXSize()/2, botonInfY, botonR);
907     // Simbolo reloj
908     BSP_LCD_SetTextColor(LCD_COLOR_BLACK);
909     BSP_LCD_DrawCircle(BSP_LCD_GetXSize()/2, botonInfY, simbRelR - 1);
910     BSP_LCD_DrawCircle(BSP_LCD_GetXSize()/2, botonInfY, simbRelR);
911     BSP_LCD_DrawCircle(BSP_LCD_GetXSize()/2, botonInfY, simbRelR + 1);
912     BSP_LCD_DrawLine(BSP_LCD_GetXSize()/2 - 1, botonInfY, BSP_LCD_GetXSize()/2 - 1, botonInfY - 10);
913     BSP_LCD_DrawLine(BSP_LCD_GetXSize()/2, botonInfY, BSP_LCD_GetXSize()/2, botonInfY - 10);
914     BSP_LCD_DrawLine(BSP_LCD_GetXSize()/2 + 1, botonInfY, BSP_LCD_GetXSize()/2 + 1, botonInfY - 10);
915     BSP_LCD_DrawLine(BSP_LCD_GetXSize()/2, botonInfY - 1, BSP_LCD_GetXSize()/2 + 10, botonInfY - 1);
916     BSP_LCD_DrawLine(BSP_LCD_GetXSize()/2, botonInfY, BSP_LCD_GetXSize()/2 + 10, botonInfY);
917     BSP_LCD_DrawLine(BSP_LCD_GetXSize()/2, botonInfY + 1, BSP_LCD_GetXSize()/2 + 10, botonInfY + 1);
918   }
919
920   static void PantallaRTC(int datoRTC, int pasoRTC)
921   {
922     // Se limpia la pantalla
923     BSP_LCD_Clear(LCD_COLOR_WHITE); // Se borra y se pone el LCD en un color
924     BSP_LCD_SetTextColor(LCD_COLOR_BLACK); // Se pone el color del texto
925
926     // Se muestra la pantalla adecuada en funcion del paso
927     if (pasoRTC == 1) // Pantalla de la hora
928     {
929       BSP_LCD_DisplayStringAtLine(1, "Hora RTC");
930
931       char hour[10];
932       sprintf(hour, "Hora: %02d", datoRTC);
933       BSP_LCD_DisplayStringAtLine(2, hour);
934     }
935     else if (pasoRTC == 2) // Pantalla de los minutos
936     {
937       BSP_LCD_DisplayStringAtLine(1, "Minutos RTC");
938
939       char mins[10];
940       sprintf(mins, "Mins: %02d", datoRTC);
941       BSP_LCD_DisplayStringAtLine(2, mins);
942     }
943     else if (pasoRTC == 3) // Pantalla del dia de la semana
944     {
945       BSP_LCD_DisplayStringAtLine(1, "Dia Semana RTC");
946
947       char dia_semana[10];
948       char week_day[20];
```

```
949
950       // Se elige el dia de la semana en funcion del dato
951       if (datoRTC == 1) sprintf(dia_semana, "Lunes");
952       else if(datoRTC == 2) sprintf(dia_semana, "Martes");
953       else if(datoRTC == 3) sprintf(dia_semana, "Miercoles");
954       else if(datoRTC == 4) sprintf(dia_semana, "Jueves");
955       else if(datoRTC == 5) sprintf(dia_semana, "Viernes");
956       else if(datoRTC == 6) sprintf(dia_semana, "Sabado");
957       else if(datoRTC == 7) sprintf(dia_semana, "Domingo");
958
959       sprintf(week_day, "Dia: %s", dia_semana);
960       BSP_LCD_DisplayStringAtLine(2, week_day);
961     }
962     else if (pasoRTC == 4) // Pantalla del dia
963     {
964       BSP_LCD_DisplayStringAtLine(1, "Dia RTC");
965
966       char day[10];
967       sprintf(day, "Dia: %02d", datoRTC);
968       BSP_LCD_DisplayStringAtLine(2, day);
969     }
970     else if (pasoRTC == 5) // Pantalla del mes
971     {
972       BSP_LCD_DisplayStringAtLine(1, "Mes RTC");
973
974       char month[10];
975       sprintf(month, "Mes: %02d", datoRTC);
976       BSP_LCD_DisplayStringAtLine(2, month);
977     }
978     else if (pasoRTC == 6) // Pantalla del año
979     {
980       BSP_LCD_DisplayStringAtLine(1, "Ano RTC");
981
982       char year[12];
983       sprintf(year, "Ano: %4d", datoRTC + 2000);
984       BSP_LCD_DisplayStringAtLine(2, year);
985     }
986
987     // Separacion
988     BSP_LCD_DrawHLine(0, separaSupY, BSP_LCD_GetXSize()); // Se dibuja una linea horizontal con
    (Xi,Yi,Longitud)
989     BSP_LCD_DrawHLine(0, separaSupY + 1, BSP_LCD_GetXSize());
990
991     // Se muestra Control del dato RTC
992     BSP_LCD_DisplayStringAtLine(5, " Control RTC");
993     // Boton de incrementar valor
994     BSP_LCD_SetTextColor(LCD_COLOR_LIGHTRED);
995     BSP_LCD_FillCircle(botonDerX, botonSupY, botonR); // Se dibuja un circulo con relleno
996     // Boton de decrementar valor
997     BSP_LCD_SetTextColor(LCD_COLOR_LIGHTBLUE);
998     BSP_LCD_FillCircle(botonIzqX, botonSupY, botonR);
999     // Simbolo +
1000    BSP_LCD_SetTextColor(LCD_COLOR_BLACK);
1001    BSP_LCD_DrawHLine(botonDerX - (botonR/2), botonSupY - 1, botonR);
1002    BSP_LCD_DrawHLine(botonDerX - (botonR/2), botonSupY, botonR);
1003    BSP_LCD_DrawHLine(botonDerX - (botonR/2), botonSupY + 1, botonR);
1004    BSP_LCD_DrawVLine(botonDerX - 1, botonSupY - (botonR/2), botonR);
1005    BSP_LCD_DrawVLine(botonDerX, botonSupY - (botonR/2), botonR);
1006    BSP_LCD_DrawVLine(botonDerX + 1, botonSupY - (botonR/2), botonR);
1007    // Simbolo -
1008    BSP_LCD_DrawHLine(botonIzqX - (botonR/2), botonSupY - 1, botonR);
1009    BSP_LCD_DrawHLine(botonIzqX - (botonR/2), botonSupY, botonR);
1010    BSP_LCD_DrawHLine(botonIzqX - (botonR/2), botonSupY + 1, botonR);
1011
1012    // Separacion
1013    BSP_LCD_DrawHLine(0, separaInfY, BSP_LCD_GetXSize());
1014    BSP_LCD_DrawHLine(0, separaInfY + 1, BSP_LCD_GetXSize());
1015
1016    // Se muestra OK y BACK
1017    BSP_LCD_DisplayStringAt(BSP_LCD_GetXSize()/2 - 85, BSP_LCD_GetYSize()/2 + 60, "OK", LEFT_MODE);
1018    BSP_LCD_DisplayStringAt(BSP_LCD_GetXSize()/2 + 38, BSP_LCD_GetYSize()/2 + 60, "BACK", LEFT_MODE);
1019    BSP_LCD_SetTextColor(LCD_COLOR_GREEN);
1020    BSP_LCD_FillCircle(botonIzqX, botonInfY, botonR);
1021    BSP_LCD_SetTextColor(LCD_COLOR_RED);
1022    BSP_LCD_FillCircle(botonDerX, botonInfY, botonR);
1023  }
1024
```

```
1025  static void PantallaManual(float temperatura)
1026  {
1027    // Se limpia la pantalla
1028    BSP_LCD_Clear(LCD_COLOR_WHITE);
1029    BSP_LCD_SetTextColor(LCD_COLOR_BLACK);
1030
1031    // Se muestra la temperatura deseada del modo manual
1032    BSP_LCD_DisplayStringAt(25, 20, "Temperatura", LEFT_MODE);
1033    BSP_LCD_DisplayStringAt(60, 40, "deseada", LEFT_MODE);
1034
1035    char buffer [10];
1036    sprintf(buffer, "%05.2f C", temperatura);
1037    BSP_LCD_DisplayStringAt(60, 60, buffer, LEFT_MODE);
1038    BSP_LCD_DrawCircle(celsiusX - 25, celsiusY - 10, celsiusR);
1039    BSP_LCD_DrawCircle(celsiusX - 25, celsiusY - 10, celsiusR + 1);
1040
1041    // Separacion
1042    BSP_LCD_DrawHLine(0, separaSupY, BSP_LCD_GetXSize());
1043    BSP_LCD_DrawHLine(0, separaSupY + 1, BSP_LCD_GetXSize());
1044
1045    // Se muentra el control del termostato
1046    BSP_LCD_DisplayStringAtLine(5, " Control Temp");
1047    // Boton incrementar temperatura
1048    BSP_LCD_SetTextColor(LCD_COLOR_LIGHTRED);
1049    BSP_LCD_FillCircle(botonDerX, botonSupY, botonR);
1050    // Boton decrementar temperatura
1051    BSP_LCD_SetTextColor(LCD_COLOR_LIGHTBLUE);
1052    BSP_LCD_FillCircle(botonIzqX, botonSupY, botonR);
1053    // Simbolo +
1054    BSP_LCD_SetTextColor(LCD_COLOR_BLACK);
1055    BSP_LCD_DrawHLine(botonDerX - (botonR/2), botonSupY - 1, botonR);
1056    BSP_LCD_DrawHLine(botonDerX - (botonR/2), botonSupY, botonR);
1057    BSP_LCD_DrawHLine(botonDerX - (botonR/2), botonSupY + 1, botonR);
1058    BSP_LCD_DrawVLine(botonDerX - 1, botonSupY - (botonR/2), botonR);
1059    BSP_LCD_DrawVLine(botonDerX, botonSupY - (botonR/2), botonR);
1060    BSP_LCD_DrawVLine(botonDerX + 1, botonSupY - (botonR/2), botonR);
1061    // Simbolo -
1062    BSP_LCD_DrawHLine(botonIzqX - (botonR/2), botonSupY - 1, botonR);
1063    BSP_LCD_DrawHLine(botonIzqX - (botonR/2), botonSupY, botonR);
1064    BSP_LCD_DrawHLine(botonIzqX - (botonR/2), botonSupY + 1, botonR);
1065
1066    // Separacion
1067    BSP_LCD_DrawHLine(0, separaInfY, BSP_LCD_GetXSize());
1068    BSP_LCD_DrawHLine(0, separaInfY + 1, BSP_LCD_GetXSize());
1069
1070    // Se muestra Guardar
1071    BSP_LCD_DisplayStringAt(63, BSP_LCD_GetYSize()/2 + 60, "GUARDAR", LEFT_MODE);
1072    // Boton guardar
1073    BSP_LCD_SetTextColor(LCD_COLOR_GREEN);
1074    BSP_LCD_FillCircle(BSP_LCD_GetXSize()/2, botonInfY, botonR);
1075  }
1076
1077  static void PantallaProgTemp(float tempe, int paso)
1078  {
1079    // Se limpia la pantalla
1080    BSP_LCD_Clear(LCD_COLOR_WHITE);
1081    BSP_LCD_SetTextColor(LCD_COLOR_BLACK);
1082
1083    // Pantalla superior
1084    BSP_LCD_DisplayStringAtLine(1, " Temperaturas");
1085    BSP_LCD_DisplayStringAt(BSP_LCD_GetXSize()/2 + 60, 60, "C", LEFT_MODE);
1086    BSP_LCD_DrawCircle(celsiusX - 7, celsiusY - 10, celsiusR);
1087    BSP_LCD_DrawCircle(celsiusX - 7, celsiusY - 10, celsiusR + 1);
1088
1089    // Se muestra la temperatura
1090    char buffer [6]; // Variable para guardar el string de la temperatura
1091    sprintf(buffer, "%05.2f", tempe); // Se convierte el float en string
1092    BSP_LCD_DisplayStringAt(BSP_LCD_GetXSize()/2 - 35, 60, buffer, LEFT_MODE);
1093
1094    // En funcion del paso se muestra T1 o T2
1095    if (paso == 1) BSP_LCD_DisplayStringAt(BSP_LCD_GetXSize()/2 - 85, 60, "T1:", LEFT_MODE);
1096    else if (paso == 2) BSP_LCD_DisplayStringAt(BSP_LCD_GetXSize()/2 - 85, 60, "T2:", LEFT_MODE);
1097
1098    // Separacion
1099    BSP_LCD_DrawHLine(0, separaSupY, BSP_LCD_GetXSize()); // Se dibuja una linea horizontal con
      (Xi,Yi,Longitud)
1100    BSP_LCD_DrawHLine(0, separaSupY + 1, BSP_LCD_GetXSize());
```

```
1101
1102      // Se muentra el control del termostato
1103      BSP_LCD_DisplayStringAtLine(5, " Control Temp");
1104      // Boton incrementar temperatura
1105      BSP_LCD_SetTextColor(LCD_COLOR_LIGHTRED);
1106      BSP_LCD_FillCircle(botonDerX, botonSupY, botonR); // Se dibuja un circulo con relleno
1107      // Boton decrementar temperatura
1108      BSP_LCD_SetTextColor(LCD_COLOR_LIGHTBLUE);
1109      BSP_LCD_FillCircle(botonIzqX, botonSupY, botonR);
1110      // Simbolo +
1111      BSP_LCD_SetTextColor(LCD_COLOR_BLACK);
1112      BSP_LCD_DrawHLine(botonDerX - (botonR/2), botonSupY - 1, botonR);
1113      BSP_LCD_DrawHLine(botonDerX - (botonR/2), botonSupY, botonR);
1114      BSP_LCD_DrawHLine(botonDerX - (botonR/2), botonSupY + 1, botonR);
1115      BSP_LCD_DrawVLine(botonDerX - 1, botonSupY - (botonR/2), botonR);
1116      BSP_LCD_DrawVLine(botonDerX, botonSupY - (botonR/2), botonR);
1117      BSP_LCD_DrawVLine(botonDerX + 1, botonSupY - (botonR/2), botonR);
1118      // Simbolo -
1119      BSP_LCD_DrawHLine(botonIzqX - (botonR/2), botonSupY - 1, botonR);
1120      BSP_LCD_DrawHLine(botonIzqX - (botonR/2), botonSupY, botonR);
1121      BSP_LCD_DrawHLine(botonIzqX - (botonR/2), botonSupY + 1, botonR);
1122
1123      // Separacion
1124      BSP_LCD_DrawHLine(0, separaInfY, BSP_LCD_GetXSize());
1125      BSP_LCD_DrawHLine(0, separaInfY + 1, BSP_LCD_GetXSize());
1126
1127      // Pantalla inferior
1128      // Se muestra OK y BACK
1129      BSP_LCD_DisplayStringAt(BSP_LCD_GetXSize()/2 - 85, BSP_LCD_GetYSize()/2 + 60, "OK", LEFT_MODE);
1130      BSP_LCD_DisplayStringAt(BSP_LCD_GetXSize()/2 + 38, BSP_LCD_GetYSize()/2 + 60, "BACK", LEFT_MODE);
1131      BSP_LCD_SetTextColor(LCD_COLOR_GREEN);
1132      BSP_LCD_FillCircle(botonIzqX, botonInfY, botonR);
1133      BSP_LCD_SetTextColor(LCD_COLOR_RED);
1134      BSP_LCD_FillCircle(botonDerX, botonInfY, botonR);
1135    }
1136
1137    static void PantallaProgDias(void)
1138    {
1139      // Se limpia la pantalla
1140      BSP_LCD_Clear(LCD_COLOR_WHITE);
1141      BSP_LCD_SetTextColor(LCD_COLOR_BLACK);
1142      BSP_LCD_DisplayStringAt(0, 0, "Programa dias", LEFT_MODE);
1143
1144      // Se muestran los dias de la semana y salir
1145      BSP_LCD_DisplayStringAt(BSP_LCD_GetXSize()/2 - 7, BSP_LCD_GetYSize()/2 + 60, "L", LEFT_MODE);
1146      BSP_LCD_DisplayStringAt(BSP_LCD_GetXSize()/2 + 63, BSP_LCD_GetYSize()/2 + 60, "M", LEFT_MODE);
1147      BSP_LCD_DisplayStringAt(BSP_LCD_GetXSize()/2 - 77, BSP_LCD_GetYSize()/2 - 35, "X", LEFT_MODE);
1148      BSP_LCD_DisplayStringAt(BSP_LCD_GetXSize()/2 - 7, BSP_LCD_GetYSize()/2 - 35, "J", LEFT_MODE);
1149      BSP_LCD_DisplayStringAt(BSP_LCD_GetXSize()/2 + 63, BSP_LCD_GetYSize()/2 - 35, "V", LEFT_MODE);
1150      BSP_LCD_DisplayStringAt(BSP_LCD_GetXSize()/2 - 77, BSP_LCD_GetYSize()/2 - 135, "S", LEFT_MODE);
1151      BSP_LCD_DisplayStringAt(BSP_LCD_GetXSize()/2 - 7, BSP_LCD_GetYSize()/2 - 135, "D", LEFT_MODE);
1152      BSP_LCD_DisplayStringAt(BSP_LCD_GetXSize()/2 + 30, BSP_LCD_GetYSize()/2 - 135, "Salir", LEFT_MODE);
1153
1154      // Boton Lunes
1155      BSP_LCD_SetTextColor(LCD_COLOR_MAGENTA);
1156      BSP_LCD_FillCircle(BSP_LCD_GetXSize()/2, botonInfY, botonR);
1157      // Boton Martes
1158      BSP_LCD_SetTextColor(LCD_COLOR_GREEN);
1159      BSP_LCD_FillCircle(botonDerX, botonInfY, botonR);
1160      // Boton Miercoles
1161      BSP_LCD_SetTextColor(LCD_COLOR_LIGHTBLUE);
1162      BSP_LCD_FillCircle(botonIzqX, botonSupY, botonR);
1163      // Boton Jueves
1164      BSP_LCD_SetTextColor(LCD_COLOR_ORANGE);
1165      BSP_LCD_FillCircle(BSP_LCD_GetXSize()/2, botonSupY, botonR);
1166      // Boton Viernes
1167      BSP_LCD_SetTextColor(LCD_COLOR_LIGHTRED);
1168      BSP_LCD_FillCircle(botonDerX, botonSupY, botonR);
1169      // Boton Sabado
1170      BSP_LCD_SetTextColor(LCD_COLOR_DARKGREEN);
1171      BSP_LCD_FillCircle(botonIzqX, botonSupY - 100, botonR);
1172      // Boton Domingo
1173      BSP_LCD_SetTextColor(LCD_COLOR_BROWN);
1174      BSP_LCD_FillCircle(BSP_LCD_GetXSize()/2, botonSupY - 100, botonR);
1175      // Boton Salir
1176      BSP_LCD_SetTextColor(LCD_COLOR_RED);
1177      BSP_LCD_FillCircle(botonDerX, botonSupY - 100, botonR);
```

```
1178  }
1179
1180  static void PantallaProgHoras(int dia, int hora_inicio, int hora_fin, bool matriz_temp[24][7])
1181  {
1182    // Se limpia la pantalla
1183    BSP_LCD_Clear(LCD_COLOR_WHITE);
1184    BSP_LCD_SetTextColor(LCD_COLOR_BLACK);
1185
1186    // Se muestra el dia que se va a programar
1187    if (dia == 0) BSP_LCD_DisplayStringAtLine(1, "Dia: Lunes");
1188    else if (dia == 1) BSP_LCD_DisplayStringAtLine(1, "Dia: Martes");
1189    else if (dia == 2) BSP_LCD_DisplayStringAtLine(1, "Dia: Miercoles");
1190    else if (dia == 3) BSP_LCD_DisplayStringAtLine(1, "Dia: Jueves");
1191    else if (dia == 4) BSP_LCD_DisplayStringAtLine(1, "Dia: Viernes");
1192    else if (dia == 5) BSP_LCD_DisplayStringAtLine(1, "Dia: Sabado");
1193    else if (dia == 6) BSP_LCD_DisplayStringAtLine(1, "Dia: Domingo");
1194
1195    // Separacion
1196    BSP_LCD_DrawHLine(0, separaSupY - 50, BSP_LCD_GetXSize());
1197    BSP_LCD_DrawHLine(0, separaSupY - 49, BSP_LCD_GetXSize());
1198
1199    // Se muestra la hora de programacion de temperatura
1200    // Si la hora es las 24 (12 de la noche) se pasa a 0 para representarlo bien
1201    if (hora_fin == 24) hora_fin = 0;
1202
1203    // Se muestra la hora de configuracion
1204    char buffer_hora[12];
1205    BSP_LCD_DisplayStringAtLine(4, "Temperatura");
1206    BSP_LCD_DisplayStringAtLine(5, "entre las");
1207    sprintf(buffer_hora, "%02d:00-%02d:00", hora_inicio, hora_fin);
1208    BSP_LCD_DisplayStringAtLine(6, buffer_hora);
1209
1210    // Se muestra la temperatura actul a esa hora
1211    if (matriz_temp[hora_inicio][dia]) BSP_LCD_DisplayStringAtLine(7, "Actual: T2");
1212    else BSP_LCD_DisplayStringAtLine(7, "Actual: T1");
1213
1214    // Separacion
1215    BSP_LCD_DrawHLine(0, separaInfY, BSP_LCD_GetXSize());
1216    BSP_LCD_DrawHLine(0, separaInfY + 1, BSP_LCD_GetXSize());
1217
1218    // Se muestra T1, Salir y T2
1219    BSP_LCD_DisplayStringAt(BSP_LCD_GetXSize()/2 - 85, BSP_LCD_GetYSize()/2 + 60, "T1", LEFT_MODE);
1220    BSP_LCD_DisplayStringAt(BSP_LCD_GetXSize()/2 - 40, BSP_LCD_GetYSize()/2 + 60, "Salir", LEFT_MODE);
1221    BSP_LCD_DisplayStringAt(BSP_LCD_GetXSize()/2 + 56, BSP_LCD_GetYSize()/2 + 60, "T2", LEFT_MODE);
1222    // Boton T1
1223    BSP_LCD_SetTextColor(LCD_COLOR_GREEN);
1224    BSP_LCD_FillCircle(botonIzqX, botonInfY, botonR);
1225    // Boton Salir
1226    BSP_LCD_SetTextColor(LCD_COLOR_RED);
1227    BSP_LCD_FillCircle(BSP_LCD_GetXSize()/2, botonInfY, botonR);
1228    // Boton T2
1229    BSP_LCD_SetTextColor(LCD_COLOR_ORANGE);
1230    BSP_LCD_FillCircle(botonDerX, botonInfY, botonR);
1231  }
1232
1233  static void ActualizarMatriz(int dia, int hora_inicio, int hora_fin, bool matriz_temp[24][7])
1234  {
1235    HAL_Delay(200); // Delay para evitar que se pulse un boton indeseado
1236
1237    while (1)
1238    {
1239      BSP_TS_GetState(&TS_State);
1240
1241      if (TS_State.TouchDetected)
1242      {
1243        uint16_t x = TS_State.X;
1244        uint16_t y = TS_State.Y;
1245
1246        if (x >= 26 && x <= 74 && y >= 246 && y <= 294) // Se pulsa T1
1247        {
1248          matriz_temp[hora_inicio][dia] = 0; // Se guarda el dato T1 en la matriz
1249          // Se aumentan las horas de configuracion
1250          hora_inicio++;
1251          hora_fin++;
1252          PantallaProgHoras(dia, hora_inicio, hora_fin, matriz_temp); // Se actualiza la hora en la
      pantalla
1253        }
```

```
1254          else if (x >= 166 && x <= 214 && y >= 246 && y <= 294) // Se pulsa T2
1255          {
1256            matriz_temp[hora_inicio][dia] = 1; // Se guarda el dato T2 en la matriz
1257            // Se aumentan las horas de configuracion
1258            hora_inicio++;
1259            hora_fin++;
1260            PantallaProgHoras(dia, hora_inicio, hora_fin, matriz_temp); // Se actualiza la hora en la
      pantalla
1261          }
1262          else if (x >= 96 && x <= 144 && y >= 246 && y <= 294) // Se pulsa Salir
1263          {
1264            // Se pone la pantalla de los dias
1265            PantallaProgDias();
1266            break;
1267          }
1268
1269          if (hora_inicio > 23) // Todas las horas programadas, se vuelve a la pantalla de los dias
1270          {
1271            // Se pone la pantalla de los dias
1272            PantallaProgDias();
1273            break;
1274          }
1275        }
1276        HAL_Delay(350);
1277      }
1278    }
1279
1280    static void ControlCaldera (float tempSensor, int diaControl, int horaControl, float tempPrograma1,
      float tempPrograma2, bool matriz_temp[24][7])
1281    {
1282      // Control de la caldera en modo programa
1283      if (matriz_temp[horaControl][diaControl]) // Si la matriz tiene un 1, la temperatura deseada es T2
1284      {
1285        if (tempSensor <= (tempPrograma2 - 0.5)) HAL_GPIO_WritePin(LED_GPIO_Port, LED_Pin,
      GPIO_PIN_SET); // Temperatura inferior a la deseada, histeresis de 0.5 grado. Se enciende el LED lo
      que indica que se cierra el rele y se activa la caldera
1286        else if (tempSensor >= (tempPrograma2 + 0.5)) HAL_GPIO_WritePin(LED_GPIO_Port, LED_Pin,
      GPIO_PIN_RESET); // Temperatura superior a la deseada, histeresis de 0.5 grado. Se apaga el LED lo
      que indica que se abre el rele y se desactiva la caldera
1287      }
1288      else // Si la matriz tiene un 0, la temperatura deseada es T1
1289      {
1290        if (tempSensor <= (tempPrograma1 - 0.5)) HAL_GPIO_WritePin(LED_GPIO_Port, LED_Pin,
      GPIO_PIN_SET); // Temperatura inferior a la deseada, histeresis de 0.5 grado. Se enciende el LED lo
      que indica que se cierra el rele y se activa la caldera
1291        else if (tempSensor >= (tempPrograma1 + 0.5)) HAL_GPIO_WritePin(LED_GPIO_Port, LED_Pin,
      GPIO_PIN_RESET); // Temperatura superior a la deseada, histeresis de 0.5 grado. Se apaga el LED lo
      que indica que se abre el rele y se desactiva la caldera
1292      }
1293    }
1294
1295    static void ESP_Init(void)
1296    {
1297      // Se indica que se va a configurar el inicio del ESP
1298      BSP_LCD_DisplayStringAtLine(3, "INICIO ESP"); // Se muestra el String en la linea 3 del LCD
1299      HAL_Delay(2000);
1300
1301      // AT de test
1302      BSP_LCD_Clear(LCD_COLOR_WHITE);
1303      BSP_LCD_DisplayStringAtLine(3, "PASO 1: AT");
1304      HAL_Delay(1000);
1305
1306      BSP_LCD_Clear(LCD_COLOR_WHITE);
1307      uint8_t mensajeAT[] = "AT\r\n";
1308      uint8_t respuesta[10];
1309      HAL_UART_Transmit(&huart5, mensajeAT, sizeof(mensajeAT), 100); // Funcion para transmitir datos
      por el puerto UART, se indica el tiempo de proceso (100 ms)
1310      HAL_UART_Receive(&huart5, respuesta, sizeof(respuesta), 100); // Funcion para recibir datos por el
      puerto UART, se indica el tiempo de proceso (100 ms)
1311      BSP_LCD_DisplayStringAtLine(3, respuesta);
1312      HAL_Delay(2000);
1313
1314      // RESET
1315      BSP_LCD_Clear(LCD_COLOR_WHITE);
1316      BSP_LCD_DisplayStringAtLine(3, "PASO 2: RESET");
1317      HAL_Delay(1000);
1318
```

```
1319      BSP_LCD_Clear(LCD_COLOR_WHITE);
1320      uint8_t mensajeRST[] = "AT+RST\r\n";
1321      HAL_UART_Transmit(&huart5, mensajeRST, sizeof(mensajeRST), 100);
1322      HAL_UART_Receive(&huart5, respuesta, sizeof(respuesta), 100);
1323      BSP_LCD_DisplayStringAtLine(3, respuesta);
1324      HAL_Delay(2000);
1325
1326      // MODO 1 (Estacion de una red)
1327      BSP_LCD_Clear(LCD_COLOR_WHITE);
1328      BSP_LCD_DisplayStringAtLine(3, "PASO 3: MODO 1");
1329      HAL_Delay(1000);
1330
1331      BSP_LCD_Clear(LCD_COLOR_WHITE);
1332      uint8_t mensajeMode[] = "AT+CWMODE=1\r\n";
1333      HAL_UART_Transmit(&huart5, mensajeMode, sizeof(mensajeMode), 100);
1334      HAL_UART_Receive(&huart5, respuesta, sizeof(respuesta), 100);
1335      BSP_LCD_DisplayStringAtLine(3, respuesta);
1336      HAL_Delay(2000);
1337
1338      // CONEXION CON LA RED WIFI
1339      BSP_LCD_Clear(LCD_COLOR_WHITE);
1340      BSP_LCD_DisplayStringAtLine(3, "PASO 4: WIFI");
1341      HAL_Delay(1000);
1342
1343      BSP_LCD_Clear(LCD_COLOR_WHITE);
1344      uint8_t mensajeNet[] = "AT+CWJAP=\"SSID\",\"PASSWORD\"\r\n";
1345      HAL_UART_Transmit(&huart5, mensajeNet, sizeof(mensajeNet), 2000);
1346      HAL_UART_Receive(&huart5, respuesta, sizeof(respuesta), 2000);
1347      BSP_LCD_DisplayStringAtLine(3, respuesta);
1348      HAL_Delay(2000);
1349
1350      // HABILITAR LA CONEXION MULTIPLE
1351      BSP_LCD_Clear(LCD_COLOR_WHITE);
1352      BSP_LCD_DisplayStringAtLine(3, "PASO 5: CONEXION");
1353      HAL_Delay(1000);
1354
1355      BSP_LCD_Clear(LCD_COLOR_WHITE);
1356      uint8_t mensajeMux[] = "AT+CIPMUX=1\r\n";
1357      HAL_UART_Transmit(&huart5, mensajeMux, sizeof(mensajeMux), 100);
1358      HAL_UART_Receive(&huart5, respuesta, sizeof(respuesta), 100);
1359      BSP_LCD_DisplayStringAtLine(3, respuesta);
1360      HAL_Delay(2000);
1361
1362      // CREAR UN SERVIDOR
1363      // Solo se puede usar este comando si se esta en AT+CIPMUX=1 (conexion multiple)
1364      BSP_LCD_Clear(LCD_COLOR_WHITE);
1365      BSP_LCD_DisplayStringAtLine(3, "PASO 6: PUERTO");
1366      HAL_Delay(1000);
1367
1368      BSP_LCD_Clear(LCD_COLOR_WHITE);
1369      uint8_t mensajePort[] = "AT+CIPSERVER=1,80\r\n";
1370      HAL_UART_Transmit(&huart5, mensajePort, sizeof(mensajePort), 100);
1371      HAL_UART_Receive(&huart5, respuesta, sizeof(respuesta), 100);
1372      BSP_LCD_DisplayStringAtLine(3, respuesta);
1373      HAL_Delay(2000);
1374    }
1375
1376    static void ResponderHTTP(bool *modo, float tempSensor, float *tempManual, float *tempPrograma1,
      float *tempPrograma2, bool matriz_temp[24][7])
1377    {
1378      // Se guardan los datos que llegan desde el ESP a traves del UART5
1379      uint8_t peticionHTTP[150]; // Por debajo de 100 no va bien
1380      HAL_UART_Receive_IT(&huart5, peticionHTTP, sizeof(peticionHTTP)); // Funcion para recibir datos
      por el puerto UART mediante interrupcion
1381      // Se busca en los datos que han llegado si hay una peticion HTTP de la app. Si no hay
      coincidencida se devuelve un NULL
1382      char *puntero_ipd = strstr(peticionHTTP, "+IPD");
1383
1384      if (puntero_ipd != NULL) // Hay peticion HTTP
1385      {
1386        HAL_Delay(100); // Delay necesario para que vaya bien
1387
1388        int idESP = 0; // Por defecto se pone el id de la conexion a 0
1389
1390        // Se busca, en los datos que han llegado, que id tiene la conexion actual, puede ser 0, 1, 2, 3
      y 4
1391        // La funcion strstr() busca un string en otro string. Si no se encuentra coincidencia, devuelve
```

```
    NULL
1392        char *puntero_id1 = strstr(peticionHTTP, "+IPD,1");
1393        char *puntero_id2 = strstr(peticionHTTP, "+IPD,2");
1394        char *puntero_id3 = strstr(peticionHTTP, "+IPD,3");
1395        char *puntero_id4 = strstr(peticionHTTP, "+IPD,4");
1396
1397        // Se define cual es el id de la conexion actual del ESP
1398        if (puntero_id1 != NULL) idESP = 1;
1399        else if (puntero_id2 != NULL) idESP = 2;
1400        else if (puntero_id3 != NULL) idESP = 3;
1401        else if (puntero_id4 != NULL) idESP = 4;
1402
1403        // Se busca, en los datos que han llegado, cual es la peticion de la app
1404        char *puntero_home = strstr(peticionHTTP, "home");
1405        char *puntero_manual = strstr(peticionHTTP, "manual");
1406        char *puntero_programaTemps = strstr(peticionHTTP, "progtemps");
1407        char *puntero_enviarPrograma = strstr(peticionHTTP, "getprograma");
1408        char *puntero_programaDia = strstr(peticionHTTP, "progdia");
1409
1410        uint8_t dataSend[17]; // 17 es el tamaño del AT+CIPSEND=0,21\r\n, sin el \0
1411        uint8_t dataClose[15]; // 15 es el tamaño del AT+CIPCLOSE=0\r\n, sin el \0
1412
1413        if (puntero_home != NULL) // Hay una peticion para enviar el modo y la temperatura actuales
1414        {
1415          uint8_t httpHome[60] = "HTTP/1.1 200 OK\r\nContent-Type: text/html\r\n\r\n"; // Se indica que
    se trata de una repuesta HTTP
1416          // Se pone un tamaño de 60 para poder concatenar los otros strings
1417
1418          // Se concatena el modo al texto de la respuesta HTTP
1419          // La funcion strcat() concatena una cadena de texto al final de otra cadena
1420          if (*modo) strcat(httpHome, "Programa");
1421          else strcat(httpHome, "Manual");
1422
1423          char buffer[5]; // Variable para guardar el string de la temperatura
1424          sprintf(buffer, "%.2f", tempSensor); // Se convierte el float en string
1425          strcat(httpHome, buffer); // Se concatena la temperatura al modo
1426
1427          HAL_Delay(1000); // Delay necesario para que vaya bien
1428
1429          // Se indica al ESP que se va a producir un envio de datos indicando el id de la conexion y el
    tamaño de la informacion
1430          sprintf(dataSend, "AT+CIPSEND=%d,%d\r\n", idESP, sizeof(httpHome)); // los \r y \n cuentan
    como 1 caracter
1431          HAL_UART_Transmit(&huart5, dataSend, sizeof(dataSend), 100);
1432          HAL_Delay(200);
1433
1434          // Se realiza el envio de todo el texto HTTP concatenado con el modo y la temperatura
1435          HAL_UART_Transmit_IT(&huart5, httpHome, sizeof(httpHome));
1436          HAL_Delay(1000);
1437
1438          // Se cierra la conexion con el id indicado
1439          sprintf(dataClose, "AT+CIPCLOSE=%d\r\n", idESP);
1440          HAL_UART_Transmit(&huart5, dataClose, sizeof(dataClose), 100);
1441          HAL_Delay(200);
1442
1443          strcpy(peticionHTTP, "Hecho"); // Se modifican los datos guardados para evitar que el
    dispositivo pueda quedarse en un bucle
1444        }
1445        else if (puntero_manual != NULL) // Hay una peticion de pasar a modo manual, se recibe una
    temperatura
1446        {
1447          *modo = 0; // Se pasa a modo manual
1448          char tempManualApp[5]; // Variable donde se guarda la temperatura que se recibe desde la app
1449
1450          // Se copia, en el string, la temperatura que viene despues de "manual" (6 caracteres) en la
    peticion
1451          // La funcion strncpy() copia un string en un array
1452          strncpy(tempManualApp, puntero_manual + 6, 4);
1453          tempManualApp[4] = '\0';
1454
1455          *tempManual = atof(tempManualApp); // Se pasa la temperatura de string a float y se guarda en
    la variable pertinente
1456
1457          PantallaInicial(tempSensor, *modo); // Se actualiza la pantalla inicial
1458
1459          strcpy(peticionHTTP, "Hecho"); // Se modifican los datos guardados para evitar que el
    dispositivo pueda quedarse en un bucle
```

```
1460      }
1461      else if (puntero_programaTemps != NULL) // Hay una peticion para actualizar las dos temperaturas del modo programa, se reciben dos temperaturas
1462      {
1463        *modo = 1; // Se pasa a modo programa
1464        // Variables donde se guardan las temperaturas que se reciben desde la app
1465        char tempPrograma1App[5];
1466        char tempPrograma2App[5];
1467
1468        // Se copian, en los string, las temperaturas que vienen despues de "progtemps" (9 caracteres) en la peticion
1469        strncpy(tempPrograma1App, puntero_programaTemps + 9, 4);
1470        tempPrograma1App[4] = '\0';
1471        strncpy(tempPrograma2App, puntero_programaTemps + 13, 4);
1472        tempPrograma2App[4] = '\0';
1473
1474        // Se pasan las temperaturas de string a float y se guardan en las variables pertinentes
1475        *tempPrograma1 = atof(tempPrograma1App);
1476        *tempPrograma2 = atof(tempPrograma2App);
1477
1478        PantallaInicial(tempSensor, *modo); // Se actualiza la pantalla inicial
1479
1480        strcpy(peticionHTTP, "Hecho"); // Se modifican los datos guardados para evitar que el dispositivo pueda quedarse en un bucle
1481      }
1482      else if (puntero_enviarPrograma != NULL) // Hay una peticion para que se envie el programa actual de un dia de la semana, se recibe un dia
1483      {
1484        uint8_t httpPrograma[70] = "HTTP/1.1 200 OK\r\nContent-Type: text/html\r\n\r\n"; // Se indica que se trata de una repuesta HTTP
1485        // Se pone un tamaño de 70 para poder concatenar el programa de un dia
1486
1487        char diaEnvio[1]; // Variable donde se guarda el dia recibido
1488
1489        // Se copia el dia que viene despues de "getprograma" (11 caracteres) en la peticion
1490        strncpy(diaEnvio, puntero_enviarPrograma + 11, 1);
1491
1492        char programaChar[24]; // Variable donde se guarda, en Char, el programa del dia solicitado
1493
1494        // En funcion del dia solicitado se convierte el programa actual de Bool a Char y se envia a la app
1495        if (diaEnvio[0] == 'l') ProgramaBoolAChar(0, matriz_temp, programaChar); // Lunes
1496        else if (diaEnvio[0] == 'm') ProgramaBoolAChar(1, matriz_temp, programaChar); // Martes
1497        else if (diaEnvio[0] == 'x') ProgramaBoolAChar(2, matriz_temp, programaChar); // Miercoles
1498        else if (diaEnvio[0] == 'j') ProgramaBoolAChar(3, matriz_temp, programaChar); // Jueves
1499        else if (diaEnvio[0] == 'v') ProgramaBoolAChar(4, matriz_temp, programaChar); // Viernes
1500        else if (diaEnvio[0] == 's') ProgramaBoolAChar(5, matriz_temp, programaChar); // Sabado
1501        else if (diaEnvio[0] == 'd') ProgramaBoolAChar(6, matriz_temp, programaChar); // Domingo
1502
1503        strcat(httpPrograma, programaChar); // Se concatena el programa del dia al texto de la respuesta HTTP
1504
1505        HAL_Delay(1000); // Delay necesario para que vaya bien
1506
1507        // Se indica al ESP que se va a producir un envio de datos indicando el id de la conexion y el tamaño de la informacion
1508        sprintf(dataSend, "AT+CIPSEND=%d,%d\r\n", idESP, sizeof(httpPrograma)); // los \r y \n cuentan como 1 caracter
1509        HAL_UART_Transmit(&huart5, dataSend, sizeof(dataSend), 100);
1510        HAL_Delay(200);
1511
1512        // Se realiza el envio de todo el texto HTTP concatenado con el programa
1513        HAL_UART_Transmit_IT(&huart5, httpPrograma, sizeof(httpPrograma));
1514        HAL_Delay(1000);
1515
1516        // Se cierra la conexion con el id indicado
1517        sprintf(dataClose, "AT+CIPCLOSE=%d\r\n", idESP);
1518        HAL_UART_Transmit(&huart5, dataClose, sizeof(dataClose), 100);
1519        HAL_Delay(200);
1520
1521        strcpy(peticionHTTP, "Hecho"); // Se modifican los datos guardados para evitar que el dispositivo pueda quedarse en un bucle
1522      }
1523      else if (puntero_programaDia != NULL) // Hay una peticion para actualizar el programa de un dia de la semana, se recibe un dia y su programa
1524      {
1525        char diaApp[2]; // Variable donde se guarda el dia recibido
```

```
1526          char programaApp[25]; // Variable donde se guarda el programa recibido
1527
1528          // Se copia, en el string, el dia que viene despues de "progdia" (7 caracteres) en el HTTP
1529          strncpy(diaApp, puntero_programaDia + 7, 1);
1530          diaApp[1] = '\0';
1531          // Se copia, en el string, el programa que viene despues del dia en el HTTP
1532          strncpy(programaApp, puntero_programaDia + 8, 24);
1533          programaApp[24] = '\0';
1534
1535          // En funcion del dia solicitado se actualiza la matriz de temperaturas con el programa
      recibido
1536          if (diaApp[0] == 'l') ActualizarMatrizHTTP(0, programaApp, matriz_temp); // Lunes
1537          else if (diaApp[0] == 'm') ActualizarMatrizHTTP(1, programaApp, matriz_temp); // Martes
1538          else if (diaApp[0] == 'x') ActualizarMatrizHTTP(2, programaApp, matriz_temp); // Miercoles
1539          else if (diaApp[0] == 'j') ActualizarMatrizHTTP(3, programaApp, matriz_temp); // Jueves
1540          else if (diaApp[0] == 'v') ActualizarMatrizHTTP(4, programaApp, matriz_temp); // Viernes
1541          else if (diaApp[0] == 's') ActualizarMatrizHTTP(5, programaApp, matriz_temp); // Sabado
1542          else if (diaApp[0] == 'd') ActualizarMatrizHTTP(6, programaApp, matriz_temp); // Domingo
1543
1544          strcpy(peticionHTTP, "Hecho"); // Se modifican los datos guardados para evitar que el
      dispositivo pueda quedarse en un bucle
1545        }
1546      }
1547    }
1548    static void ProgramaBoolAChar(int dia, bool matriz_temp[24][7], char programaChar[24])
1549    {
1550      // Se recorren las filas (horas) de la columna (dia) de la matriz de temperaturas
1551      // y se pasan los valores que indican la temperatura de Bool a Char
1552      for (int i = 0; i < 24; i++)
1553      {
1554        if (matriz_temp[i][dia]) programaChar[i] = '1'; // La temperatura a la hora de ese dia es T2
1555        else programaChar[i] = '0'; // La temperatura es T1
1556      }
1557
1558    }
1559    static void ActualizarMatrizHTTP(int dia, char programaApp[25], bool matriz_temp[24][7])
1560    {
1561      // Se recorren las filas (horas) de la columna (dia) de la matriz de temperaturas
1562      // y se actualizan las temperaturas
1563      for (int i = 0; i < 24; i++)
1564      {
1565         if (programaApp[i] == '1') matriz_temp[i][dia] = 1; // Se ha marcado T2 en la app
1566         else matriz_temp[i][dia] = 0; // Se ha marcado T1 en la app
1567      }
1568    }
1569
1570    /* USER CODE END 4 */
1571
1572    /**
1573      * @brief  This function is executed in case of error occurrence.
1574      * @param  None
1575      * @retval None
1576      */
1577    void _Error_Handler(char * file, int line)
1578    {
1579      /* USER CODE BEGIN Error_Handler_Debug */
1580      /* User can add his own implementation to report the HAL error return state */
1581      while(1)
1582      {
1583      }
1584      /* USER CODE END Error_Handler_Debug */
1585    }
1586
1587    #ifdef USE_FULL_ASSERT
1588
1589    /**
1590       * @brief Reports the name of the source file and the source line number
1591       * where the assert_param error has occurred.
1592       * @param file: pointer to the source file name
1593       * @param line: assert_param error line source number
1594       * @retval None
1595       */
1596    void assert_failed(uint8_t* file, uint32_t line)
1597    {
1598      /* USER CODE BEGIN 6 */
1599      /* User can add his own implementation to report the file name and line number,
1600        ex: printf("Wrong parameters value: file %s on line %d\r\n", file, line) */
```

```
1601      /* USER CODE END 6 */
1602
1603    }
1604
1605    #endif
1606
1607    /**
1608      * @}
1609      */
1610
1611    /**
1612      * @}
1613    */
1614
1615    /************************ (C) COPYRIGHT STMicroelectronics *****END OF FILE****/
1616
```

Código Kotlin aplicación Android HomeT

```xml
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.home_t">

    <!-- Permiso para el uso de Internet en la app -->
    <uses-permission android:name="android.permission.INTERNET" />

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/Theme.HomeT"
        android:usesCleartextTraffic="true"> <!-- Permiso para poder usar una URL de tipo http. De serie solo permite https -->
        <activity
            android:name=".SundayProgramActivity"
            android:exported="false" />
        <activity
            android:name=".SaturdayProgramActivity"
            android:exported="false" />
        <activity
            android:name=".FridayProgramActivity"
            android:exported="false" />
        <activity
            android:name=".ThursdayProgramActivity"
            android:exported="false" />
        <activity
            android:name=".WednesdayProgramActivity"
            android:exported="false" />
        <activity
            android:name=".TuesdayProgramActivity"
            android:exported="false" />
        <activity
            android:name=".MondayProgramActivity"
            android:exported="false" />
        <activity
            android:name=".NewPasswordActivity"
            android:exported="false" />
        <activity
            android:name=".RegistrationActivity"
            android:exported="false" />
        <activity
            android:name=".HomeActivity"
            android:exported="false" />
        <activity
            android:name=".AuthenticationActivity"
            android:exported="true"
            android:theme="@style/SplashTheme"> <!-- Se pone como tema de la actividad inicial el tema de la Splash Screen -->
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
```

```
52
53                 <category android:name="android.intent.category.
   LAUNCHER" />
54             </intent-filter>
55         </activity>
56     </application>
57
58 </manifest>
```

```kt
package com.example.home_t

import android.content.Intent
import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import android.util.Patterns
import androidx.appcompat.app.AlertDialog
import com.example.home_t.databinding.ActivityAuthenticationBinding
import com.google.firebase.auth.FirebaseAuth

/*
    Actividad Authentication: Controla el inicio de sesion de la aplicacion, la pantalla inicial de la app.
    Tambien permite navegar a las actividades encargadas de controlar el cambio
    de contraseña (NewPasswordActivity), la creacion de una nueva cuenta (RegistrationActivity)
    y el menu de inicio de la aplicacion (HomeActivity con HomeFragment).
 */

class AuthenticationActivity : AppCompatActivity() {
    private lateinit var binding: ActivityAuthenticationBinding

    override fun onCreate(savedInstanceState: Bundle?) {
        // Se establece el tema normal de la app y se quita el Splash Screen
        setTheme(R.style.Theme_HomeT)

        super.onCreate(savedInstanceState)
        binding = ActivityAuthenticationBinding.inflate(layoutInflater)
        setContentView(binding.root)

        setupLogin() // Funcion que controla la logica de la actividad Authentication
    }

    // Se sobreescribe la funcion que controla lo que ocurre cuando se pulsa Back
    override fun onBackPressed() {
        // Cuando se pulsa Back, nos salimos de la app
        val intent = Intent(Intent.ACTION_MAIN)
        intent.addCategory(Intent.CATEGORY_HOME)
        intent.flags = Intent.FLAG_ACTIVITY_NEW_TASK
        startActivity(intent)
        finish()
    }

    private fun setupLogin() {
        // La funcion setOnClickListener nos permite poner ciertos elementos de la pantalla a la espera de un click sobre ellos

```

```
45          // Se pulsa en olvido contraseña, se inicia la actividad
    NewPassword
46          binding.forgotPasswordText.setOnClickListener {
47              startActivity(
48                  Intent(
49                      this,
50                      NewPasswordActivity::class.java
51                  )
52              )
53          }
54
55          // Se pulsa en crear cuenta, se inicia la actividad
    Registration
56          binding.signUpButton.setOnClickListener {
57              startActivity(
58                  Intent(
59                      this,
60                      RegistrationActivity::class.java
61                  )
62              )
63          }
64
65          // Se pulsa en inicio de sesion
66          binding.signInButton.setOnClickListener {
67
68              // Se comprueba que los campos de email y contraseña no
    estan vacios
69              if (binding.emailEditText.text.isNotEmpty() && binding.
    passwordEditText.text.isNotEmpty()) {
70
71                  // Se comprueba que el email introducido tiene un
    formato valido de email
72                  if (!Patterns.EMAIL_ADDRESS.matcher(binding.
    emailEditText.text).matches()) {
73
74                      binding.emailEditText.error = "Debe introducir un
    email válido"
75                      binding.emailEditText.requestFocus()
76
77                  } else {
78
79                      // Se comprueban el email y la contraseña
    introducidos en Firebase
80                      FirebaseAuth.getInstance().
    signInWithEmailAndPassword(
81                          binding.emailEditText.text.toString().trim(),
    // Se pasa a String y se eliminan los espacios en blanco
82                          binding.passwordEditText.text.toString().trim
    ()
83                      ).addOnCompleteListener {
84
85                          // Si la comprobacion ha sido correcta
86                          if (it.isSuccessful) {
```

```
87
88                              // Si la cuenta esta verificada, se
   inicia la actividad Home
89                              if (FirebaseAuth.getInstance().
   currentUser?.isEmailVerified == true) {
90
91                                  startActivity(Intent(this,
   HomeActivity::class.java))
92
93                              } else {
94                                  showError(1) // El email no esta
   verificado
95                              }
96
97                          } else {
98                              showError(2) // Se ha producido un error
   al iniciar la sesion
99                          }
100
101                     }
102                 }
103             } else {
104
105                 if (binding.emailEditText.text.isEmpty()) {
106
107                     binding.emailEditText.error = "Debe introducir un
    email"
108                     binding.emailEditText.requestFocus()
109
110                 } else {
111
112                     binding.passwordEditText.error = "Debe introducir
    una contraseña"
113                     binding.passwordEditText.requestFocus()
114
115                 }
116
117             }
118         }
119     }
120
121     // Funcion que maneja los errores al inicializar sesion, se
   muestra un cuadro de dialogo con botones
122     private fun showError(error: Int) {
123         // Se crea el cuadro de dialogo de ERROR
124         val errorAlert = AlertDialog.Builder(this)
125         errorAlert.setTitle("ERROR")
126
127         // En funcion del error, se completa el cuadro de dialogo con
    la informacion y los botones correspondientes
128         if (error == 1) {
129
130             errorAlert.setMessage("La cuenta no está verificada.
```

```
Pulse en Enviar para enviar un correo de verificación.")
            errorAlert.setNeutralButton("Enviar correo") { _, _ ->

                FirebaseAuth.getInstance().currentUser?.
sendEmailVerification() // Se envia un correo al email para verificar
 la cuenta

            }

        } else {

            errorAlert.setMessage("Se ha producido un error,
compruebe los datos introducidos.")

        }

        errorAlert.setPositiveButton("Aceptar", null) // El boton
Aceptar simplemente quita el cuadro de dialogo
        errorAlert.create().show()
    }
}
```

```kotlin
package com.example.home_t

import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import android.util.Patterns
import androidx.appcompat.app.AlertDialog
import com.example.home_t.databinding.ActivityRegistrationBinding
import com.google.firebase.auth.FirebaseAuth

/*
    Actividad Registration: Controla la creacion de una cuenta.
    Para crear una cuenta hace falta indicar un email que no este registrado y
    una contraseña valida.
 */

class RegistrationActivity : AppCompatActivity() {

    private lateinit var binding: ActivityRegistrationBinding

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        binding = ActivityRegistrationBinding.inflate(layoutInflater)
        setContentView(binding.root)

        setupRegistration() // Funcion que controla la logica de la actividad Registration
    }

    private fun setupRegistration() {

        // Se pulsa en volver, se vuelve a la actividad Authentication
        binding.goBackButton.setOnClickListener { onBackPressed() }

        // Se pulsa en crear cuenta
        binding.createAccountButton.setOnClickListener {

            // Se comprueba que los campos del email y de las contraseñas no esten vacios
            if (binding.emailEditText.text.isNotEmpty() && binding.passwordEditText.text.isNotEmpty()
                && binding.passwordEditText2.text.isNotEmpty()
            ) {

                // Se comprueba que el email introducido tiene un formato valido de email
                if (!Patterns.EMAIL_ADDRESS.matcher(binding.emailEditText.text).matches()) {

                    binding.emailEditText.error = "Debe introducir un email válido"
                    binding.emailEditText.requestFocus()

```

```
47                } else if (binding.passwordEditText.length() < 6) {
48
49                    // Por politica de Firebase, la contraseña debe
   tener al menos 6 caracteres
50                    binding.passwordEditText.error =
51                        "La contraseña debe tener al menos 6
   caracteres"
52                    binding.passwordEditText.requestFocus()
53
54                } else if (binding.passwordEditText.text.toString()
55                        .trim() != binding.passwordEditText2.text.
   toString().trim()
56                ) {
57
58                    // Las contraseña y la repeticion de la contraseña
    no coinciden
59                    binding.passwordEditText2.error = "Las contraseñas
    no coinciden"
60                    binding.passwordEditText2.requestFocus()
61
62                } else {
63
64                    // Se registra o crea la cuenta en Firebase
65                    FirebaseAuth.getInstance().
   createUserWithEmailAndPassword(
66                        binding.emailEditText.text.toString().trim(),
67                        binding.passwordEditText.text.toString().trim
   ()
68                    ).addOnCompleteListener {
69
70                        if (it.isSuccessful) {
71
72                            // Se manda un email de verificacion
73                            FirebaseAuth.getInstance().currentUser?.
   sendEmailVerification()
74                                ?.addOnCompleteListener { it2 ->
75
76                                    if (it2.isSuccessful) {
77
78                                        showAlert(1) // Se avisa del
   correo enviado
79
80                                    } else {
81
82                                        showAlert(2) // Se ha
   producido un error al enviar el correo de verificacion
83
84                                    }
85                                }
86
87                        } else {
88
89                            showAlert(3) // Se ha producido un error
```

```
89 en el proceso o el email ya esta registrada
90
91                         }
92                     }
93
94                 }
95
96             } else {
97
98                 when {
99
100                    binding.emailEditText.text.isEmpty() -> {
101
102                        binding.emailEditText.error = "Debe
    introducir un email"
103                        binding.emailEditText.requestFocus()
104
105                    }
106                    binding.passwordEditText.text.isEmpty() -> {
107
108                        binding.passwordEditText.error = "Debe
    introducir una contraseña"
109                        binding.passwordEditText.requestFocus()
110
111                    }
112                    binding.passwordEditText2.text.isEmpty() -> {
113
114                        binding.passwordEditText2.error = "Debe
    confirmar la contraseña"
115                        binding.passwordEditText2.requestFocus()
116
117                    }
118                }
119
120            }
121
122        }
123
124    }
125
126    // Funcion que maneja las alertas al crear una cuenta, se muestra
     un cuadro de dialogo con botones
127    private fun showAlert(alert: Int) {
128
129        // Se crea el cuadro de dialogo
130        val verifyAlert = AlertDialog.Builder(this)
131
132        // En funcion de la alerta, se completa el cuadro de dialogo
    con la informacion correspondiente
133        when (alert) {
134            1 -> {
135
136                verifyAlert.setTitle("VERIFICA TU CUENTA")
```

```
                verifyAlert.setMessage("Se ha enviado un correo para verificar su cuenta.")

            }
            2 -> {

                verifyAlert.setTitle("ERROR")
                verifyAlert.setMessage("Se ha producido un error al enviar el correo de verificación.")

            }
            3 -> {

                verifyAlert.setTitle("ERROR")
                verifyAlert.setMessage("Se ha producido un error en el proceso o la cuenta ya existe.")

            }
        }

        verifyAlert.setPositiveButton("Aceptar", null)
        verifyAlert.create().show()

    }
}

```

```kotlin
package com.example.home_t

import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import android.util.Patterns
import androidx.appcompat.app.AlertDialog
import com.example.home_t.databinding.ActivityNewPasswordBinding
import com.google.firebase.auth.FirebaseAuth

/*
    Actividad NewPassword: Controla el cambio de contraseña de una cuenta.
    Para cambiar la contraseña hace falta indicar el email y seguir las instrucciones
    que aparecen en el correo que se recibe.
 */

class NewPasswordActivity : AppCompatActivity() {

    private lateinit var binding: ActivityNewPasswordBinding

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        binding = ActivityNewPasswordBinding.inflate(layoutInflater)
        setContentView(binding.root)

        setupNewPw() // Funcion que controla la logica de la actividad NewPassword
    }

    private fun setupNewPw() {

        // Se pulsa en volver, se vuelve a la actividad Authentication
        binding.goBackButton.setOnClickListener { onBackPressed() }

        // Se pulsa enviar correo al email indicado
        binding.sendEmailButton.setOnClickListener {

            // Se comprueba que el campo del email no este vacio
            if (binding.emailEditText.text.isNotEmpty()) {

                // Se comprueba que el email introducido tiene un formato valido de email
                if (!Patterns.EMAIL_ADDRESS.matcher(binding.emailEditText.text).matches()) {

                    binding.emailEditText.error = "Debe introducir un email válido"
                    binding.emailEditText.requestFocus()

                } else {

                    // Se manda un correo para cambiar la contraseña
```

```
al email indicado
                    FirebaseAuth.getInstance()
                        .sendPasswordResetEmail(binding.emailEditText.
text.toString().trim())
                        .addOnCompleteListener {

                            // Si se ha enviado el correo
correctamente
                            if (it.isSuccessful) {
                                emailInfo() // Se avisa del correo
enviado
                            } else {
                                showError() // Se ha producido un
error
                            }
                        }
                }
            } else {

                binding.emailEditText.error = "Debe introducir un
email"
                binding.emailEditText.requestFocus()

            }
        }
    }

    private fun emailInfo() {
        val emailAlert = AlertDialog.Builder(this)
        emailAlert.setMessage("Siga las instrucciones en el email que
se ha enviado.")
        emailAlert.setPositiveButton("Aceptar", null)
        emailAlert.create().show()
    }

    private fun showError() {
        val errorAlert = AlertDialog.Builder(this)
        errorAlert.setTitle("ERROR")
        errorAlert.setMessage("Se ha producido un error, compruebe el
email introducido.")
        errorAlert.setPositiveButton("Aceptar", null)
        errorAlert.create().show()
    }
}
```

```kotlin
package com.example.home_t

import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import androidx.navigation.NavController
import androidx.navigation.fragment.NavHostFragment
import androidx.navigation.ui.setupWithNavController
import com.example.home_t.databinding.ActivityHomeBinding
import com.google.android.material.bottomnavigation.BottomNavigationView

/*
    Actividad Home: Actividad que contiene el menu de la app, el navegador y el contenedor de fragmentos.
    El contenedor de fragmentos mostrara el fragmento correspondiente a lo seleccionado en el menu del navegador.
 */

class HomeActivity : AppCompatActivity() {

    private lateinit var binding: ActivityHomeBinding
    private lateinit var navController: NavController

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)

        binding = ActivityHomeBinding.inflate(layoutInflater)
        setContentView(binding.root)

        val bottomNavigation = findViewById<BottomNavigationView>(R.id.bottomNavigation) // Se crea un objeto del navegador inferior
        val navHostFragment = supportFragmentManager.findFragmentById(R.id.fragmentContainerView) as NavHostFragment // Se crea un objeto del contenedor de fragmentos
        navController = navHostFragment.navController // Se asigna el control del contenedor a la variable navController


        bottomNavigation.setupWithNavController(navController) // Se enlaza el navegador con el contenedor
    }
}
```

```kotlin
package com.example.home_t.fragments.menu

import android.content.Intent

import android.os.Bundle
import android.view.LayoutInflater
import android.view.View
import android.view.ViewGroup
import androidx.fragment.app.Fragment
import com.example.home_t.AuthenticationActivity
import com.example.home_t.databinding.FragmentHomeBinding
import com.google.firebase.auth.FirebaseAuth
import okhttp3.*
import java.io.IOException

/*
    Fragmento Home: Fragmento de inicio al acceder a la aplicacion.
    Muestra la temperatura actual en la vivienda y el modo elegido.
    Se usa la biblioteca OkHttp para poder usar el protocolo HTTP.
 */

class HomeFragment : Fragment() {

    private var _binding: FragmentHomeBinding? = null
    private val binding get() = _binding!!

    // URL con la direccion IP del ESP
    private val ipUrl = "http://192.168.1.53/"

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setHasOptionsMenu(true)
    }

    override fun onCreateView(
        inflater: LayoutInflater,
        container: ViewGroup?,
        savedInstanceState: Bundle?
    ): View {
        _binding = FragmentHomeBinding.inflate(inflater, container,
false)
        return binding.root
    }

    override fun onViewCreated(view: View, savedInstanceState: Bundle
?) {
        getModeAndTemp() // Se obtienen, y se muestran, el modo y la
temperatura actuales en la vivienda
        setupHome() // Funcion que controla la logica del fragmento
Home
    }

    override fun onDestroyView() {
```

```
50          super.onDestroyView()
51          _binding = null
52      }
53
54      private fun getModeAndTemp() {
55          val client = OkHttpClient() // Se crea el cliente OkHttp
56          val url = ipUrl + "home" // Se completa la URL con un string
    identificativo del fragmento
57          // Se define la llamada HTTP
58          val request = Request.Builder()
59              .url(url)
60              .addHeader("Connection", "close")
61              .build()
62
63          // Se realiza la llamada HTTP de forma asincrona
64          client.newCall(request).enqueue(object : Callback {
65
66              // Si ocurre algun problema en la llamada
67              override fun onFailure(call: Call, e: IOException) {
68                  e.printStackTrace() // Se muestra la excepcion o
    error ocurrido
69              }
70
71              // Si se recibe una respuesta a la llamada
72              override fun onResponse(call: Call, response: Response
    ) { // Se recibe la respuesta de la llamada HTTP realizada
73                  response.use {
74                      if (response.isSuccessful) { // La respuesta se
    ha recibido correctamente
75                          val myResponse = response.body?.string() //
    Se guarda el cuerpo de la respuesta
76
77                          // Se filtra el cuerpo de la repuesta por
    letras (se consigue el modo actual)
78                          // y por digitos (se obtiene la temperatura
    actual)
79                          val mode = myResponse?.filter(Char::isLetter
    ) // Se obtiene el modo
80                          val temperatureDouble = myResponse?.filter(
    Char::isDigit)?.toDouble() // Se obtiene la temperatura
81
82                          // Se divide la temperatura obtenida entre
    100, ya que el numero enviado es de 4 digitos, y se convierte en
    String
83                          val temperature = temperatureDouble?.div(100
    ).toString() + " °C"
84
85                          // Los cambios visuales en la pantalla se
    hacen en el hilo donde corre el interfaz de usuario
86                          activity?.runOnUiThread {
87                              // La respuesta no es nula
88                              if (myResponse != null) {
89                                  binding.modeOnText.text = mode // Se
```

```
89 actualiza el modo actual
90                                 binding.temperatureOnText.text =
   temperature // Se actualiza la temperatura actual
91                             }
92                         }
93                     }
94
95                 }
96             }
97         }
98         )
99     }
100
101     private fun setupHome() {
102         binding.userText.text = FirebaseAuth.getInstance().
    currentUser?.email // Se muestra el email del usuario
103
104         // Si se pulsa en cerrar sesion, se cierra la cuenta y se
    inicia la actividad Authentication (volvemos atras)
105         binding.logoutButton.setOnClickListener {
106             FirebaseAuth.getInstance().signOut() // Cerramos la
    sesion en Firebase
107             startActivity(Intent(context, AuthenticationActivity::
    class.java)) // Se inicia la actividad Authentication
108         }
109     }
110 }
```

```kotlin
package com.example.home_t.fragments.menu

import android.os.Bundle
import android.view.Gravity
import android.view.LayoutInflater
import android.view.View
import android.view.ViewGroup
import android.widget.Toast
import androidx.fragment.app.Fragment
import com.example.home_t.databinding.FragmentManualModeBinding
import okhttp3.*
import java.io.IOException

/*
    Fragmento ManualMode: Fragmento del modo namual.
    Permite elegir la temperatura deseada y pasar a modo manual.
 */

class ManualModeFragment : Fragment() {

    private var _binding: FragmentManualModeBinding? = null
    private val binding get() = _binding!!

    // URL con la direccion IP del ESP
    private val ipUrl = "http://192.168.1.53/"

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setHasOptionsMenu(true)
    }

    override fun onCreateView(
        inflater: LayoutInflater,
        container: ViewGroup?,
        savedInstanceState: Bundle?
    ): View {
        _binding = FragmentManualModeBinding.inflate(inflater,
container, false)
        return binding.root
    }

    override fun onViewCreated(view: View, savedInstanceState: Bundle
?) {
        setupManualMode() // Funcion que controla la logica del
fragmento ManualMode
    }

    override fun onDestroyView() {
        super.onDestroyView()
        _binding = null
    }

    private fun setupManualMode() {
```

```
51          // Se pulsa en guardar
52          binding.saveButton.setOnClickListener {
53
54              // Se comprueba que el campo de la temperatura no este
    vacio
55              if (binding.temperatureEditText.text.isNotEmpty()) {
56
57                  val stringManualTemp =
58                      binding.temperatureEditText.text.toString() // Se
     pasa la temperatura introducida a string
59                  val manualTemp = stringManualTemp.toDouble() // Se
    pasa el string a double
60
61                  // Se comprueba que la temperatura introducida es
    multiplo de 0.5
62                  if ((manualTemp % 0.5) == 0.0) {
63
64                      sendManualTemp(manualTemp) // Se envia la
    temperatura manual al microcontrolador
65
66                      // Se avisa con un Toast que se ha enviado la
    temperatura
67                      val toast = Toast.makeText(context, "Temperatura
    enviada", Toast.LENGTH_SHORT)
68                      toast.setGravity(Gravity.CENTER, 0, 0)
69                      toast.show()
70
71                  } else {
72                      showError(2) // La temperatura no es multiplo de
    0.5
73                  }
74
75              } else {
76                  showError(1) // No se ha introducido ninguna
    temperatura
77              }
78          }
79      }
80
81      private fun sendManualTemp(manualTemp: Double) {
82          val client = OkHttpClient() // Se crea el cliente OkHttp
83          val url =
84              ipUrl + "manual" + manualTemp  // Se completa la URL con
    un string identificativo del fragmento y la temperatura manual
    deseada
85          // Se define la llamada HTTP
86          val request = Request.Builder()
87              .url(url)
88              .addHeader("Connection", "close")
89              .build()
90
91          // Se realiza la llamada HTTP de forma asincrona
92          client.newCall(request).enqueue(object : Callback {
```

```
93
94              // Si ocurre algun problema en la llamada
95              override fun onFailure(call: Call, e: IOException) {
96                  e.printStackTrace()
97              }
98
99              override fun onResponse(call: Call, response: Response) {
100                 // Como no tenemos nada que recibir, no hacemos nada
    con la respuesta
101             }
102         })
103     }
104
105     // Funcion que maneja los errores al guardar la temperatura
    manual
106     private fun showError(error: Int) {
107         if (error == 1) {
108             binding.temperatureEditText.error = "Introduzca una
    temperatura"
109         } else {
110             binding.temperatureEditText.error = "La temperatura debe
    ser múltiplo de 0.5"
111         }
112         binding.temperatureEditText.requestFocus()
113     }
114
115 }
```

```kotlin
package com.example.home_t.fragments.menu

import android.content.Intent
import android.os.Bundle
import android.view.Gravity
import android.view.LayoutInflater
import android.view.View
import android.view.ViewGroup
import android.widget.Toast
import androidx.fragment.app.Fragment
import com.example.home_t.*
import com.example.home_t.databinding.FragmentProgramModeBinding
import okhttp3.*
import java.io.IOException
import java.util.*

/*
    Fragmento ProgramMode: Fragmento del modo programa.
    Permite elegir dos temperaturas deseadas y acceder a la configuracion
    de los programas de los dias de la semana.
 */

class ProgramModeFragment : Fragment() {

    private var _binding: FragmentProgramModeBinding? = null
    private val binding get() = _binding!!

    // URL con la direccion IP del ESP
    private val ipUrl = "http://192.168.1.53/"

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setHasOptionsMenu(true)
    }

    override fun onCreateView(
        inflater: LayoutInflater,
        container: ViewGroup?,
        savedInstanceState: Bundle?
    ): View {
        _binding = FragmentProgramModeBinding.inflate(inflater, container, false)
        return binding.root
    }

    override fun onViewCreated(view: View, savedInstanceState: Bundle?) {
        setupProgramMode() // Funcion que controla la logica del fragmento ProgramMode
    }

    override fun onDestroyView() {
```

```
50          super.onDestroyView()
51          _binding = null
52      }
53
54      private fun setupProgramMode() {
55          // Se pulsa en guardar
56          binding.saveButton.setOnClickListener {
57
58              // Se comprueba que los campos de las temperaturas no
    esten vacios
59              if (binding.temp1EditText.text.isNotEmpty() && binding.
    temp2EditText.text.isNotEmpty()) {
60
61                  // Se pasan las temperaturas introducidas a string
62                  val stringTemp1 = binding.temp1EditText.text.toString
    ()
63                  val stringTemp2 = binding.temp2EditText.text.toString
    ()
64                  // Se pasan los string a double
65                  val temp1 = stringTemp1.toDouble()
66                  val temp2 = stringTemp2.toDouble()
67
68                  // Se comprueban que las temperaturas introducidas
    son multiplo de 0.5
69                  if ((temp1 % 0.5) == 0.0 && (temp2 % 0.5) == 0.0) {
70
71                      sendProgramTemps(temp1, temp2) // Se envian las
    temperaturas de programa al microcontrolador
72
73                      // Se avisa con un Toast que se han enviado las
    temperaturas
74                      val toast = Toast.makeText(context, "Temperaturas
     enviadas", Toast.LENGTH_SHORT)
75                      toast.setGravity(Gravity.CENTER, 0, 0)
76                      toast.show()
77
78                  } else {
79                      // Se comprueba cual es la temperatura que no es
    multiplo de 0.5
80                      if ((temp1 % 0.5) != 0.0) {
81                          binding.temp1EditText.error =
82                              "La temperatura debe ser multiplo de 0.5
    . Por ejemplo, 18.5, 19, etc."
83                          binding.temp1EditText.requestFocus()
84                      } else {
85                          binding.temp2EditText.error =
86                              "La temperatura debe ser multiplo de 0.5
    . Por ejemplo, 18.5, 19, etc."
87                          binding.temp2EditText.requestFocus()
88                      }
89                  }
90
91              } else {
```

```
92                // Se comprueba cual es el campo que esta vacio
93                if (binding.temp1EditText.text.isEmpty()) {
94
95                    binding.temp1EditText.error = "Debe introducir
    una temperatura 1"
96                    binding.temp1EditText.requestFocus()
97
98                } else {
99
100                   binding.temp2EditText.error = "Debe introducir
    una temperatura 2"
101                   binding.temp2EditText.requestFocus()
102
103               }
104
105           }
106       }
107
108       // Si se pulsa en algun dia para configurar su programa, se
    inicia la actividad pertinente
109       // Programa Lunes
110       binding.mondayButton.setOnClickListener {
111           startActivity(Intent(context, MondayProgramActivity::
    class.java))
112       }
113
114       // Programa Martes
115       binding.tuesdayButton.setOnClickListener {
116           startActivity(Intent(context, TuesdayProgramActivity::
    class.java))
117       }
118
119       // Programa Miercoles
120       binding.wednesdayButton.setOnClickListener {
121           startActivity(Intent(context, WednesdayProgramActivity::
    class.java))
122       }
123
124       // Programa Jueves
125       binding.thursdayButton.setOnClickListener {
126           startActivity(Intent(context, ThursdayProgramActivity::
    class.java))
127       }
128
129       // Programa Viernes
130       binding.fridayButton.setOnClickListener {
131           startActivity(Intent(context, FridayProgramActivity::
    class.java))
132       }
133
134       // Programa Sabado
135       binding.saturdayButton.setOnClickListener {
136           startActivity(Intent(context, SaturdayProgramActivity::
```

```
class.java))
        }

        // Programa Domingo
        binding.sundayButton.setOnClickListener {
            startActivity(Intent(context, SundayProgramActivity::
class.java))
        }
    }

    private fun sendProgramTemps(temp1: Double, temp2: Double) {
        // Para que el microcontrolador pueda leer correctamente las
dos temperaturas enviadas
        // hay que hacer que la temp1 siempre tenga 4 caracteres (3
digitos y el punto)
        val temp1String = String.format(Locale.ENGLISH, "%04.1f",
temp1) // Se pasa a formato ingles para representar el decimal con un
 punto

        val client = OkHttpClient() // Se crea el cliente OkHttp
        val url =
            ipUrl + "progtemps" + temp1String + temp2 // Se completa
la URL con un string identificativo del fragmento y las temperaturas
de programa
        // Se define la llamada HTTP
        val request = Request.Builder()
            .url(url)
            .addHeader("Connection", "close")
            .build()

        // Se realiza la llamada HTTP de forma asincrona
        client.newCall(request).enqueue(object : Callback {

            // Si ocurre algun problema en la llamada
            override fun onFailure(call: Call, e: IOException) {
                e.printStackTrace()
            }

            override fun onResponse(call: Call, response: Response) {
                // Como no tenemos nada que recibir, no hacemos nada
con la respuesta
            }
        })
    }
}
```

```kotlin
package com.example.home_t

import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import android.widget.Toast
import com.example.home_t.databinding.ActivityMondayProgramBinding
import okhttp3.*
import java.io.IOException

/*
    Actividad MondayProgram: Controla la configuracion del programa del dia Lunes.
    Permite elegir la temperatura deseada, entre las dos temperaturas de programa actuales,
    en cada hora del dia. Ademas, se puede ver el programa actual.
 */

class MondayProgramActivity : AppCompatActivity() {

    private lateinit var binding: ActivityMondayProgramBinding

    // URL con la direccion IP del ESP
    private val ipUrl = "http://192.168.1.53/"

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        binding = ActivityMondayProgramBinding.inflate(layoutInflater)
        setContentView(binding.root)

        setupMondayProgram() // Funcion que controla la logica de la actividad MondayProgram
    }

    private fun setupMondayProgram() {
        // Se pulsa en ver el programa actual
        binding.updateButton.setOnClickListener {
            getProgram() // Funcion que obtiene el programa actual del microcontrolador
        }

        // Se pulsa en guardar el programa seleccionado
        binding.saveButton.setOnClickListener {

            var mondayProgram = ""

            // Se añaden a la variable las 24 temperaturas seleccionadas para cada hora del dia
            // La primera temepratura corresponde a las 12 de la noche, la ultima a las 11 de la noche
            mondayProgram += if (binding.zeroButtonT1.isChecked) { "0" } else { "1" }
            mondayProgram += if (binding.oneButtonT1.isChecked) { "0" } else { "1" }
```

```
46              mondayProgram += if (binding.twoButtonT1.isChecked) { "0"
    } else { "1" }
47              mondayProgram += if (binding.threeButtonT1.isChecked) { "0
   " } else { "1" }
48              mondayProgram += if (binding.fourButtonT1.isChecked) { "0"
    } else { "1" }
49              mondayProgram += if (binding.fiveButtonT1.isChecked) { "0"
    } else { "1" }
50              mondayProgram += if (binding.sixButtonT1.isChecked) { "0"
    } else { "1" }
51              mondayProgram += if (binding.sevenButtonT1.isChecked) { "0
   " } else { "1" }
52              mondayProgram += if (binding.eightButtonT1.isChecked) { "0
   " } else { "1" }
53              mondayProgram += if (binding.nineButtonT1.isChecked) { "0"
    } else { "1" }
54              mondayProgram += if (binding.tenButtonT1.isChecked) { "0"
    } else { "1" }
55              mondayProgram += if (binding.elevenButtonT1.isChecked) { "
   0" } else { "1" }
56              mondayProgram += if (binding.twelveButtonT1.isChecked) { "
   0" } else { "1" }
57              mondayProgram += if (binding.thirteenButtonT1.isChecked
   ) { "0" } else { "1" }
58              mondayProgram += if (binding.fourteenButtonT1.isChecked
   ) { "0" } else { "1" }
59              mondayProgram += if (binding.fifteenButtonT1.isChecked) {
   "0" } else { "1" }
60              mondayProgram += if (binding.sixteenButtonT1.isChecked) {
   "0" } else { "1" }
61              mondayProgram += if (binding.seventeenButtonT1.isChecked
   ) { "0" } else { "1" }
62              mondayProgram += if (binding.eighteenButtonT1.isChecked
   ) { "0" } else { "1" }
63              mondayProgram += if (binding.nineteenButtonT1.isChecked
   ) { "0" } else { "1" }
64              mondayProgram += if (binding.twentyButtonT1.isChecked) { "
   0" } else { "1" }
65              mondayProgram += if (binding.twentyOneButtonT1.isChecked
   ) { "0" } else { "1" }
66              mondayProgram += if (binding.twentyTwoButtonT1.isChecked
   ) { "0" } else { "1" }
67              mondayProgram += if (binding.twentyThreeButtonT1.isChecked
   ) { "0" } else { "1" }
68
69              sendProgram(mondayProgram) // Funcion que manda el
   programa deseado al microcontrolador
70
71              Toast.makeText(this, "Programa enviado", Toast.
   LENGTH_SHORT).show()
72
73          }
74
```

```
75      }
76
77      private fun getProgram() {
78          val client = OkHttpClient() // Se crea el cliente OkHttp
79          val url = ipUrl + "getprograma" + "l" // Se completa la URL
    con un string identificativo de la accion a realizar y el dia del
    programa
80          // Se define la llamada HTTP
81          val request = Request.Builder()
82              .url(url)
83              .addHeader("Connection", "close")
84              .build()
85
86          // Se realiza la llamada HTTP de forma asincrona
87          client.newCall(request).enqueue(object : Callback {
88
89              // Si ocurre algun problema en la llamada
90              override fun onFailure(call: Call, e: IOException) {
91                  e.printStackTrace()
92              }
93
94              // Si se recibe una respuesta a la llamada
95              override fun onResponse(call: Call, response: Response) {
96                  response.use {
97                      if (response.isSuccessful) { // La respuesta se
    ha recibido correctamente
98                          val myResponse = response.body?.string() //
    Se guarda el cuerpo de la respuesta
99
100                         // Se filtra el cuerpo de la repuesta por
    digitos (se obtiene el programa actual)
101                         val program = myResponse?.filter(Char::
    isDigit)
102
103                         // Los cambios visuales en la pantalla se
    hacen en el hilo donde corre el interfaz de usuario
104                         runOnUiThread {
105                             if (myResponse != null) {
106                                 // Se actualiza la pantalla con las
    temperaturas actuales del programa
107                                 if (program?.get(0) == '1') binding.
    zeroButtonT2.isChecked = true
108                                 if (program?.get(1) == '1') binding.
    oneButtonT2.isChecked = true
109                                 if (program?.get(2) == '1') binding.
    twoButtonT2.isChecked = true
110                                 if (program?.get(3) == '1') binding.
    threeButtonT2.isChecked = true
111                                 if (program?.get(4) == '1') binding.
    fourButtonT2.isChecked = true
112                                 if (program?.get(5) == '1') binding.
    fiveButtonT2.isChecked = true
113                                 if (program?.get(6) == '1') binding.
```

```
113 sixButtonT2.isChecked = true
114                                 if (program?.get(7) == '1') binding.
    sevenButtonT2.isChecked = true
115                                 if (program?.get(8) == '1') binding.
    eightButtonT2.isChecked = true
116                                 if (program?.get(9) == '1') binding.
    nineButtonT2.isChecked = true
117                                 if (program?.get(10) == '1') binding.
    tenButtonT2.isChecked = true
118                                 if (program?.get(11) == '1') binding.
    elevenButtonT2.isChecked = true
119                                 if (program?.get(12) == '1') binding.
    twelveButtonT2.isChecked = true
120                                 if (program?.get(13) == '1') binding.
    thirteenButtonT2.isChecked = true
121                                 if (program?.get(14) == '1') binding.
    fourteenButtonT2.isChecked = true
122                                 if (program?.get(15) == '1') binding.
    fifteenButtonT2.isChecked = true
123                                 if (program?.get(16) == '1') binding.
    sixteenButtonT2.isChecked = true
124                                 if (program?.get(17) == '1') binding.
    seventeenButtonT2.isChecked = true
125                                 if (program?.get(18) == '1') binding.
    eighteenButtonT2.isChecked = true
126                                 if (program?.get(19) == '1') binding.
    nineteenButtonT2.isChecked = true
127                                 if (program?.get(20) == '1') binding.
    twentyButtonT2.isChecked = true
128                                 if (program?.get(21) == '1') binding.
    twentyOneButtonT2.isChecked = true
129                                 if (program?.get(22) == '1') binding.
    twentyTwoButtonT2.isChecked = true
130                                 if (program?.get(23) == '1') binding.
    twentyThreeButtonT2.isChecked = true
131
132                             }
133                         }
134                     }
135
136                 }
137             }
138         }
139         )
140     }
141
142     private fun sendProgram(program: String) {
143         val client = OkHttpClient() // Se crea el cliente OkHttp
144         val url = ipUrl + "progdia" + "l" + program // Se completa la
     URL con un string identificativo de la accion a realizar y el dia
    del programa
145         // Se define la llamada HTTP
146         val request = Request.Builder()
```

```
147             .url(url)
148             .addHeader("Connection", "close")
149             .build()
150 
151         // Se realiza la llamada HTTP de forma asincrona
152         client.newCall(request).enqueue(object : Callback {
153 
154             // Si ocurre algun problema en la llamada
155             override fun onFailure(call: Call, e: IOException) {
156                 e.printStackTrace()
157             }
158 
159             override fun onResponse(call: Call, response: Response) {
160                 // Como no tenemos nada que recibir, no hacemos nada
    con la respuesta
161             }
162 
163         })
164     }
165 }
```

```kotlin
package com.example.home_t

import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import android.widget.Toast
import com.example.home_t.databinding.ActivityTuesdayProgramBinding
import okhttp3.*
import java.io.IOException

/*
    Actividad TuesdayProgram: Controla la configuracion del programa del dia Martes.
    Permite elegir la temperatura deseada, entre las dos temperaturas de programa actuales,
    en cada hora del dia. Ademas, se puede ver el programa actual.
 */

class TuesdayProgramActivity : AppCompatActivity() {

    private lateinit var binding: ActivityTuesdayProgramBinding

    // URL con la direccion IP del ESP
    private val ipUrl = "http://192.168.1.53/"

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        binding = ActivityTuesdayProgramBinding.inflate(layoutInflater)
        setContentView(binding.root)

        setupTuesdayProgram() // Funcion que controla la logica de la actividad TuesdayProgram
    }

    private fun setupTuesdayProgram() {
        // Se pulsa en ver el programa actual
        binding.updateButton.setOnClickListener {
            getProgram() // Funcion que obtiene el programa actual del microcontrolador
        }

        // Se pulsa en guardar el programa seleccionado
        binding.saveButton.setOnClickListener {

            var tuesdayProgram = ""

            // Se añaden a la variable las 24 temperaturas seleccionadas para cada hora del dia
            // La primera temepratura corresponde a las 12 de la noche, la ultima a las 11 de la noche
            tuesdayProgram += if (binding.zeroButtonT1.isChecked) { "0" } else { "1" }
            tuesdayProgram += if (binding.oneButtonT1.isChecked) { "0"
```

```
45  } else { "1" }
46              tuesdayProgram += if (binding.twoButtonT1.isChecked) { "0"
    } else { "1" }
47              tuesdayProgram += if (binding.threeButtonT1.isChecked) { "
   0" } else { "1" }
48              tuesdayProgram += if (binding.fourButtonT1.isChecked) { "0
   " } else { "1" }
49              tuesdayProgram += if (binding.fiveButtonT1.isChecked) { "0
   " } else { "1" }
50              tuesdayProgram += if (binding.sixButtonT1.isChecked) { "0"
    } else { "1" }
51              tuesdayProgram += if (binding.sevenButtonT1.isChecked) { "
   0" } else { "1" }
52              tuesdayProgram += if (binding.eightButtonT1.isChecked) { "
   0" } else { "1" }
53              tuesdayProgram += if (binding.nineButtonT1.isChecked) { "0
   " } else { "1" }
54              tuesdayProgram += if (binding.tenButtonT1.isChecked) { "0"
    } else { "1" }
55              tuesdayProgram += if (binding.elevenButtonT1.isChecked) {
   "0" } else { "1" }
56              tuesdayProgram += if (binding.twelveButtonT1.isChecked) {
   "0" } else { "1" }
57              tuesdayProgram += if (binding.thirteenButtonT1.isChecked
   ) { "0" } else { "1" }
58              tuesdayProgram += if (binding.fourteenButtonT1.isChecked
   ) { "0" } else { "1" }
59              tuesdayProgram += if (binding.fifteenButtonT1.isChecked
   ) { "0" } else { "1" }
60              tuesdayProgram += if (binding.sixteenButtonT1.isChecked
   ) { "0" } else { "1" }
61              tuesdayProgram += if (binding.seventeenButtonT1.isChecked
   ) { "0" } else { "1" }
62              tuesdayProgram += if (binding.eighteenButtonT1.isChecked
   ) { "0" } else { "1" }
63              tuesdayProgram += if (binding.nineteenButtonT1.isChecked
   ) { "0" } else { "1" }
64              tuesdayProgram += if (binding.twentyButtonT1.isChecked) {
   "0" } else { "1" }
65              tuesdayProgram += if (binding.twentyOneButtonT1.isChecked
   ) { "0" } else { "1" }
66              tuesdayProgram += if (binding.twentyTwoButtonT1.isChecked
   ) { "0" } else { "1" }
67              tuesdayProgram += if (binding.twentyThreeButtonT1.
   isChecked) { "0" } else { "1" }
68
69              sendProgram(tuesdayProgram) // Funcion que manda el
   programa deseado al microcontrolador
70
71              Toast.makeText(this, "Programa guardado", Toast.
   LENGTH_SHORT).show()
72          }
73      }
```

```
74
75      private fun getProgram() {
76          val client = OkHttpClient() // Se crea el cliente OkHttp
77          val url = ipUrl + "getprograma" + "m" // Se completa la URL
    con un string identificativo de la accion a realizar y el dia del
    programa
78          // Se define la llamada HTTP
79          val request = Request.Builder()
80              .url(url)
81              .addHeader("Connection", "close")
82              .build()
83
84          // Se realiza la llamada HTTP de forma asincrona
85          client.newCall(request).enqueue(object : Callback {
86
87              // Si ocurre algun problema en la llamada
88              override fun onFailure(call: Call, e: IOException) {
89                  e.printStackTrace()
90              }
91
92              // Si se recibe una respuesta a la llamada
93              override fun onResponse(call: Call, response: Response) {
94                  response.use {
95                      if (response.isSuccessful) { // La respuesta se
    ha recibido correctamente
96                          val myResponse = response.body?.string() //
    Se guarda el cuerpo de la respuesta
97
98                          // Se filtra el cuerpo de la repuesta por
    digitos (se obtiene el programa actual)
99                          val program = myResponse?.filter(Char::
    isDigit)
100
101                         // Los cambios visuales en la pantalla se
    hacen en el hilo donde corre el interfaz de usuario
102                         runOnUiThread {
103                             if (myResponse != null) {
104                                 // Se actualiza la pantalla con las
    temperaturas actuales del programa
105                                 if (program?.get(0) == '1') binding.
    zeroButtonT2.isChecked = true
106                                 if (program?.get(1) == '1') binding.
    oneButtonT2.isChecked = true
107                                 if (program?.get(2) == '1') binding.
    twoButtonT2.isChecked = true
108                                 if (program?.get(3) == '1') binding.
    threeButtonT2.isChecked = true
109                                 if (program?.get(4) == '1') binding.
    fourButtonT2.isChecked = true
110                                 if (program?.get(5) == '1') binding.
    fiveButtonT2.isChecked = true
111                                 if (program?.get(6) == '1') binding.
    sixButtonT2.isChecked = true
```

```
                                    if (program?.get(7) == '1') binding.
    sevenButtonT2.isChecked = true
                                    if (program?.get(8) == '1') binding.
    eightButtonT2.isChecked = true
                                    if (program?.get(9) == '1') binding.
    nineButtonT2.isChecked = true
                                    if (program?.get(10) == '1') binding.
    tenButtonT2.isChecked = true
                                    if (program?.get(11) == '1') binding.
    elevenButtonT2.isChecked = true
                                    if (program?.get(12) == '1') binding.
    twelveButtonT2.isChecked = true
                                    if (program?.get(13) == '1') binding.
    thirteenButtonT2.isChecked = true
                                    if (program?.get(14) == '1') binding.
    fourteenButtonT2.isChecked = true
                                    if (program?.get(15) == '1') binding.
    fifteenButtonT2.isChecked = true
                                    if (program?.get(16) == '1') binding.
    sixteenButtonT2.isChecked = true
                                    if (program?.get(17) == '1') binding.
    seventeenButtonT2.isChecked = true
                                    if (program?.get(18) == '1') binding.
    eighteenButtonT2.isChecked = true
                                    if (program?.get(19) == '1') binding.
    nineteenButtonT2.isChecked = true
                                    if (program?.get(20) == '1') binding.
    twentyButtonT2.isChecked = true
                                    if (program?.get(21) == '1') binding.
    twentyOneButtonT2.isChecked = true
                                    if (program?.get(22) == '1') binding.
    twentyTwoButtonT2.isChecked = true
                                    if (program?.get(23) == '1') binding.
    twentyThreeButtonT2.isChecked = true

                                }
                            }
                        }

                    }
                }
            }
            )
        }

        private fun sendProgram(program: String) {
            val client = OkHttpClient() // Se crea el cliente OkHttp
            val url = ipUrl + "progdia" + "m" + program // Se completa la
     URL con un string identificativo de la accion a realizar y el dia
    del programa
            // Se define la llamada HTTP
            val request = Request.Builder()
                .url(url)
```

```
146             .addHeader("Connection", "close")
147             .build()
148 
149         // Se realiza la llamada HTTP de forma asincrona
150         client.newCall(request).enqueue(object : Callback {
151 
152             // Si ocurre algun problema en la llamada
153             override fun onFailure(call: Call, e: IOException) {
154                 e.printStackTrace()
155             }
156 
157             override fun onResponse(call: Call, response: Response) {
158                 // Como no tenemos nada que recibir, no hacemos nada
    con la respuesta
159             }
160 
161         })
162     }
163 }
```

```kotlin
package com.example.home_t

import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import android.widget.Toast
import com.example.home_t.databinding.ActivityWednesdayProgramBinding
import okhttp3.*
import java.io.IOException

/*
    Actividad WednesdayProgram: Controla la configuracion del programa del dia Miercoles.
    Permite elegir la temperatura deseada, entre las dos temperaturas de programa actuales,
    en cada hora del dia. Ademas, se puede ver el programa actual.
 */

class WednesdayProgramActivity : AppCompatActivity() {

    private lateinit var binding: ActivityWednesdayProgramBinding

    // URL con la direccion IP del ESP
    private val ipUrl = "http://192.168.1.53/"

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        binding = ActivityWednesdayProgramBinding.inflate(layoutInflater)
        setContentView(binding.root)

        setupWednesdayProgram() // Funcion que controla la logica de la actividad WednesdayProgram
    }

    private fun setupWednesdayProgram() {
        // Se pulsa en ver el programa actual
        binding.updateButton.setOnClickListener {
            getProgram() // Funcion que obtiene el programa actual del microcontrolador
        }

        // Se pulsa en guardar el programa seleccionado
        binding.saveButton.setOnClickListener {

            var wednesdayProgram = ""

            // Se añaden a la variable las 24 temperaturas seleccionadas para cada hora del dia
            // La primera temepratura corresponde a las 12 de la noche, la ultima a las 11 de la noche
            wednesdayProgram += if (binding.zeroButtonT1.isChecked) { "0" } else { "1" }
            wednesdayProgram += if (binding.oneButtonT1.isChecked) { "
```

```
45 0" } else { "1" }
46             wednesdayProgram += if (binding.twoButtonT1.isChecked) { "
   0" } else { "1" }
47             wednesdayProgram += if (binding.threeButtonT1.isChecked
   ) { "0" } else { "1" }
48             wednesdayProgram += if (binding.fourButtonT1.isChecked) {
   "0" } else { "1" }
49             wednesdayProgram += if (binding.fiveButtonT1.isChecked) {
   "0" } else { "1" }
50             wednesdayProgram += if (binding.sixButtonT1.isChecked) { "
   0" } else { "1" }
51             wednesdayProgram += if (binding.sevenButtonT1.isChecked
   ) { "0" } else { "1" }
52             wednesdayProgram += if (binding.eightButtonT1.isChecked
   ) { "0" } else { "1" }
53             wednesdayProgram += if (binding.nineButtonT1.isChecked) {
   "0" } else { "1" }
54             wednesdayProgram += if (binding.tenButtonT1.isChecked) { "
   0" } else { "1" }
55             wednesdayProgram += if (binding.elevenButtonT1.isChecked
   ) { "0" } else { "1" }
56             wednesdayProgram += if (binding.twelveButtonT1.isChecked
   ) { "0" } else { "1" }
57             wednesdayProgram += if (binding.thirteenButtonT1.isChecked
   ) { "0" } else { "1" }
58             wednesdayProgram += if (binding.fourteenButtonT1.isChecked
   ) { "0" } else { "1" }
59             wednesdayProgram += if (binding.fifteenButtonT1.isChecked
   ) { "0" } else { "1" }
60             wednesdayProgram += if (binding.sixteenButtonT1.isChecked
   ) { "0" } else { "1" }
61             wednesdayProgram += if (binding.seventeenButtonT1.
   isChecked) { "0" } else { "1" }
62             wednesdayProgram += if (binding.eighteenButtonT1.isChecked
   ) { "0" } else { "1" }
63             wednesdayProgram += if (binding.nineteenButtonT1.isChecked
   ) { "0" } else { "1" }
64             wednesdayProgram += if (binding.twentyButtonT1.isChecked
   ) { "0" } else { "1" }
65             wednesdayProgram += if (binding.twentyOneButtonT1.
   isChecked) { "0" } else { "1" }
66             wednesdayProgram += if (binding.twentyTwoButtonT1.
   isChecked) { "0" } else { "1" }
67             wednesdayProgram += if (binding.twentyThreeButtonT1.
   isChecked) { "0" } else { "1" }
68
69             sendProgram(wednesdayProgram) // Funcion que manda el
   programa deseado al microcontrolador
70
71             Toast.makeText(this, "Programa guardado", Toast.
   LENGTH_SHORT).show()
72         }
73
```

```
74      }
75
76      private fun getProgram() {
77          val client = OkHttpClient() // Se crea el cliente OkHttp
78          val url = ipUrl + "getprograma" + "x" // Se completa la URL
    con un string identificativo de la accion a realizar y el dia del
    programa
79          // Se define la llamada HTTP
80          val request = Request.Builder()
81              .url(url)
82              .addHeader("Connection", "close")
83              .build()
84
85          // Se realiza la llamada HTTP de forma asincrona
86          client.newCall(request).enqueue(object : Callback {
87
88              // Si ocurre algun problema en la llamada
89              override fun onFailure(call: Call, e: IOException) {
90                  e.printStackTrace()
91              }
92
93              // Si se recibe una respuesta a la llamada
94              override fun onResponse(call: Call, response: Response) {
95                  response.use {
96                      if (response.isSuccessful) { // La respuesta se
    ha recibido correctamente
97                          val myResponse = response.body?.string() //
    Se guarda el cuerpo de la respuesta
98
99                          // Se filtra el cuerpo de la repuesta por
    digitos (se obtiene el programa actual)
100                         val program = myResponse?.filter(Char::
    isDigit)
101
102                         // Los cambios visuales en la pantalla se
    hacen en el hilo donde corre el interfaz de usuario
103                         runOnUiThread {
104                             if (myResponse != null) {
105                                 // Se actualiza la pantalla con las
    temperaturas actuales del programa
106                                 if (program?.get(0) == '1') binding.
    zeroButtonT2.isChecked = true
107                                 if (program?.get(1) == '1') binding.
    oneButtonT2.isChecked = true
108                                 if (program?.get(2) == '1') binding.
    twoButtonT2.isChecked = true
109                                 if (program?.get(3) == '1') binding.
    threeButtonT2.isChecked = true
110                                 if (program?.get(4) == '1') binding.
    fourButtonT2.isChecked = true
111                                 if (program?.get(5) == '1') binding.
    fiveButtonT2.isChecked = true
112                                 if (program?.get(6) == '1') binding.
```

```
112 sixButtonT2.isChecked = true
113                                 if (program?.get(7) == '1') binding.
    sevenButtonT2.isChecked = true
114                                 if (program?.get(8) == '1') binding.
    eightButtonT2.isChecked = true
115                                 if (program?.get(9) == '1') binding.
    nineButtonT2.isChecked = true
116                                 if (program?.get(10) == '1') binding.
    tenButtonT2.isChecked = true
117                                 if (program?.get(11) == '1') binding.
    elevenButtonT2.isChecked = true
118                                 if (program?.get(12) == '1') binding.
    twelveButtonT2.isChecked = true
119                                 if (program?.get(13) == '1') binding.
    thirteenButtonT2.isChecked = true
120                                 if (program?.get(14) == '1') binding.
    fourteenButtonT2.isChecked = true
121                                 if (program?.get(15) == '1') binding.
    fifteenButtonT2.isChecked = true
122                                 if (program?.get(16) == '1') binding.
    sixteenButtonT2.isChecked = true
123                                 if (program?.get(17) == '1') binding.
    seventeenButtonT2.isChecked = true
124                                 if (program?.get(18) == '1') binding.
    eighteenButtonT2.isChecked = true
125                                 if (program?.get(19) == '1') binding.
    nineteenButtonT2.isChecked = true
126                                 if (program?.get(20) == '1') binding.
    twentyButtonT2.isChecked = true
127                                 if (program?.get(21) == '1') binding.
    twentyOneButtonT2.isChecked = true
128                                 if (program?.get(22) == '1') binding.
    twentyTwoButtonT2.isChecked = true
129                                 if (program?.get(23) == '1') binding.
    twentyThreeButtonT2.isChecked = true
130
131                             }
132                         }
133                     }
134
135                 }
136             }
137         }
138         )
139     }
140
141     private fun sendProgram(program: String) {
142         val client = OkHttpClient() // Se crea el cliente OkHttp
143         val url = ipUrl + "progdia" + "x" + program // Se completa la
     URL con un string identificativo de la accion a realizar y el dia
    del programa
144         // Se define la llamada HTTP
145         val request = Request.Builder()
```

```
146             .url(url)
147             .addHeader("Connection", "close")
148             .build()
149 
150         // Se realiza la llamada HTTP de forma asincrona
151         client.newCall(request).enqueue(object : Callback {
152 
153             // Si ocurre algun problema en la llamada
154             override fun onFailure(call: Call, e: IOException) {
155                 e.printStackTrace()
156             }
157 
158             override fun onResponse(call: Call, response: Response) {
159                 // Como no tenemos nada que recibir, no hacemos nada
    con la respuesta
160             }
161 
162         })
163     }
164 }
165 
```

```kotlin
package com.example.home_t

import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import android.widget.Toast
import com.example.home_t.databinding.ActivityThursdayProgramBinding
import okhttp3.*
import java.io.IOException

/*
    Actividad ThursdayProgram: Controla la configuracion del programa del dia Jueves.
    Permite elegir la temperatura deseada, entre las dos temperaturas de programa actuales,
    en cada hora del dia. Ademas, se puede ver el programa actual.
 */

class ThursdayProgramActivity : AppCompatActivity() {

    private lateinit var binding: ActivityThursdayProgramBinding

    // URL con la direccion IP del ESP
    private val ipUrl = "http://192.168.1.53/"

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        binding = ActivityThursdayProgramBinding.inflate(layoutInflater)
        setContentView(binding.root)

        setupThursdayProgram() // Funcion que controla la logica de la actividad ThursdayProgram
    }

    private fun setupThursdayProgram() {
        // Se pulsa en ver el programa actual
        binding.updateButton.setOnClickListener {
            getProgram() // Funcion que obtiene el programa actual del microcontrolador
        }

        // Se pulsa en guardar el programa seleccionado
        binding.saveButton.setOnClickListener {

            var thursdayProgram = ""

            // Se añaden a la variable las 24 temperaturas seleccionadas para cada hora del dia
            // La primera temepratura corresponde a las 12 de la noche, la ultima a las 11 de la noche
            thursdayProgram += if (binding.zeroButtonT1.isChecked) { "0" } else { "1" }
            thursdayProgram += if (binding.oneButtonT1.isChecked) { "0
```

```
45 " } else { "1" }
46             thursdayProgram += if (binding.twoButtonT1.isChecked) { "0
   " } else { "1" }
47             thursdayProgram += if (binding.threeButtonT1.isChecked) {
   "0" } else { "1" }
48             thursdayProgram += if (binding.fourButtonT1.isChecked) { "
   0" } else { "1" }
49             thursdayProgram += if (binding.fiveButtonT1.isChecked) { "
   0" } else { "1" }
50             thursdayProgram += if (binding.sixButtonT1.isChecked) { "0
   " } else { "1" }
51             thursdayProgram += if (binding.sevenButtonT1.isChecked) {
   "0" } else { "1" }
52             thursdayProgram += if (binding.eightButtonT1.isChecked) {
   "0" } else { "1" }
53             thursdayProgram += if (binding.nineButtonT1.isChecked) { "
   0" } else { "1" }
54             thursdayProgram += if (binding.tenButtonT1.isChecked) { "0
   " } else { "1" }
55             thursdayProgram += if (binding.elevenButtonT1.isChecked
   ) { "0" } else { "1" }
56             thursdayProgram += if (binding.twelveButtonT1.isChecked
   ) { "0" } else { "1" }
57             thursdayProgram += if (binding.thirteenButtonT1.isChecked
   ) { "0" } else { "1" }
58             thursdayProgram += if (binding.fourteenButtonT1.isChecked
   ) { "0" } else { "1" }
59             thursdayProgram += if (binding.fifteenButtonT1.isChecked
   ) { "0" } else { "1" }
60             thursdayProgram += if (binding.sixteenButtonT1.isChecked
   ) { "0" } else { "1" }
61             thursdayProgram += if (binding.seventeenButtonT1.isChecked
   ) { "0" } else { "1" }
62             thursdayProgram += if (binding.eighteenButtonT1.isChecked
   ) { "0" } else { "1" }
63             thursdayProgram += if (binding.nineteenButtonT1.isChecked
   ) { "0" } else { "1" }
64             thursdayProgram += if (binding.twentyButtonT1.isChecked
   ) { "0" } else { "1" }
65             thursdayProgram += if (binding.twentyOneButtonT1.isChecked
   ) { "0" } else { "1" }
66             thursdayProgram += if (binding.twentyTwoButtonT1.isChecked
   ) { "0" } else { "1" }
67             thursdayProgram += if (binding.twentyThreeButtonT1.
   isChecked) { "0" } else { "1" }
68
69             sendProgram(thursdayProgram) // Funcion que manda el
   programa deseado al microcontrolador
70
71             Toast.makeText(this, "Programa guardado", Toast.
   LENGTH_SHORT).show()
72
73         }
```

```
74
75      }
76
77      private fun getProgram() {
78          val client = OkHttpClient() // Se crea el cliente OkHttp
79          val url = ipUrl + "getprograma" + "j" // Se completa la URL
    con un string identificativo de la accion a realizar y el dia del
    programa
80          // Se define la llamada HTTP
81          val request = Request.Builder()
82              .url(url)
83              .addHeader("Connection", "close")
84              .build()
85
86          // Se realiza la llamada HTTP de forma asincrona
87          client.newCall(request).enqueue(object : Callback {
88
89              // Si ocurre algun problema en la llamada
90              override fun onFailure(call: Call, e: IOException) {
91                  e.printStackTrace()
92              }
93
94              // Si se recibe una respuesta a la llamada
95              override fun onResponse(call: Call, response: Response) {
96                  response.use {
97                      if (response.isSuccessful) { // La respuesta se
    ha recibido correctamente
98                          val myResponse = response.body?.string() //
    Se guarda el cuerpo de la respuesta
99
100                         // Se filtra el cuerpo de la repuesta por
    digitos (se obtiene el programa actual)
101                         val program = myResponse?.filter(Char::
    isDigit)
102
103                         // Los cambios visuales en la pantalla se
    hacen en el hilo donde corre el interfaz de usuario
104                         runOnUiThread {
105                             if (myResponse != null) {
106                                 // Se actualiza la pantalla con las
    temperaturas actuales del programa
107                                 if (program?.get(0) == '1') binding.
    zeroButtonT2.isChecked = true
108                                 if (program?.get(1) == '1') binding.
    oneButtonT2.isChecked = true
109                                 if (program?.get(2) == '1') binding.
    twoButtonT2.isChecked = true
110                                 if (program?.get(3) == '1') binding.
    threeButtonT2.isChecked = true
111                                 if (program?.get(4) == '1') binding.
    fourButtonT2.isChecked = true
112                                 if (program?.get(5) == '1') binding.
    fiveButtonT2.isChecked = true
```

```
                                    if (program?.get(6) == '1') binding.
    sixButtonT2.isChecked = true
                                    if (program?.get(7) == '1') binding.
    sevenButtonT2.isChecked = true
                                    if (program?.get(8) == '1') binding.
    eightButtonT2.isChecked = true
                                    if (program?.get(9) == '1') binding.
    nineButtonT2.isChecked = true
                                    if (program?.get(10) == '1') binding.
    tenButtonT2.isChecked = true
                                    if (program?.get(11) == '1') binding.
    elevenButtonT2.isChecked = true
                                    if (program?.get(12) == '1') binding.
    twelveButtonT2.isChecked = true
                                    if (program?.get(13) == '1') binding.
    thirteenButtonT2.isChecked = true
                                    if (program?.get(14) == '1') binding.
    fourteenButtonT2.isChecked = true
                                    if (program?.get(15) == '1') binding.
    fifteenButtonT2.isChecked = true
                                    if (program?.get(16) == '1') binding.
    sixteenButtonT2.isChecked = true
                                    if (program?.get(17) == '1') binding.
    seventeenButtonT2.isChecked = true
                                    if (program?.get(18) == '1') binding.
    eighteenButtonT2.isChecked = true
                                    if (program?.get(19) == '1') binding.
    nineteenButtonT2.isChecked = true
                                    if (program?.get(20) == '1') binding.
    twentyButtonT2.isChecked = true
                                    if (program?.get(21) == '1') binding.
    twentyOneButtonT2.isChecked = true
                                    if (program?.get(22) == '1') binding.
    twentyTwoButtonT2.isChecked = true
                                    if (program?.get(23) == '1') binding.
    twentyThreeButtonT2.isChecked = true

                                }
                            }
                        }

                    }
                }
            }
            )
        }

        private fun sendProgram(program: String) {
            val client = OkHttpClient() // Se crea el cliente OkHttp
            val url = ipUrl + "progdia" + "j" + program // Se completa la
     URL con un string identificativo de la accion a realizar y el dia
    del programa
            // Se define la llamada HTTP
```

```
146         val request = Request.Builder()
147             .url(url)
148             .addHeader("Connection", "close")
149             .build()
150 
151         // Se realiza la llamada HTTP de forma asincrona
152         client.newCall(request).enqueue(object : Callback {
153 
154             // Si ocurre algun problema en la llamada
155             override fun onFailure(call: Call, e: IOException) {
156                 e.printStackTrace()
157             }
158 
159             override fun onResponse(call: Call, response: Response) {
160                 // Como no tenemos nada que recibir, no hacemos nada
    con la respuesta
161             }
162 
163         })
164     }
165 }
```

```kotlin
package com.example.home_t

import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import android.widget.Toast
import com.example.home_t.databinding.ActivityFridayProgramBinding
import okhttp3.*
import java.io.IOException

/*
    Actividad FridayProgram: Controla la configuracion del programa del dia Viernes.
    Permite elegir la temperatura deseada, entre las dos temperaturas de programa actuales,
    en cada hora del dia. Ademas, se puede ver el programa actual.
 */

class FridayProgramActivity : AppCompatActivity() {

    private lateinit var binding: ActivityFridayProgramBinding

    // URL con la direccion IP del ESP
    private val ipUrl = "http://192.168.1.53/"

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        binding = ActivityFridayProgramBinding.inflate(layoutInflater)
        setContentView(binding.root)

        setupFridayProgram() // Funcion que controla la logica de la actividad FridayProgram
    }

    private fun setupFridayProgram() {
        // Se pulsa en ver el programa actual
        binding.updateButton.setOnClickListener {
            getProgram() // Funcion que obtiene el programa actual del microcontrolador
        }

        // Se pulsa en guardar el programa seleccionado
        binding.saveButton.setOnClickListener {

            var fridayProgram = ""

            // Se añaden a la variable las 24 temperaturas seleccionadas para cada hora del dia
            // La primera temepratura corresponde a las 12 de la noche, la ultima a las 11 de la noche
            fridayProgram += if (binding.zeroButtonT1.isChecked) { "0" } else { "1" }
            fridayProgram += if (binding.oneButtonT1.isChecked) { "0" } else { "1" }
```

```
46              fridayProgram += if (binding.twoButtonT1.isChecked) { "0"
    } else { "1" }
47              fridayProgram += if (binding.threeButtonT1.isChecked) { "0
   " } else { "1" }
48              fridayProgram += if (binding.fourButtonT1.isChecked) { "0"
    } else { "1" }
49              fridayProgram += if (binding.fiveButtonT1.isChecked) { "0"
    } else { "1" }
50              fridayProgram += if (binding.sixButtonT1.isChecked) { "0"
    } else { "1" }
51              fridayProgram += if (binding.sevenButtonT1.isChecked) { "0
   " } else { "1" }
52              fridayProgram += if (binding.eightButtonT1.isChecked) { "0
   " } else { "1" }
53              fridayProgram += if (binding.nineButtonT1.isChecked) { "0"
    } else { "1" }
54              fridayProgram += if (binding.tenButtonT1.isChecked) { "0"
    } else { "1" }
55              fridayProgram += if (binding.elevenButtonT1.isChecked) { "
   0" } else { "1" }
56              fridayProgram += if (binding.twelveButtonT1.isChecked) { "
   0" } else { "1" }
57              fridayProgram += if (binding.thirteenButtonT1.isChecked
   ) { "0" } else { "1" }
58              fridayProgram += if (binding.fourteenButtonT1.isChecked
   ) { "0" } else { "1" }
59              fridayProgram += if (binding.fifteenButtonT1.isChecked) {
   "0" } else { "1" }
60              fridayProgram += if (binding.sixteenButtonT1.isChecked) {
   "0" } else { "1" }
61              fridayProgram += if (binding.seventeenButtonT1.isChecked
   ) { "0" } else { "1" }
62              fridayProgram += if (binding.eighteenButtonT1.isChecked
   ) { "0" } else { "1" }
63              fridayProgram += if (binding.nineteenButtonT1.isChecked
   ) { "0" } else { "1" }
64              fridayProgram += if (binding.twentyButtonT1.isChecked) { "
   0" } else { "1" }
65              fridayProgram += if (binding.twentyOneButtonT1.isChecked
   ) { "0" } else { "1" }
66              fridayProgram += if (binding.twentyTwoButtonT1.isChecked
   ) { "0" } else { "1" }
67              fridayProgram += if (binding.twentyThreeButtonT1.isChecked
   ) { "0" } else { "1" }
68
69              sendProgram(fridayProgram) // Funcion que manda el
   programa deseado al microcontrolador
70
71              Toast.makeText(this, "Programa guardado", Toast.
   LENGTH_SHORT).show()
72          }
73
74      }
```

```
75
76      private fun getProgram() {
77          val client = OkHttpClient() // Se crea el cliente OkHttp
78          val url = ipUrl + "getprograma" + "v" // Se completa la URL
    con un string identificativo de la accion a realizar y el dia del
    programa
79          // Se define la llamada HTTP
80          val request = Request.Builder()
81              .url(url)
82              .addHeader("Connection", "close")
83              .build()
84
85          // Se realiza la llamada HTTP de forma asincrona
86          client.newCall(request).enqueue(object : Callback {
87
88              // Si ocurre algun problema en la llamada
89              override fun onFailure(call: Call, e: IOException) {
90                  e.printStackTrace()
91              }
92
93              // Si se recibe una respuesta a la llamada
94              override fun onResponse(call: Call, response: Response) {
95                  response.use {
96                      if (response.isSuccessful) { // La respuesta se
    ha recibido correctamente
97                          val myResponse = response.body?.string() //
    Se guarda el cuerpo de la respuesta
98
99                          // Se filtra el cuerpo de la repuesta por
    digitos (se obtiene el programa actual)
100                         val program = myResponse?.filter(Char::
    isDigit)
101
102                         // Los cambios visuales en la pantalla se
    hacen en el hilo donde corre el interfaz de usuario
103                         runOnUiThread {
104                             if (myResponse != null) {
105                                 // Se actualiza la pantalla con las
    temperaturas actuales del programa
106                                 if (program?.get(0) == '1') binding.
    zeroButtonT2.isChecked = true
107                                 if (program?.get(1) == '1') binding.
    oneButtonT2.isChecked = true
108                                 if (program?.get(2) == '1') binding.
    twoButtonT2.isChecked = true
109                                 if (program?.get(3) == '1') binding.
    threeButtonT2.isChecked = true
110                                 if (program?.get(4) == '1') binding.
    fourButtonT2.isChecked = true
111                                 if (program?.get(5) == '1') binding.
    fiveButtonT2.isChecked = true
112                                 if (program?.get(6) == '1') binding.
    sixButtonT2.isChecked = true
```

```
113                                 if (program?.get(7) == '1') binding.
    sevenButtonT2.isChecked = true
114                                 if (program?.get(8) == '1') binding.
    eightButtonT2.isChecked = true
115                                 if (program?.get(9) == '1') binding.
    nineButtonT2.isChecked = true
116                                 if (program?.get(10) == '1') binding.
    tenButtonT2.isChecked = true
117                                 if (program?.get(11) == '1') binding.
    elevenButtonT2.isChecked = true
118                                 if (program?.get(12) == '1') binding.
    twelveButtonT2.isChecked = true
119                                 if (program?.get(13) == '1') binding.
    thirteenButtonT2.isChecked = true
120                                 if (program?.get(14) == '1') binding.
    fourteenButtonT2.isChecked = true
121                                 if (program?.get(15) == '1') binding.
    fifteenButtonT2.isChecked = true
122                                 if (program?.get(16) == '1') binding.
    sixteenButtonT2.isChecked = true
123                                 if (program?.get(17) == '1') binding.
    seventeenButtonT2.isChecked = true
124                                 if (program?.get(18) == '1') binding.
    eighteenButtonT2.isChecked = true
125                                 if (program?.get(19) == '1') binding.
    nineteenButtonT2.isChecked = true
126                                 if (program?.get(20) == '1') binding.
    twentyButtonT2.isChecked = true
127                                 if (program?.get(21) == '1') binding.
    twentyOneButtonT2.isChecked = true
128                                 if (program?.get(22) == '1') binding.
    twentyTwoButtonT2.isChecked = true
129                                 if (program?.get(23) == '1') binding.
    twentyThreeButtonT2.isChecked = true
130
131                             }
132                         }
133                     }
134
135                 }
136             }
137         }
138         )
139     }
140
141     private fun sendProgram(program: String) {
142         val client = OkHttpClient() // Se crea el cliente OkHttp
143         val url = ipUrl + "progdia" + "v" + program // Se completa la
     URL con un string identificativo de la accion a realizar y el dia
    del programa
144         // Se define la llamada HTTP
145         val request = Request.Builder()
146             .url(url)
```

```
147             .addHeader("Connection", "close")
148             .build()
149 
150         // Se realiza la llamada HTTP de forma asincrona
151         client.newCall(request).enqueue(object : Callback {
152 
153             // Si ocurre algun problema en la llamada
154             override fun onFailure(call: Call, e: IOException) {
155                 e.printStackTrace()
156             }
157 
158             override fun onResponse(call: Call, response: Response) {
159                 // Como no tenemos nada que recibir, no hacemos nada
    con la respuesta
160             }
161 
162         })
163     }
164 }
```

```kotlin
package com.example.home_t

import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import android.widget.Toast
import com.example.home_t.databinding.ActivitySaturdayProgramBinding
import okhttp3.*
import java.io.IOException

/*
    Actividad SaturdayProgram: Controla la configuracion del programa del dia Sabado.
    Permite elegir la temperatura deseada, entre las dos temperaturas de programa actuales,
    en cada hora del dia. Ademas, se puede ver el programa actual.
 */

class SaturdayProgramActivity : AppCompatActivity() {

    private lateinit var binding: ActivitySaturdayProgramBinding

    // URL con la direccion IP del ESP
    private val ipUrl = "http://192.168.1.53/"

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        binding = ActivitySaturdayProgramBinding.inflate(layoutInflater)
        setContentView(binding.root)

        setupSaturdayProgram() // Funcion que controla la logica de la actividad SaturdayProgram
    }

    private fun setupSaturdayProgram() {
        // Se pulsa en ver el programa actual
        binding.updateButton.setOnClickListener {
            getProgram() // Funcion que obtiene el programa actual del microcontrolador
        }

        // Se pulsa en guardar el programa seleccionado
        binding.saveButton.setOnClickListener {

            var saturdayProgram = ""

            // Se añaden a la variable las 24 temperaturas seleccionadas para cada hora del dia
            // La primera temepratura corresponde a las 12 de la noche, la ultima a las 11 de la noche
            saturdayProgram += if (binding.zeroButtonT1.isChecked) { "0" } else { "1" }
            saturdayProgram += if (binding.oneButtonT1.isChecked) { "0
```

```
45 " } else { "1" }
46             saturdayProgram += if (binding.twoButtonT1.isChecked) { "0
   " } else { "1" }
47             saturdayProgram += if (binding.threeButtonT1.isChecked) {
   "0" } else { "1" }
48             saturdayProgram += if (binding.fourButtonT1.isChecked) { "
   0" } else { "1" }
49             saturdayProgram += if (binding.fiveButtonT1.isChecked) { "
   0" } else { "1" }
50             saturdayProgram += if (binding.sixButtonT1.isChecked) { "0
   " } else { "1" }
51             saturdayProgram += if (binding.sevenButtonT1.isChecked) {
   "0" } else { "1" }
52             saturdayProgram += if (binding.eightButtonT1.isChecked) {
   "0" } else { "1" }
53             saturdayProgram += if (binding.nineButtonT1.isChecked) { "
   0" } else { "1" }
54             saturdayProgram += if (binding.tenButtonT1.isChecked) { "0
   " } else { "1" }
55             saturdayProgram += if (binding.elevenButtonT1.isChecked
   ) { "0" } else { "1" }
56             saturdayProgram += if (binding.twelveButtonT1.isChecked
   ) { "0" } else { "1" }
57             saturdayProgram += if (binding.thirteenButtonT1.isChecked
   ) { "0" } else { "1" }
58             saturdayProgram += if (binding.fourteenButtonT1.isChecked
   ) { "0" } else { "1" }
59             saturdayProgram += if (binding.fifteenButtonT1.isChecked
   ) { "0" } else { "1" }
60             saturdayProgram += if (binding.sixteenButtonT1.isChecked
   ) { "0" } else { "1" }
61             saturdayProgram += if (binding.seventeenButtonT1.isChecked
   ) { "0" } else { "1" }
62             saturdayProgram += if (binding.eighteenButtonT1.isChecked
   ) { "0" } else { "1" }
63             saturdayProgram += if (binding.nineteenButtonT1.isChecked
   ) { "0" } else { "1" }
64             saturdayProgram += if (binding.twentyButtonT1.isChecked
   ) { "0" } else { "1" }
65             saturdayProgram += if (binding.twentyOneButtonT1.isChecked
   ) { "0" } else { "1" }
66             saturdayProgram += if (binding.twentyTwoButtonT1.isChecked
   ) { "0" } else { "1" }
67             saturdayProgram += if (binding.twentyThreeButtonT1.
   isChecked) { "0" } else { "1" }
68
69             sendProgram(saturdayProgram) // Funcion que manda el
   programa deseado al microcontrolador
70
71             Toast.makeText(this, "Programa guardado", Toast.
   LENGTH_SHORT).show()
72
73         }
```

```
74
75      }
76
77      private fun getProgram() {
78          val client = OkHttpClient() // Se crea el cliente OkHttp
79          val url = ipUrl + "getprograma" + "s" // Se completa la URL
    con un string identificativo de la accion a realizar y el dia del
    programa
80          // Se define la llamada HTTP
81          val request = Request.Builder()
82              .url(url)
83              .addHeader("Connection", "close")
84              .build()
85
86          // Se realiza la llamada HTTP de forma asincrona
87          client.newCall(request).enqueue(object : Callback {
88
89              // Si ocurre algun problema en la llamada
90              override fun onFailure(call: Call, e: IOException) {
91                  e.printStackTrace()
92              }
93
94              // Si se recibe una respuesta a la llamada
95              override fun onResponse(call: Call, response: Response) {
96                  response.use {
97                      if (response.isSuccessful) { // La respuesta se
    ha recibido correctamente
98                          val myResponse = response.body?.string() //
    Se guarda el cuerpo de la respuesta
99
100                         // Se filtra el cuerpo de la repuesta por
    digitos (se obtiene el programa actual)
101                         val program = myResponse?.filter(Char::
    isDigit)
102
103                         // Los cambios visuales en la pantalla se
    hacen en el hilo donde corre el interfaz de usuario
104                         runOnUiThread {
105                             if (myResponse != null) {
106                                 // Se actualiza la pantalla con las
    temperaturas actuales del programa
107                                 if (program?.get(0) == '1') binding.
    zeroButtonT2.isChecked = true
108                                 if (program?.get(1) == '1') binding.
    oneButtonT2.isChecked = true
109                                 if (program?.get(2) == '1') binding.
    twoButtonT2.isChecked = true
110                                 if (program?.get(3) == '1') binding.
    threeButtonT2.isChecked = true
111                                 if (program?.get(4) == '1') binding.
    fourButtonT2.isChecked = true
112                                 if (program?.get(5) == '1') binding.
    fiveButtonT2.isChecked = true
```

```
                                if (program?.get(6) == '1') binding.
    sixButtonT2.isChecked = true
                                if (program?.get(7) == '1') binding.
    sevenButtonT2.isChecked = true
                                if (program?.get(8) == '1') binding.
    eightButtonT2.isChecked = true
                                if (program?.get(9) == '1') binding.
    nineButtonT2.isChecked = true
                                if (program?.get(10) == '1') binding.
    tenButtonT2.isChecked = true
                                if (program?.get(11) == '1') binding.
    elevenButtonT2.isChecked = true
                                if (program?.get(12) == '1') binding.
    twelveButtonT2.isChecked = true
                                if (program?.get(13) == '1') binding.
    thirteenButtonT2.isChecked = true
                                if (program?.get(14) == '1') binding.
    fourteenButtonT2.isChecked = true
                                if (program?.get(15) == '1') binding.
    fifteenButtonT2.isChecked = true
                                if (program?.get(16) == '1') binding.
    sixteenButtonT2.isChecked = true
                                if (program?.get(17) == '1') binding.
    seventeenButtonT2.isChecked = true
                                if (program?.get(18) == '1') binding.
    eighteenButtonT2.isChecked = true
                                if (program?.get(19) == '1') binding.
    nineteenButtonT2.isChecked = true
                                if (program?.get(20) == '1') binding.
    twentyButtonT2.isChecked = true
                                if (program?.get(21) == '1') binding.
    twentyOneButtonT2.isChecked = true
                                if (program?.get(22) == '1') binding.
    twentyTwoButtonT2.isChecked = true
                                if (program?.get(23) == '1') binding.
    twentyThreeButtonT2.isChecked = true

                            }
                        }
                    }

                }
            }
        }
        )
    }

    private fun sendProgram(program: String) {
        val client = OkHttpClient() // Se crea el cliente OkHttp
        val url = ipUrl + "progdia" + "s" + program // Se completa la
     URL con un string identificativo de la accion a realizar y el dia
    del programa
        // Se define la llamada HTTP
```

```
146         val request = Request.Builder()
147             .url(url)
148             .addHeader("Connection", "close")
149             .build()
150 
151         // Se realiza la llamada HTTP de forma asincrona
152         client.newCall(request).enqueue(object : Callback {
153 
154             // Si ocurre algun problema en la llamada
155             override fun onFailure(call: Call, e: IOException) {
156                 e.printStackTrace()
157             }
158 
159             override fun onResponse(call: Call, response: Response) {
160                 // Como no tenemos nada que recibir, no hacemos nada
    con la respuesta
161             }
162 
163         })
164     }
165 }
```

```kotlin
package com.example.home_t

import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import android.widget.Toast
import com.example.home_t.databinding.ActivitySundayProgramBinding
import okhttp3.*
import java.io.IOException

/*
    Actividad SundayProgram: Controla la configuracion del programa del dia Domingo.
    Permite elegir la temperatura deseada, entre las dos temperaturas de programa actuales,
    en cada hora del dia. Ademas, se puede ver el programa actual.
 */

class SundayProgramActivity : AppCompatActivity() {

    private lateinit var binding: ActivitySundayProgramBinding

    // URL con la direccion IP del ESP
    private val ipUrl = "http://192.168.1.53/"

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        binding = ActivitySundayProgramBinding.inflate(layoutInflater)
        setContentView(binding.root)

        setupSundayProgram() // Funcion que controla la logica de la actividad SundayProgram
    }

    private fun setupSundayProgram() {
        // Se pulsa en ver el programa actual
        binding.updateButton.setOnClickListener {
            getProgram() // Funcion que obtiene el programa actual del microcontrolador
        }

        // Se pulsa en guardar el programa seleccionado
        binding.saveButton.setOnClickListener {

            var sundayProgram = ""

            // Se añaden a la variable las 24 temperaturas seleccionadas para cada hora del dia
            // La primera temepratura corresponde a las 12 de la noche, la ultima a las 11 de la noche
            sundayProgram += if (binding.zeroButtonT1.isChecked) { "0" } else { "1" }
            sundayProgram += if (binding.oneButtonT1.isChecked) { "0" } else { "1" }
```

```
46              sundayProgram += if (binding.twoButtonT1.isChecked) { "0"
    } else { "1" }
47              sundayProgram += if (binding.threeButtonT1.isChecked) { "0
   " } else { "1" }
48              sundayProgram += if (binding.fourButtonT1.isChecked) { "0"
    } else { "1" }
49              sundayProgram += if (binding.fiveButtonT1.isChecked) { "0"
    } else { "1" }
50              sundayProgram += if (binding.sixButtonT1.isChecked) { "0"
    } else { "1" }
51              sundayProgram += if (binding.sevenButtonT1.isChecked) { "0
   " } else { "1" }
52              sundayProgram += if (binding.eightButtonT1.isChecked) { "0
   " } else { "1" }
53              sundayProgram += if (binding.nineButtonT1.isChecked) { "0"
    } else { "1" }
54              sundayProgram += if (binding.tenButtonT1.isChecked) { "0"
    } else { "1" }
55              sundayProgram += if (binding.elevenButtonT1.isChecked) { "
   0" } else { "1" }
56              sundayProgram += if (binding.twelveButtonT1.isChecked) { "
   0" } else { "1" }
57              sundayProgram += if (binding.thirteenButtonT1.isChecked
   ) { "0" } else { "1" }
58              sundayProgram += if (binding.fourteenButtonT1.isChecked
   ) { "0" } else { "1" }
59              sundayProgram += if (binding.fifteenButtonT1.isChecked) {
   "0" } else { "1" }
60              sundayProgram += if (binding.sixteenButtonT1.isChecked) {
   "0" } else { "1" }
61              sundayProgram += if (binding.seventeenButtonT1.isChecked
   ) { "0" } else { "1" }
62              sundayProgram += if (binding.eighteenButtonT1.isChecked
   ) { "0" } else { "1" }
63              sundayProgram += if (binding.nineteenButtonT1.isChecked
   ) { "0" } else { "1" }
64              sundayProgram += if (binding.twentyButtonT1.isChecked) { "
   0" } else { "1" }
65              sundayProgram += if (binding.twentyOneButtonT1.isChecked
   ) { "0" } else { "1" }
66              sundayProgram += if (binding.twentyTwoButtonT1.isChecked
   ) { "0" } else { "1" }
67              sundayProgram += if (binding.twentyThreeButtonT1.isChecked
   ) { "0" } else { "1" }
68
69              sendProgram(sundayProgram) // Funcion que manda el
   programa deseado al microcontrolador
70
71              Toast.makeText(this, "Programa guardado", Toast.
   LENGTH_SHORT).show()
72          }
73
74      }
```

```
75
76      private fun getProgram() {
77          val client = OkHttpClient() // Se crea el cliente OkHttp
78          val url = ipUrl + "getprograma" + "d" // Se completa la URL
    con un string identificativo de la accion a realizar y el dia del
    programa
79          // Se define la llamada HTTP
80          val request = Request.Builder()
81              .url(url)
82              .addHeader("Connection", "close")
83              .build()
84
85          // Se realiza la llamada HTTP de forma asincrona
86          client.newCall(request).enqueue(object : Callback {
87
88              // Si ocurre algun problema en la llamada
89              override fun onFailure(call: Call, e: IOException) {
90                  e.printStackTrace()
91              }
92
93              // Si se recibe una respuesta a la llamada
94              override fun onResponse(call: Call, response: Response) {
95                  response.use {
96                      if (response.isSuccessful) { // La respuesta se
    ha recibido correctamente
97                          val myResponse = response.body?.string() //
    Se guarda el cuerpo de la respuesta
98
99                          // Se filtra el cuerpo de la repuesta por
    digitos (se obtiene el programa actual)
100                         val program = myResponse?.filter(Char::
    isDigit)
101
102                         // Los cambios visuales en la pantalla se
    hacen en el hilo donde corre el interfaz de usuario
103                         runOnUiThread {
104                             if (myResponse != null) {
105                                 // Se actualiza la pantalla con las
    temperaturas actuales del programa
106                                 if (program?.get(0) == '1') binding.
    zeroButtonT2.isChecked = true
107                                 if (program?.get(1) == '1') binding.
    oneButtonT2.isChecked = true
108                                 if (program?.get(2) == '1') binding.
    twoButtonT2.isChecked = true
109                                 if (program?.get(3) == '1') binding.
    threeButtonT2.isChecked = true
110                                 if (program?.get(4) == '1') binding.
    fourButtonT2.isChecked = true
111                                 if (program?.get(5) == '1') binding.
    fiveButtonT2.isChecked = true
112                                 if (program?.get(6) == '1') binding.
    sixButtonT2.isChecked = true
```

```
113                                 if (program?.get(7) == '1') binding.
    sevenButtonT2.isChecked = true
114                                 if (program?.get(8) == '1') binding.
    eightButtonT2.isChecked = true
115                                 if (program?.get(9) == '1') binding.
    nineButtonT2.isChecked = true
116                                 if (program?.get(10) == '1') binding.
    tenButtonT2.isChecked = true
117                                 if (program?.get(11) == '1') binding.
    elevenButtonT2.isChecked = true
118                                 if (program?.get(12) == '1') binding.
    twelveButtonT2.isChecked = true
119                                 if (program?.get(13) == '1') binding.
    thirteenButtonT2.isChecked = true
120                                 if (program?.get(14) == '1') binding.
    fourteenButtonT2.isChecked = true
121                                 if (program?.get(15) == '1') binding.
    fifteenButtonT2.isChecked = true
122                                 if (program?.get(16) == '1') binding.
    sixteenButtonT2.isChecked = true
123                                 if (program?.get(17) == '1') binding.
    seventeenButtonT2.isChecked = true
124                                 if (program?.get(18) == '1') binding.
    eighteenButtonT2.isChecked = true
125                                 if (program?.get(19) == '1') binding.
    nineteenButtonT2.isChecked = true
126                                 if (program?.get(20) == '1') binding.
    twentyButtonT2.isChecked = true
127                                 if (program?.get(21) == '1') binding.
    twentyOneButtonT2.isChecked = true
128                                 if (program?.get(22) == '1') binding.
    twentyTwoButtonT2.isChecked = true
129                                 if (program?.get(23) == '1') binding.
    twentyThreeButtonT2.isChecked = true
130
131                             }
132                         }
133                     }
134
135                 }
136             }
137         }
138         )
139     }
140
141     private fun sendProgram(program: String) {
142         val client = OkHttpClient() // Se crea el cliente OkHttp
143         val url = ipUrl + "progdia" + "d" + program // Se completa la
     URL con un string identificativo de la accion a realizar y el dia
    del programa
144         // Se define la llamada HTTP
145         val request = Request.Builder()
146             .url(url)
```

```
147             .addHeader("Connection", "close")
148             .build()
149 
150         // Se realiza la llamada HTTP de forma asincrona
151         client.newCall(request).enqueue(object : Callback {
152 
153             // Si ocurre algun problema en la llamada
154             override fun onFailure(call: Call, e: IOException) {
155                 e.printStackTrace()
156             }
157 
158             override fun onResponse(call: Call, response: Response) {
159                 // Como no tenemos nada que recibir, no hacemos nada
    con la respuesta
160             }
161 
162         })
163     }
164 }
```

```xml
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:padding="20dp"
    tools:context=".AuthenticationActivity">

    <ImageView
        android:id="@+id/app_icon"
        android:layout_width="64dp"
        android:layout_height="64dp"
        android:layout_marginTop="48dp"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:srcCompat="@drawable/home_t_icon" />

    <TextView
        android:id="@+id/app_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="16dp"
        android:text="HomeT"
        android:textAppearance="@style/TextAppearance.MaterialComponents.Body1"
        android:textSize="32sp"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/app_icon" />

    <TextView
        android:id="@+id/session_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="8dp"
        android:text="Inicia sesión para continuar"
        android:textAppearance="@style/TextAppearance.MaterialComponents.Body1"
        android:textSize="26sp"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/app_text" />

    <EditText
        android:id="@+id/email_edit_text"
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:layout_marginTop="32dp"
        android:ems="10"
        android:hint="Email"
```

```
        android:imeOptions="actionDone"
        android:inputType="textEmailAddress"
        android:minHeight="48dp"
        android:drawableStart="@drawable/ic_email"
        android:drawablePadding="6dp"
        android:padding="10dp"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/session_text" />

    <EditText
        android:id="@+id/password_edit_text"
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:layout_marginTop="16dp"
        android:ems="10"
        android:hint="Contraseña"
        android:imeOptions="actionDone"
        android:inputType="textPassword"
        android:minHeight="48dp"
        android:drawableStart="@drawable/ic_lock"
        android:drawablePadding="6dp"
        android:padding="10dp"
        app:layout_constraintStart_toStartOf="@id/email_edit_text"
        app:layout_constraintEnd_toEndOf="@id/email_edit_text"
        app:layout_constraintTop_toBottomOf="@+id/email_edit_text" />

    <TextView
        android:id="@+id/forgot_password_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="16dp"
        android:text="¿Has olvidado la contraseña?"
        android:textAppearance="@style/TextAppearance.
MaterialComponents.Body1"
        android:textSize="18sp"
        app:layout_constraintEnd_toEndOf="@+id/password_edit_text"
        app:layout_constraintTop_toBottomOf="@id/password_edit_text"
 />

    <Button
        android:id="@+id/sign_in_button"
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:layout_marginTop="32dp"
        android:text="Iniciar sesión"
        app:layout_constraintStart_toStartOf="@id/email_edit_text"
        app:layout_constraintEnd_toEndOf="@+id/email_edit_text"
        app:layout_constraintTop_toBottomOf="@+id/
forgot_password_text" />

    <Button
        android:id="@+id/sign_up_button"
```

```
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:layout_marginTop="16dp"
        android:text="Crear cuenta"
        app:layout_constraintStart_toStartOf="@id/email_edit_text"
        app:layout_constraintEnd_toEndOf="@+id/email_edit_text"
        app:layout_constraintTop_toBottomOf="@+id/sign_in_button" />

</androidx.constraintlayout.widget.ConstraintLayout>
```

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:padding="20dp"
    tools:context=".RegistrationActivity">

    <ImageView
        android:id="@+id/app_icon"
        android:layout_width="64dp"
        android:layout_height="64dp"
        android:layout_marginTop="48dp"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:srcCompat="@drawable/home_t_icon" />

    <TextView
        android:id="@+id/app_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="16dp"
        android:text="HomeT"
        android:textAppearance="@style/TextAppearance.MaterialComponents.Body1"
        android:textSize="32sp"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/app_icon" />

    <TextView
        android:id="@+id/account_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="8dp"
        android:text="Nueva cuenta"
        android:textAppearance="@style/TextAppearance.MaterialComponents.Body1"
        android:textSize="26sp"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/app_text" />

    <EditText
        android:id="@+id/email_edit_text"
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:layout_marginTop="16dp"
        android:ems="10"
        android:hint="Email"
```

```
        android:imeOptions="actionDone"
        android:inputType="textEmailAddress"
        android:minHeight="48dp"
        android:drawableStart="@drawable/ic_email"
        android:drawablePadding="6dp"
        android:padding="10dp"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/account_text" />

    <EditText
        android:id="@+id/password_edit_text"
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:layout_marginTop="16dp"
        android:ems="10"
        android:hint="Contraseña"
        android:imeOptions="actionDone"
        android:inputType="textPassword"
        android:minHeight="48dp"
        android:drawableStart="@drawable/ic_lock"
        android:drawablePadding="6dp"
        android:padding="10dp"
        app:layout_constraintStart_toStartOf="@+id/email_edit_text"
        app:layout_constraintEnd_toEndOf="@id/email_edit_text"
        app:layout_constraintTop_toBottomOf="@+id/email_edit_text" />

    <EditText
        android:id="@+id/password_edit_text_2"
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:layout_marginTop="16dp"
        android:ems="10"
        android:hint="Confirmar contraseña"
        android:imeOptions="actionDone"
        android:inputType="textPassword"
        android:minHeight="48dp"
        android:drawableStart="@drawable/ic_lock"
        android:drawablePadding="6dp"
        android:padding="10dp"
        app:layout_constraintStart_toStartOf="@id/email_edit_text"
        app:layout_constraintEnd_toEndOf="@id/email_edit_text"
        app:layout_constraintTop_toBottomOf="@+id/password_edit_text"
    />

    <Button
        android:id="@+id/create_account_button"
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:layout_marginTop="32dp"
        android:text="Crear cuenta"
        app:layout_constraintStart_toStartOf="@id/email_edit_text"
        app:layout_constraintEnd_toEndOf="@id/email_edit_text"
```

```
        app:layout_constraintTop_toBottomOf="@+id/
password_edit_text_2" />

    <Button
        android:id="@+id/go_back_button"
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:layout_marginTop="16dp"
        android:text="Volver"
        app:layout_constraintStart_toStartOf="@id/email_edit_text"
        app:layout_constraintEnd_toEndOf="@id/email_edit_text"
        app:layout_constraintTop_toBottomOf="@+id/
create_account_button" />

</androidx.constraintlayout.widget.ConstraintLayout>
```

```xml
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:padding="20dp"
    tools:context=".NewPasswordActivity">

    <ImageView
        android:id="@+id/app_icon"
        android:layout_width="64dp"
        android:layout_height="64dp"
        android:layout_marginTop="48dp"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:srcCompat="@drawable/home_t_icon" />

    <TextView
        android:id="@+id/app_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="16dp"
        android:text="HomeT"
        android:textAppearance="@style/TextAppearance.MaterialComponents.Body1"
        android:textSize="32sp"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/app_icon" />

    <TextView
        android:id="@+id/new_password_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="8dp"
        android:text="Nueva contraseña"
        android:textAppearance="@style/TextAppearance.MaterialComponents.Body1"
        android:textSize="26sp"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/app_text" />

    <EditText
        android:id="@+id/email_edit_text"
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:layout_marginTop="32dp"
        android:ems="10"
        android:hint="Introduce tu email"
```

```
        android:imeOptions="actionDone"
        android:inputType="textEmailAddress"
        android:minHeight="48dp"
        android:drawableStart="@drawable/ic_email"
        android:drawablePadding="6dp"
        android:padding="10dp"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/new_password_text"
    />

    <Button
        android:id="@+id/send_email_button"
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:layout_marginTop="32dp"
        android:text="Enviar correo"
        app:layout_constraintStart_toStartOf="@id/email_edit_text"
        app:layout_constraintEnd_toEndOf="@id/email_edit_text"
        app:layout_constraintTop_toBottomOf="@+id/email_edit_text" />

    <Button
        android:id="@+id/go_back_button"
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:layout_marginTop="16dp"
        android:text="Volver"
        app:layout_constraintStart_toStartOf="@id/send_email_button"
        app:layout_constraintEnd_toEndOf="@id/email_edit_text"
        app:layout_constraintTop_toBottomOf="@+id/send_email_button"
    />

</androidx.constraintlayout.widget.ConstraintLayout>
```

```xml
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".HomeActivity">

    <!-- Navegador de pantalla inferior, se le asocia el menu creado -->
    <!-- Permite seleciconar el fragmento a mostrar -->
    <com.google.android.material.bottomnavigation.BottomNavigationView
        android:id="@+id/bottomNavigation"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:background="?attr/colorSecondary"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:menu="@menu/menu" />

    <!-- Contenedor de los fragmentos, se le asocia el navegador creado -->
    <!-- Muestra los fragmentos -->
    <androidx.fragment.app.FragmentContainerView
        android:id="@+id/fragmentContainerView"
        android:name="androidx.navigation.fragment.NavHostFragment"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        app:defaultNavHost="true"
        app:layout_constraintBottom_toTopOf="@+id/bottomNavigation"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:navGraph="@navigation/my_navigation" />

</androidx.constraintlayout.widget.ConstraintLayout>
```

```xml
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".fragments.menu.HomeFragment">

    <TextView
        android:id="@+id/welcomeText"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="40dp"
        android:text="Bienvenido a HomeT"
        android:textAppearance="@style/TextAppearance.MaterialComponents.Body1"
        android:textSize="32sp"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

    <View
        android:id="@+id/underlineWelcome"
        android:layout_width="0dp"
        android:layout_height="3dp"
        android:background="@color/black"
        app:layout_constraintEnd_toEndOf="@id/welcomeText"
        app:layout_constraintStart_toStartOf="@id/welcomeText"
        app:layout_constraintTop_toBottomOf="@id/welcomeText" />

    <TextView
        android:id="@+id/profileText"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="32dp"
        android:text="CUENTA"
        android:textAppearance="@style/TextAppearance.MaterialComponents.Body1"
        android:textSize="32sp"
        app:layout_constraintStart_toStartOf="@id/welcomeText"
        app:layout_constraintEnd_toEndOf="@id/welcomeText"
        app:layout_constraintTop_toBottomOf="@id/welcomeText" />

    <View
        android:id="@+id/underlineProfile"
        android:layout_width="0dp"
        android:layout_height="3dp"
        android:background="?attr/colorPrimary"
        app:layout_constraintEnd_toEndOf="@id/profileText"
        app:layout_constraintStart_toStartOf="@id/profileText"
        app:layout_constraintTop_toBottomOf="@id/profileText" />

```

```
    <ImageView
        android:id="@+id/userIcon"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        app:layout_constraintBottom_toBottomOf="@+id/userText"
        app:layout_constraintStart_toStartOf="@id/welcomeText"
        app:layout_constraintTop_toTopOf="@+id/userText"
        app:srcCompat="@drawable/ic_user" />

    <TextView
        android:id="@+id/userText"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="16dp"
        android:textAppearance="@style/TextAppearance.
MaterialComponents.Body1"
        android:textSize="20sp"
        android:layout_marginStart="10dp"
        app:layout_constraintStart_toEndOf="@id/userIcon"
        app:layout_constraintTop_toBottomOf="@+id/profileText"
        tools:text="Usuario" />

    <ImageView
        android:id="@+id/logoutIcon"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        app:layout_constraintBottom_toBottomOf="@id/logoutButton"
        app:layout_constraintTop_toTopOf="@id/logoutButton"
        app:layout_constraintStart_toStartOf="@id/welcomeText"
        app:srcCompat="@drawable/ic_logout" />

    <Button
        android:id="@+id/logoutButton"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="16dp"
        android:text="Cerrar sesión"
        android:layout_marginStart="10dp"
        app:layout_constraintStart_toEndOf="@id/logoutIcon"
        app:layout_constraintTop_toBottomOf="@+id/userText" />

    <TextView
        android:id="@+id/valuesText"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="32dp"
        android:text="VALORES ACTUALES"
        android:textAppearance="@style/TextAppearance.
MaterialComponents.Body1"
        android:textSize="32sp"
        app:layout_constraintStart_toStartOf="@id/welcomeText"
        app:layout_constraintEnd_toEndOf="@id/welcomeText"
        app:layout_constraintTop_toBottomOf="@id/logoutButton" />
```

```
102
103     <View
104         android:id="@+id/underlineValues"
105         android:layout_width="0dp"
106         android:layout_height="3dp"
107         android:background="?attr/colorSecondary"
108         app:layout_constraintEnd_toEndOf="@id/valuesText"
109         app:layout_constraintStart_toStartOf="@id/valuesText"
110         app:layout_constraintTop_toBottomOf="@id/valuesText" />
111
112     <TextView
113         android:id="@+id/modeText"
114         android:layout_width="wrap_content"
115         android:layout_height="wrap_content"
116         android:layout_marginTop="16dp"
117         android:text="Modo:"
118         android:textAppearance="@style/TextAppearance.
    MaterialComponents.Body1"
119         android:textSize="32sp"
120         app:layout_constraintStart_toStartOf="@id/welcomeText"
121         app:layout_constraintTop_toBottomOf="@id/valuesText" />
122
123     <TextView
124         android:id="@+id/modeOnText"
125         android:layout_width="wrap_content"
126         android:layout_height="wrap_content"
127         android:layout_marginStart="8dp"
128         android:textAppearance="@style/TextAppearance.
    MaterialComponents.Body1"
129         android:textSize="32sp"
130         app:layout_constraintStart_toEndOf="@id/modeText"
131         app:layout_constraintTop_toTopOf="@id/modeText"
132         tools:text="Manual" />
133
134     <TextView
135         android:id="@+id/temperatureText"
136         android:layout_width="wrap_content"
137         android:layout_height="wrap_content"
138         android:layout_marginTop="16dp"
139         android:text="Temperatura:"
140         android:textAppearance="@style/TextAppearance.
    MaterialComponents.Body1"
141         android:textSize="32sp"
142         app:layout_constraintStart_toStartOf="@id/welcomeText"
143         app:layout_constraintTop_toBottomOf="@+id/modeOnText" />
144
145     <TextView
146         android:id="@+id/temperatureOnText"
147         android:layout_width="wrap_content"
148         android:layout_height="wrap_content"
149         android:layout_marginStart="8dp"
150         android:textAppearance="@style/TextAppearance.
    MaterialComponents.Body1"
```

```
        android:textSize="32sp"
        app:layout_constraintStart_toEndOf="@id/temperatureText"
        app:layout_constraintTop_toTopOf="@+id/temperatureText"
        tools:text="20,5 °C" />

</androidx.constraintlayout.widget.ConstraintLayout>
```

```xml
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".fragments.menu.ManualModeFragment">

    <!-- TODO: Update blank fragment layout -->
    <TextView
        android:id="@+id/manualText"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="40dp"
        android:text="MODO MANUAL"
        android:textAppearance="@style/TextAppearance.MaterialComponents.Body1"
        android:textSize="32sp"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

    <TextView
        android:id="@+id/manualExplainText"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="8dp"
        android:text="Introduce la temperatura, múltiplo de 0.5,"
        android:textAppearance="@style/TextAppearance.MaterialComponents.Body1"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/manualText" />

    <TextView
        android:id="@+id/manualExplainText1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="que deseas. Pulsa GUARDAR para pasar a"
        android:textAppearance="@style/TextAppearance.MaterialComponents.Body1"
        app:layout_constraintStart_toStartOf="@id/manualExplainText"
        app:layout_constraintTop_toBottomOf="@+id/manualExplainText" />

    <TextView
        android:id="@+id/manualExplainText2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Modo Manual con esa temperatura."
        android:textAppearance="@style/TextAppearance.MaterialComponents.Body1"
```

```
        app:layout_constraintStart_toStartOf="@id/manualExplainText"
        app:layout_constraintTop_toBottomOf="@+id/manualExplainText1"
    />

    <TextView
        android:id="@+id/temperatureText"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="16dp"
        android:text="Temperatura deseada"
        android:textAppearance="@style/TextAppearance.
MaterialComponents.Body1"
        android:textSize="30sp"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@id/manualExplainText2"
    />

    <EditText
        android:id="@+id/temperatureEditText"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="16dp"
        android:hint="18.5 °C"
        android:imeOptions="actionDone"
        android:inputType="numberDecimal"
        android:textSize="30sp"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@id/temperatureText" />

    <Button
        android:id="@+id/saveButton"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="16dp"
        android:text="Guardar"
        app:layout_constraintEnd_toEndOf="@id/temperatureEditText"
        app:layout_constraintStart_toStartOf="@id/temperatureEditText
"
        app:layout_constraintTop_toBottomOf="@+id/temperatureEditText
" />

</androidx.constraintlayout.widget.ConstraintLayout>
```

```xml
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".fragments.menu.ProgramModeFragment">

    <!-- TODO: Update blank fragment layout -->
    <TextView
        android:id="@+id/programText"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="40dp"
        android:text="MODO PROGRAMA"
        android:textAppearance="@style/TextAppearance.MaterialComponents.Body1"
        android:textSize="32sp"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

    <TextView
        android:id="@+id/programExplainText"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="8dp"
        android:text="Introduce dos temperaturas, múltiplo de 0.5,"
        android:textAppearance="@style/TextAppearance.MaterialComponents.Body1"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/programText" />

    <TextView
        android:id="@+id/programExplainText1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="que deseas. Para guardarlas y pasar a Modo"
        android:textAppearance="@style/TextAppearance.MaterialComponents.Body1"
        app:layout_constraintStart_toStartOf="@id/programExplainText"
        app:layout_constraintTop_toBottomOf="@id/programExplainText" />

    <TextView
        android:id="@+id/programExplainText2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Programa pulsa GUARDAR."
        android:textAppearance="@style/TextAppearance.MaterialComponents.Body1"
```

```
48          app:layout_constraintStart_toStartOf="@id/programExplainText"
49          app:layout_constraintTop_toBottomOf="@id/programExplainText1"
    />
50
51      <TextView
52          android:id="@+id/programExplainText3"
53          android:layout_width="wrap_content"
54          android:layout_height="wrap_content"
55          android:text="Pulsa en un día para configurar o ver el"
56          android:textAppearance="@style/TextAppearance.
    MaterialComponents.Body1"
57          app:layout_constraintStart_toStartOf="@id/programExplainText"
58          app:layout_constraintTop_toBottomOf="@id/programExplainText2"
    />
59
60      <TextView
61          android:id="@+id/programExplainText4"
62          android:layout_width="wrap_content"
63          android:layout_height="wrap_content"
64          android:text="programa actual."
65          android:textAppearance="@style/TextAppearance.
    MaterialComponents.Body1"
66          app:layout_constraintStart_toStartOf="@id/programExplainText"
67          app:layout_constraintTop_toBottomOf="@id/programExplainText3"
    />
68
69      <TextView
70          android:id="@+id/temperaturesText"
71          android:layout_width="wrap_content"
72          android:layout_height="wrap_content"
73          android:layout_marginTop="12dp"
74          android:text="Temperaturas deseadas"
75          android:textAppearance="@style/TextAppearance.
    MaterialComponents.Body1"
76          android:textSize="30sp"
77          app:layout_constraintEnd_toEndOf="parent"
78          app:layout_constraintStart_toStartOf="parent"
79          app:layout_constraintTop_toBottomOf="@id/programExplainText4"
    />
80
81      <EditText
82          android:id="@+id/temp1EditText"
83          android:layout_width="wrap_content"
84          android:layout_height="wrap_content"
85          android:layout_marginTop="12dp"
86          android:hint="18.5 °C"
87          android:imeOptions="actionDone"
88          android:inputType="numberDecimal"
89          android:textSize="32sp"
90          app:layout_constraintEnd_toEndOf="parent"
91          app:layout_constraintStart_toStartOf="parent"
92          app:layout_constraintTop_toBottomOf="@id/temperaturesText" />
93
```

```
    <TextView
        android:id="@+id/temp1Text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginStart="16dp"
        android:text="T1:"
        android:textAppearance="@style/TextAppearance.
MaterialComponents.Body1"
        android:textSize="32sp"
        app:layout_constraintBottom_toBottomOf="@id/temp1EditText"
        app:layout_constraintStart_toStartOf="@id/programText"
        app:layout_constraintTop_toTopOf="@+id/temp1EditText" />

    <EditText
        android:id="@+id/temp2EditText"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="12dp"
        android:hint="22.0 °C"
        android:imeOptions="actionDone"
        android:inputType="numberDecimal"
        android:textSize="32sp"
        app:layout_constraintStart_toStartOf="@id/temp1EditText"
        app:layout_constraintTop_toBottomOf="@id/temp1EditText" />

    <TextView
        android:id="@+id/temp2Text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginStart="16dp"
        android:text="T2:"
        android:textAppearance="@style/TextAppearance.
MaterialComponents.Body1"
        android:textSize="32sp"
        app:layout_constraintBottom_toBottomOf="@id/temp2EditText"
        app:layout_constraintStart_toStartOf="@id/programText"
        app:layout_constraintTop_toTopOf="@+id/temp2EditText" />

    <Button
        android:id="@+id/saveButton"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="12dp"
        android:text="Guardar"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/temp2EditText" />

    <TextView
        android:id="@+id/currentProgramsText"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="12dp"
```

```
145         android:text="Programas diarios"
146         android:textAppearance="@style/TextAppearance.
    MaterialComponents.Body1"
147         android:textSize="30sp"
148         app:layout_constraintEnd_toEndOf="parent"
149         app:layout_constraintStart_toStartOf="parent"
150         app:layout_constraintTop_toBottomOf="@id/saveButton" />
151 
152     <Button
153         android:id="@+id/mondayButton"
154         android:layout_width="wrap_content"
155         android:layout_height="wrap_content"
156         android:layout_marginTop="12dp"
157         android:text="L"
158         app:layout_constraintStart_toStartOf="@id/temperaturesText"
159         app:layout_constraintTop_toBottomOf="@+id/currentProgramsText
    " />
160 
161     <Button
162         android:id="@+id/tuesdayButton"
163         android:layout_width="wrap_content"
164         android:layout_height="wrap_content"
165         android:layout_marginTop="12dp"
166         android:text="M"
167         app:layout_constraintStart_toStartOf="parent"
168         app:layout_constraintEnd_toEndOf="parent"
169         app:layout_constraintTop_toBottomOf="@+id/currentProgramsText
    " />
170 
171     <Button
172         android:id="@+id/wednesdayButton"
173         android:layout_width="wrap_content"
174         android:layout_height="wrap_content"
175         android:layout_marginTop="12dp"
176         android:text="X"
177         app:layout_constraintEnd_toEndOf="@+id/temperaturesText"
178         app:layout_constraintTop_toBottomOf="@+id/currentProgramsText
    " />
179 
180     <Button
181         android:id="@+id/thursdayButton"
182         android:layout_width="wrap_content"
183         android:layout_height="wrap_content"
184         android:layout_marginTop="12dp"
185         android:text="J"
186         app:layout_constraintStart_toStartOf="@id/mondayButton"
187         app:layout_constraintTop_toBottomOf="@+id/mondayButton" />
188 
189     <Button
190         android:id="@+id/fridayButton"
191         android:layout_width="wrap_content"
192         android:layout_height="wrap_content"
193         android:layout_marginTop="12dp"
```

```
        android:text="V"
        app:layout_constraintStart_toStartOf="@id/tuesdayButton"
        app:layout_constraintTop_toBottomOf="@+id/tuesdayButton" />

    <Button
        android:id="@+id/saturdayButton"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="12dp"
        android:text="S"
        app:layout_constraintStart_toStartOf="@id/wednesdayButton"
        app:layout_constraintTop_toBottomOf="@id/wednesdayButton" />

    <Button
        android:id="@+id/sundayButton"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="12dp"
        android:text="D"
        app:layout_constraintStart_toStartOf="@id/fridayButton"
        app:layout_constraintTop_toBottomOf="@id/fridayButton" />

</androidx.constraintlayout.widget.ConstraintLayout>
```

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MondayProgramActivity">

    <TextView
        android:id="@+id/day_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="LUNES"
        android:textSize="22sp"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintTop_toBottomOf="@id/nineteen_hour_group" />

    <Button
        android:id="@+id/save_button"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Guardar"
        app:layout_constraintStart_toStartOf="@+id/nineteen_hour_group"
        app:layout_constraintTop_toBottomOf="@+id/day_text" />

    <Button
        android:id="@+id/update_button"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginStart="12dp"
        android:text="Ver actual"
        app:layout_constraintStart_toEndOf="@+id/save_button"
        app:layout_constraintTop_toTopOf="@id/save_button" />

    <View
        android:id="@+id/divider_h_up"
        android:layout_width="0dp"
        android:layout_height="2dp"
        android:layout_marginTop="-3dp"
        android:background="?attr/colorPrimaryVariant"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="@+id/eight_hour_text" />

    <View
        android:id="@+id/divider_v"
        android:layout_width="2dp"
        android:layout_height="0dp"
        android:layout_marginStart="5dp"
```

```
        android:background="?attr/colorPrimaryVariant"
        app:layout_constraintBottom_toBottomOf="@+id/divider_h_down"
        app:layout_constraintStart_toEndOf="@+id/eight_hour_group"
        app:layout_constraintTop_toTopOf="@id/divider_h_up" />

    <View
        android:id="@+id/divider_h_down"
        android:layout_width="0dp"
        android:layout_height="2dp"
        android:layout_marginTop="2dp"
        android:background="?attr/colorPrimaryVariant"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@id/nineteen_hour_text"
/>

    <TextView
        android:id="@+id/eight_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginStart="6dp"
        android:layout_marginTop="14dp"
        android:text="08:00-09:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

    <RadioGroup
        android:id="@+id/eight_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/eight_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/eight_hour_text"
        app:layout_constraintStart_toEndOf="@+id/eight_hour_text"
        app:layout_constraintTop_toTopOf="@+id/eight_hour_text">

        <RadioButton
            android:id="@+id/eight_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/eight_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/nine_hour_text"
        android:layout_width="wrap_content"
```

```
        android:layout_height="wrap_content"
        android:layout_marginTop="30dp"
        android:text="09:00-10:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@id/eight_hour_text"
        app:layout_constraintTop_toBottomOf="@id/eight_hour_text" />

    <RadioGroup
        android:id="@+id/nine_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/nine_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/nine_hour_text"
        app:layout_constraintStart_toStartOf="@+id/eight_hour_group"
        app:layout_constraintTop_toTopOf="@+id/nine_hour_text">

        <RadioButton
            android:id="@+id/nine_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/nine_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/ten_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="30dp"
        android:text="10:00-11:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@+id/eight_hour_text"
        app:layout_constraintTop_toBottomOf="@id/nine_hour_text" />

    <RadioGroup
        android:id="@+id/ten_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/ten_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/ten_hour_text"
        app:layout_constraintStart_toStartOf="@+id/eight_hour_group"
        app:layout_constraintTop_toTopOf="@+id/ten_hour_text">

        <RadioButton
            android:id="@+id/ten_button_T1"
            android:layout_width="wrap_content"
```

```
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/ten_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/eleven_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="30dp"
        android:text="11:00-12:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@id/eight_hour_text"
        app:layout_constraintTop_toBottomOf="@id/ten_hour_text" />

    <RadioGroup
        android:id="@+id/eleven_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/eleven_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/eleven_hour_text
"
        app:layout_constraintStart_toStartOf="@+id/eight_hour_group"
        app:layout_constraintTop_toTopOf="@+id/eleven_hour_text">

        <RadioButton
            android:id="@+id/eleven_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/eleven_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/twelve_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="30dp"
        android:text="12:00-13:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@id/eight_hour_text"
        app:layout_constraintTop_toBottomOf="@id/eleven_hour_text" />
```

```
208
209     <RadioGroup
210         android:id="@+id/twelve_hour_group"
211         android:layout_width="wrap_content"
212         android:layout_height="wrap_content"
213         android:checkedButton="@id/twelve_button_T1"
214         android:orientation="horizontal"
215         app:layout_constraintBottom_toBottomOf="@+id/twelve_hour_text
    "
216         app:layout_constraintStart_toStartOf="@+id/eight_hour_group"
217         app:layout_constraintTop_toTopOf="@+id/twelve_hour_text">
218
219         <RadioButton
220             android:id="@+id/twelve_button_T1"
221             android:layout_width="wrap_content"
222             android:layout_height="wrap_content"
223             android:text="T1" />
224
225         <RadioButton
226             android:id="@+id/twelve_button_T2"
227             android:layout_width="wrap_content"
228             android:layout_height="wrap_content"
229             android:text="T2" />
230     </RadioGroup>
231
232     <TextView
233         android:id="@+id/thirteen_hour_text"
234         android:layout_width="wrap_content"
235         android:layout_height="wrap_content"
236         android:layout_marginTop="30dp"
237         android:text="13:00-14:00"
238         android:textSize="16sp"
239         app:layout_constraintStart_toStartOf="@id/eight_hour_text"
240         app:layout_constraintTop_toBottomOf="@id/twelve_hour_text" />
241
242     <RadioGroup
243         android:id="@+id/thirteen_hour_group"
244         android:layout_width="wrap_content"
245         android:layout_height="wrap_content"
246         android:checkedButton="@id/thirteen_button_T1"
247         android:orientation="horizontal"
248         app:layout_constraintBottom_toBottomOf="@+id/
    thirteen_hour_text"
249         app:layout_constraintStart_toStartOf="@+id/eight_hour_group"
250         app:layout_constraintTop_toTopOf="@+id/thirteen_hour_text">
251
252         <RadioButton
253             android:id="@+id/thirteen_button_T1"
254             android:layout_width="wrap_content"
255             android:layout_height="wrap_content"
256             android:text="T1" />
257
258         <RadioButton
```

```
            android:id="@+id/thirteen_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/fourteen_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="30dp"
        android:text="14:00-15:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@id/eight_hour_text"
        app:layout_constraintTop_toBottomOf="@id/thirteen_hour_text"
    />

    <RadioGroup
        android:id="@+id/fourteen_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/fourteen_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/
    fourteen_hour_text"
        app:layout_constraintStart_toStartOf="@+id/eight_hour_group"
        app:layout_constraintTop_toTopOf="@+id/fourteen_hour_text">

        <RadioButton
            android:id="@+id/fourteen_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/fourteen_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/fifteen_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="30dp"
        android:text="15:00-16:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@id/eight_hour_text"
        app:layout_constraintTop_toBottomOf="@id/fourteen_hour_text"
    />

    <RadioGroup
```

```
309         android:id="@+id/fifteen_hour_group"
310         android:layout_width="wrap_content"
311         android:layout_height="wrap_content"
312         android:checkedButton="@id/fifteen_button_T1"
313         android:orientation="horizontal"
314         app:layout_constraintBottom_toBottomOf="@+id/
    fifteen_hour_text"
315         app:layout_constraintStart_toStartOf="@+id/eight_hour_group"
316         app:layout_constraintTop_toTopOf="@+id/fifteen_hour_text">
317 
318         <RadioButton
319             android:id="@+id/fifteen_button_T1"
320             android:layout_width="wrap_content"
321             android:layout_height="wrap_content"
322             android:text="T1" />
323 
324         <RadioButton
325             android:id="@+id/fifteen_button_T2"
326             android:layout_width="wrap_content"
327             android:layout_height="wrap_content"
328             android:text="T2" />
329     </RadioGroup>
330 
331     <TextView
332         android:id="@+id/sixteen_hour_text"
333         android:layout_width="wrap_content"
334         android:layout_height="wrap_content"
335         android:layout_marginTop="30dp"
336         android:text="16:00-17:00"
337         android:textSize="16sp"
338         app:layout_constraintStart_toStartOf="@id/eight_hour_text"
339         app:layout_constraintTop_toBottomOf="@id/fifteen_hour_text"
     />
340 
341     <RadioGroup
342         android:id="@+id/sixteen_hour_group"
343         android:layout_width="wrap_content"
344         android:layout_height="wrap_content"
345         android:checkedButton="@id/sixteen_button_T1"
346         android:orientation="horizontal"
347         app:layout_constraintBottom_toBottomOf="@+id/
    sixteen_hour_text"
348         app:layout_constraintStart_toStartOf="@+id/eight_hour_group"
349         app:layout_constraintTop_toTopOf="@+id/sixteen_hour_text">
350 
351         <RadioButton
352             android:id="@+id/sixteen_button_T1"
353             android:layout_width="wrap_content"
354             android:layout_height="wrap_content"
355             android:text="T1" />
356 
357         <RadioButton
358             android:id="@+id/sixteen_button_T2"
```

```
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/seventeen_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="30dp"
        android:text="17:00-18:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@id/eight_hour_text"
        app:layout_constraintTop_toBottomOf="@id/sixteen_hour_text"
    />

    <RadioGroup
        android:id="@+id/seventeen_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/seventeen_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/
seventeen_hour_text"
        app:layout_constraintStart_toStartOf="@+id/eight_hour_group"
        app:layout_constraintTop_toTopOf="@+id/seventeen_hour_text">

        <RadioButton
            android:id="@+id/seventeen_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/seventeen_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/eighteen_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="30dp"
        android:text="18:00-19:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@id/eight_hour_text"
        app:layout_constraintTop_toBottomOf="@id/seventeen_hour_text"
    />

    <RadioGroup
        android:id="@+id/eighteen_hour_group"
```

```
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/eighteen_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/
eighteen_hour_text"
        app:layout_constraintStart_toStartOf="@+id/eight_hour_group"
        app:layout_constraintTop_toTopOf="@+id/eighteen_hour_text">

        <RadioButton
            android:id="@+id/eighteen_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/eighteen_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/nineteen_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="30dp"
        android:text="19:00-20:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@id/eight_hour_text"
        app:layout_constraintTop_toBottomOf="@id/eighteen_hour_text"
 />

    <RadioGroup
        android:id="@+id/nineteen_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/nineteen_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/
nineteen_hour_text"
        app:layout_constraintStart_toStartOf="@+id/eight_hour_group"
        app:layout_constraintTop_toTopOf="@+id/nineteen_hour_text">

        <RadioButton
            android:id="@+id/nineteen_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/nineteen_button_T2"
            android:layout_width="wrap_content"
```

```
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/twenty_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginStart="10dp"
        android:text="20:00-21:00"
        android:textSize="16sp"
        app:layout_constraintStart_toEndOf="@+id/eight_hour_group"
        app:layout_constraintTop_toTopOf="@+id/eight_hour_text" />

    <RadioGroup
        android:id="@+id/twenty_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/twenty_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/twenty_hour_text
"
        app:layout_constraintStart_toEndOf="@+id/twenty_hour_text"
        app:layout_constraintTop_toTopOf="@+id/twenty_hour_text">

        <RadioButton
            android:id="@+id/twenty_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/twenty_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/twenty_one_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="21:00-22:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_text"
        app:layout_constraintTop_toTopOf="@id/nine_hour_text" />

    <RadioGroup
        android:id="@+id/twenty_one_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/twenty_one_button_T1"
        android:orientation="horizontal"
```

```
511         app:layout_constraintBottom_toBottomOf="@+id/
    twenty_one_hour_text"
512         app:layout_constraintStart_toStartOf="@+id/twenty_hour_group"
513         app:layout_constraintTop_toTopOf="@+id/twenty_one_hour_text">
514 
515         <RadioButton
516             android:id="@+id/twenty_one_button_T1"
517             android:layout_width="wrap_content"
518             android:layout_height="wrap_content"
519             android:text="T1" />
520 
521         <RadioButton
522             android:id="@+id/twenty_one_button_T2"
523             android:layout_width="wrap_content"
524             android:layout_height="wrap_content"
525             android:text="T2" />
526     </RadioGroup>
527 
528     <TextView
529         android:id="@+id/twenty_two_hour_text"
530         android:layout_width="wrap_content"
531         android:layout_height="wrap_content"
532         android:text="22:00-23:00"
533         android:textSize="16sp"
534         app:layout_constraintStart_toStartOf="@+id/twenty_hour_text"
535         app:layout_constraintTop_toTopOf="@id/ten_hour_text" />
536 
537     <RadioGroup
538         android:id="@+id/twenty_two_hour_group"
539         android:layout_width="wrap_content"
540         android:layout_height="wrap_content"
541         android:checkedButton="@id/twenty_two_button_T1"
542         android:orientation="horizontal"
543         app:layout_constraintBottom_toBottomOf="@+id/
    twenty_two_hour_text"
544         app:layout_constraintStart_toEndOf="@+id/twenty_two_hour_text
    "
545         app:layout_constraintTop_toTopOf="@+id/twenty_two_hour_text">
546 
547         <RadioButton
548             android:id="@+id/twenty_two_button_T1"
549             android:layout_width="wrap_content"
550             android:layout_height="wrap_content"
551             android:text="T1" />
552 
553         <RadioButton
554             android:id="@+id/twenty_two_button_T2"
555             android:layout_width="wrap_content"
556             android:layout_height="wrap_content"
557             android:text="T2" />
558     </RadioGroup>
559 
560     <TextView
```

```
        android:id="@+id/twenty_three_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="23:00-00:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_text"
        app:layout_constraintTop_toTopOf="@id/eleven_hour_text" />

    <RadioGroup
        android:id="@+id/twenty_three_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/twenty_three_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/
twenty_three_hour_text"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_group"
        app:layout_constraintTop_toTopOf="@+id/twenty_three_hour_text
">

        <RadioButton
            android:id="@+id/twenty_three_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/twenty_three_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/zero_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="00:00-01:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_text"
        app:layout_constraintTop_toTopOf="@id/twelve_hour_text" />

    <RadioGroup
        android:id="@+id/zero_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/zero_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/zero_hour_text"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_group"
        app:layout_constraintTop_toTopOf="@+id/zero_hour_text">

        <RadioButton
```

```
            android:id="@+id/zero_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/zero_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/one_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="01:00-02:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_text"
        app:layout_constraintTop_toTopOf="@id/thirteen_hour_text" />

    <RadioGroup
        android:id="@+id/one_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/one_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/one_hour_text"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_group"
        app:layout_constraintTop_toTopOf="@+id/one_hour_text">

        <RadioButton
            android:id="@+id/one_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/one_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/two_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="02:00-03:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_text"
        app:layout_constraintTop_toTopOf="@id/fourteen_hour_text" />

```

```
    <RadioGroup
        android:id="@+id/two_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/two_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/two_hour_text"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_group"
        app:layout_constraintTop_toTopOf="@+id/two_hour_text">

        <RadioButton
            android:id="@+id/two_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/two_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/three_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="03:00-04:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_text"
        app:layout_constraintTop_toTopOf="@id/fifteen_hour_text" />

    <RadioGroup
        android:id="@+id/three_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/three_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/three_hour_text"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_group"
        app:layout_constraintTop_toTopOf="@+id/three_hour_text">

        <RadioButton
            android:id="@+id/three_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/three_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
```

```
    </RadioGroup>

    <TextView
        android:id="@+id/four_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="04:00-05:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_text"
        app:layout_constraintTop_toTopOf="@id/sixteen_hour_text" />

    <RadioGroup
        android:id="@+id/four_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/four_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/four_hour_text"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_group"
        app:layout_constraintTop_toTopOf="@+id/four_hour_text">

        <RadioButton
            android:id="@+id/four_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/four_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/five_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="05:00-06:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_text"
        app:layout_constraintTop_toTopOf="@id/seventeen_hour_text" />

    <RadioGroup
        android:id="@+id/five_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/five_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/five_hour_text"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_group"
        app:layout_constraintTop_toTopOf="@+id/five_hour_text">

```

```
        <RadioButton
            android:id="@+id/five_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/five_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/six_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="06:00-07:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_text"
        app:layout_constraintTop_toTopOf="@id/eighteen_hour_text" />

    <RadioGroup
        android:id="@+id/six_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/six_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/six_hour_text"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_group"
        app:layout_constraintTop_toTopOf="@+id/six_hour_text">

        <RadioButton
            android:id="@+id/six_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/six_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/seven_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="07:00-08:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_text"
        app:layout_constraintTop_toTopOf="@id/nineteen_hour_text" />
```

```
824
825     <RadioGroup
826         android:id="@+id/seven_hour_group"
827         android:layout_width="wrap_content"
828         android:layout_height="wrap_content"
829         android:checkedButton="@id/seven_button_T1"
830         android:orientation="horizontal"
831         app:layout_constraintBottom_toBottomOf="@+id/seven_hour_text"
832         app:layout_constraintStart_toStartOf="@+id/twenty_hour_group"
833         app:layout_constraintTop_toTopOf="@+id/seven_hour_text">
834
835         <RadioButton
836             android:id="@+id/seven_button_T1"
837             android:layout_width="wrap_content"
838             android:layout_height="wrap_content"
839             android:text="T1" />
840
841         <RadioButton
842             android:id="@+id/seven_button_T2"
843             android:layout_width="wrap_content"
844             android:layout_height="wrap_content"
845             android:text="T2" />
846     </RadioGroup>
847
848 </androidx.constraintlayout.widget.ConstraintLayout>
```

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".TuesdayProgramActivity">

    <TextView
        android:id="@+id/day_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="MARTES"
        android:textSize="22sp"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintTop_toBottomOf="@id/nineteen_hour_group" />

    <Button
        android:id="@+id/save_button"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Guardar"
        app:layout_constraintStart_toStartOf="@+id/nineteen_hour_group"
        app:layout_constraintTop_toBottomOf="@+id/day_text" />

    <Button
        android:id="@+id/update_button"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginStart="12dp"
        android:text="Ver actual"
        app:layout_constraintStart_toEndOf="@+id/save_button"
        app:layout_constraintTop_toTopOf="@id/save_button" />

    <View
        android:id="@+id/divider_h_up"
        android:layout_width="0dp"
        android:layout_height="2dp"
        android:layout_marginTop="-3dp"
        android:background="?attr/colorPrimaryVariant"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="@+id/eight_hour_text" />

    <View
        android:id="@+id/divider_v"
        android:layout_width="2dp"
        android:layout_height="0dp"
        android:layout_marginStart="5dp"
```

```
        android:background="?attr/colorPrimaryVariant"
        app:layout_constraintBottom_toBottomOf="@+id/divider_h_down"
        app:layout_constraintStart_toEndOf="@+id/eight_hour_group"
        app:layout_constraintTop_toTopOf="@id/divider_h_up" />

    <View
        android:id="@+id/divider_h_down"
        android:layout_width="0dp"
        android:layout_height="2dp"
        android:layout_marginTop="2dp"
        android:background="?attr/colorPrimaryVariant"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@id/nineteen_hour_text"
/>

    <TextView
        android:id="@+id/eight_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginStart="6dp"
        android:layout_marginTop="14dp"
        android:text="08:00-09:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

    <RadioGroup
        android:id="@+id/eight_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/eight_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/eight_hour_text"
        app:layout_constraintStart_toEndOf="@+id/eight_hour_text"
        app:layout_constraintTop_toTopOf="@+id/eight_hour_text">

        <RadioButton
            android:id="@+id/eight_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/eight_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/nine_hour_text"
        android:layout_width="wrap_content"
```

```
        android:layout_height="wrap_content"
        android:layout_marginTop="30dp"
        android:text="09:00-10:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@id/eight_hour_text"
        app:layout_constraintTop_toBottomOf="@id/eight_hour_text" />

    <RadioGroup
        android:id="@+id/nine_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/nine_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/nine_hour_text"
        app:layout_constraintStart_toStartOf="@+id/eight_hour_group"
        app:layout_constraintTop_toTopOf="@+id/nine_hour_text">

        <RadioButton
            android:id="@+id/nine_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/nine_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/ten_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="30dp"
        android:text="10:00-11:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@+id/eight_hour_text"
        app:layout_constraintTop_toBottomOf="@id/nine_hour_text" />

    <RadioGroup
        android:id="@+id/ten_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/ten_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/ten_hour_text"
        app:layout_constraintStart_toStartOf="@+id/eight_hour_group"
        app:layout_constraintTop_toTopOf="@+id/ten_hour_text">

        <RadioButton
            android:id="@+id/ten_button_T1"
            android:layout_width="wrap_content"
```

```
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/ten_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/eleven_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="30dp"
        android:text="11:00-12:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@id/eight_hour_text"
        app:layout_constraintTop_toBottomOf="@id/ten_hour_text" />

    <RadioGroup
        android:id="@+id/eleven_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/eleven_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/eleven_hour_text
"
        app:layout_constraintStart_toStartOf="@+id/eight_hour_group"
        app:layout_constraintTop_toTopOf="@+id/eleven_hour_text">

        <RadioButton
            android:id="@+id/eleven_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/eleven_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/twelve_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="30dp"
        android:text="12:00-13:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@id/eight_hour_text"
        app:layout_constraintTop_toBottomOf="@id/eleven_hour_text" />
```

```
208
209     <RadioGroup
210         android:id="@+id/twelve_hour_group"
211         android:layout_width="wrap_content"
212         android:layout_height="wrap_content"
213         android:checkedButton="@id/twelve_button_T1"
214         android:orientation="horizontal"
215         app:layout_constraintBottom_toBottomOf="@+id/twelve_hour_text
    "
216         app:layout_constraintStart_toStartOf="@+id/eight_hour_group"
217         app:layout_constraintTop_toTopOf="@+id/twelve_hour_text">
218
219         <RadioButton
220             android:id="@+id/twelve_button_T1"
221             android:layout_width="wrap_content"
222             android:layout_height="wrap_content"
223             android:text="T1" />
224
225         <RadioButton
226             android:id="@+id/twelve_button_T2"
227             android:layout_width="wrap_content"
228             android:layout_height="wrap_content"
229             android:text="T2" />
230     </RadioGroup>
231
232     <TextView
233         android:id="@+id/thirteen_hour_text"
234         android:layout_width="wrap_content"
235         android:layout_height="wrap_content"
236         android:layout_marginTop="30dp"
237         android:text="13:00-14:00"
238         android:textSize="16sp"
239         app:layout_constraintStart_toStartOf="@id/eight_hour_text"
240         app:layout_constraintTop_toBottomOf="@id/twelve_hour_text" />
241
242     <RadioGroup
243         android:id="@+id/thirteen_hour_group"
244         android:layout_width="wrap_content"
245         android:layout_height="wrap_content"
246         android:checkedButton="@id/thirteen_button_T1"
247         android:orientation="horizontal"
248         app:layout_constraintBottom_toBottomOf="@+id/
    thirteen_hour_text"
249         app:layout_constraintStart_toStartOf="@+id/eight_hour_group"
250         app:layout_constraintTop_toTopOf="@+id/thirteen_hour_text">
251
252         <RadioButton
253             android:id="@+id/thirteen_button_T1"
254             android:layout_width="wrap_content"
255             android:layout_height="wrap_content"
256             android:text="T1" />
257
258         <RadioButton
```

```
            android:id="@+id/thirteen_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/fourteen_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="30dp"
        android:text="14:00-15:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@id/eight_hour_text"
        app:layout_constraintTop_toBottomOf="@id/thirteen_hour_text"
    />

    <RadioGroup
        android:id="@+id/fourteen_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/fourteen_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/
    fourteen_hour_text"
        app:layout_constraintStart_toStartOf="@+id/eight_hour_group"
        app:layout_constraintTop_toTopOf="@+id/fourteen_hour_text">

        <RadioButton
            android:id="@+id/fourteen_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/fourteen_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/fifteen_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="30dp"
        android:text="15:00-16:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@id/eight_hour_text"
        app:layout_constraintTop_toBottomOf="@id/fourteen_hour_text"
    />

    <RadioGroup
```

```
309         android:id="@+id/fifteen_hour_group"
310         android:layout_width="wrap_content"
311         android:layout_height="wrap_content"
312         android:checkedButton="@id/fifteen_button_T1"
313         android:orientation="horizontal"
314         app:layout_constraintBottom_toBottomOf="@+id/
    fifteen_hour_text"
315         app:layout_constraintStart_toStartOf="@+id/eight_hour_group"
316         app:layout_constraintTop_toTopOf="@+id/fifteen_hour_text">
317 
318         <RadioButton
319             android:id="@+id/fifteen_button_T1"
320             android:layout_width="wrap_content"
321             android:layout_height="wrap_content"
322             android:text="T1" />
323 
324         <RadioButton
325             android:id="@+id/fifteen_button_T2"
326             android:layout_width="wrap_content"
327             android:layout_height="wrap_content"
328             android:text="T2" />
329     </RadioGroup>
330 
331     <TextView
332         android:id="@+id/sixteen_hour_text"
333         android:layout_width="wrap_content"
334         android:layout_height="wrap_content"
335         android:layout_marginTop="30dp"
336         android:text="16:00-17:00"
337         android:textSize="16sp"
338         app:layout_constraintStart_toStartOf="@id/eight_hour_text"
339         app:layout_constraintTop_toBottomOf="@id/fifteen_hour_text"
     />
340 
341     <RadioGroup
342         android:id="@+id/sixteen_hour_group"
343         android:layout_width="wrap_content"
344         android:layout_height="wrap_content"
345         android:checkedButton="@id/sixteen_button_T1"
346         android:orientation="horizontal"
347         app:layout_constraintBottom_toBottomOf="@+id/
    sixteen_hour_text"
348         app:layout_constraintStart_toStartOf="@+id/eight_hour_group"
349         app:layout_constraintTop_toTopOf="@+id/sixteen_hour_text">
350 
351         <RadioButton
352             android:id="@+id/sixteen_button_T1"
353             android:layout_width="wrap_content"
354             android:layout_height="wrap_content"
355             android:text="T1" />
356 
357         <RadioButton
358             android:id="@+id/sixteen_button_T2"
```

```
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/seventeen_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="30dp"
        android:text="17:00-18:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@id/eight_hour_text"
        app:layout_constraintTop_toBottomOf="@id/sixteen_hour_text"
    />

    <RadioGroup
        android:id="@+id/seventeen_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/seventeen_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/
seventeen_hour_text"
        app:layout_constraintStart_toStartOf="@+id/eight_hour_group"
        app:layout_constraintTop_toTopOf="@+id/seventeen_hour_text">

        <RadioButton
            android:id="@+id/seventeen_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/seventeen_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/eighteen_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="30dp"
        android:text="18:00-19:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@id/eight_hour_text"
        app:layout_constraintTop_toBottomOf="@id/seventeen_hour_text"
    />

    <RadioGroup
        android:id="@+id/eighteen_hour_group"
```

```
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/eighteen_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/
eighteen_hour_text"
        app:layout_constraintStart_toStartOf="@+id/eight_hour_group"
        app:layout_constraintTop_toTopOf="@+id/eighteen_hour_text">

        <RadioButton
            android:id="@+id/eighteen_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/eighteen_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/nineteen_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="30dp"
        android:text="19:00-20:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@id/eight_hour_text"
        app:layout_constraintTop_toBottomOf="@id/eighteen_hour_text"
 />

    <RadioGroup
        android:id="@+id/nineteen_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/nineteen_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/
nineteen_hour_text"
        app:layout_constraintStart_toStartOf="@+id/eight_hour_group"
        app:layout_constraintTop_toTopOf="@+id/nineteen_hour_text">

        <RadioButton
            android:id="@+id/nineteen_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/nineteen_button_T2"
            android:layout_width="wrap_content"
```

```
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/twenty_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginStart="10dp"
        android:text="20:00-21:00"
        android:textSize="16sp"
        app:layout_constraintStart_toEndOf="@+id/eight_hour_group"
        app:layout_constraintTop_toTopOf="@+id/eight_hour_text" />

    <RadioGroup
        android:id="@+id/twenty_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/twenty_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/twenty_hour_text
"
        app:layout_constraintStart_toEndOf="@+id/twenty_hour_text"
        app:layout_constraintTop_toTopOf="@+id/twenty_hour_text">

        <RadioButton
            android:id="@+id/twenty_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/twenty_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/twenty_one_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="21:00-22:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_text"
        app:layout_constraintTop_toTopOf="@id/nine_hour_text" />

    <RadioGroup
        android:id="@+id/twenty_one_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/twenty_one_button_T1"
        android:orientation="horizontal"
```

```
        app:layout_constraintBottom_toBottomOf="@+id/
    twenty_one_hour_text"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_group"
        app:layout_constraintTop_toTopOf="@+id/twenty_one_hour_text">

        <RadioButton
            android:id="@+id/twenty_one_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/twenty_one_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/twenty_two_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="22:00-23:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_text"
        app:layout_constraintTop_toTopOf="@id/ten_hour_text" />

    <RadioGroup
        android:id="@+id/twenty_two_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/twenty_two_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/
    twenty_two_hour_text"
        app:layout_constraintStart_toEndOf="@+id/twenty_two_hour_text
    "
        app:layout_constraintTop_toTopOf="@+id/twenty_two_hour_text">

        <RadioButton
            android:id="@+id/twenty_two_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/twenty_two_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
```

```
        android:id="@+id/twenty_three_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="23:00-00:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_text"
        app:layout_constraintTop_toTopOf="@id/eleven_hour_text" />

    <RadioGroup
        android:id="@+id/twenty_three_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/twenty_three_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/
twenty_three_hour_text"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_group"
        app:layout_constraintTop_toTopOf="@+id/twenty_three_hour_text
">

        <RadioButton
            android:id="@+id/twenty_three_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/twenty_three_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/zero_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="00:00-01:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_text"
        app:layout_constraintTop_toTopOf="@id/twelve_hour_text" />

    <RadioGroup
        android:id="@+id/zero_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/zero_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/zero_hour_text"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_group"
        app:layout_constraintTop_toTopOf="@+id/zero_hour_text">

        <RadioButton
```

```
            android:id="@+id/zero_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/zero_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/one_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="01:00-02:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_text"
        app:layout_constraintTop_toTopOf="@id/thirteen_hour_text" />

    <RadioGroup
        android:id="@+id/one_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/one_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/one_hour_text"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_group"
        app:layout_constraintTop_toTopOf="@+id/one_hour_text">

        <RadioButton
            android:id="@+id/one_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/one_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/two_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="02:00-03:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_text"
        app:layout_constraintTop_toTopOf="@id/fourteen_hour_text" />

```

```
    <RadioGroup
        android:id="@+id/two_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/two_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/two_hour_text"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_group"
        app:layout_constraintTop_toTopOf="@+id/two_hour_text">

        <RadioButton
            android:id="@+id/two_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/two_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/three_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="03:00-04:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_text"
        app:layout_constraintTop_toTopOf="@id/fifteen_hour_text" />

    <RadioGroup
        android:id="@+id/three_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/three_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/three_hour_text"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_group"
        app:layout_constraintTop_toTopOf="@+id/three_hour_text">

        <RadioButton
            android:id="@+id/three_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/three_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
```

```
    </RadioGroup>

    <TextView
        android:id="@+id/four_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="04:00-05:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_text"
        app:layout_constraintTop_toTopOf="@id/sixteen_hour_text" />

    <RadioGroup
        android:id="@+id/four_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/four_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/four_hour_text"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_group"
        app:layout_constraintTop_toTopOf="@+id/four_hour_text">

        <RadioButton
            android:id="@+id/four_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/four_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/five_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="05:00-06:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_text"
        app:layout_constraintTop_toTopOf="@id/seventeen_hour_text" />

    <RadioGroup
        android:id="@+id/five_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/five_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/five_hour_text"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_group"
        app:layout_constraintTop_toTopOf="@+id/five_hour_text">

```

```
        <RadioButton
            android:id="@+id/five_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/five_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/six_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="06:00-07:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_text"
        app:layout_constraintTop_toTopOf="@id/eighteen_hour_text" />

    <RadioGroup
        android:id="@+id/six_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/six_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/six_hour_text"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_group"
        app:layout_constraintTop_toTopOf="@+id/six_hour_text">

        <RadioButton
            android:id="@+id/six_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/six_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/seven_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="07:00-08:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_text"
        app:layout_constraintTop_toTopOf="@id/nineteen_hour_text" />
```

```
824
825     <RadioGroup
826         android:id="@+id/seven_hour_group"
827         android:layout_width="wrap_content"
828         android:layout_height="wrap_content"
829         android:checkedButton="@id/seven_button_T1"
830         android:orientation="horizontal"
831         app:layout_constraintBottom_toBottomOf="@+id/seven_hour_text"
832         app:layout_constraintStart_toStartOf="@+id/twenty_hour_group"
833         app:layout_constraintTop_toTopOf="@+id/seven_hour_text">
834
835         <RadioButton
836             android:id="@+id/seven_button_T1"
837             android:layout_width="wrap_content"
838             android:layout_height="wrap_content"
839             android:text="T1" />
840
841         <RadioButton
842             android:id="@+id/seven_button_T2"
843             android:layout_width="wrap_content"
844             android:layout_height="wrap_content"
845             android:text="T2" />
846     </RadioGroup>
847
848 </androidx.constraintlayout.widget.ConstraintLayout>
```

```xml
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".WednesdayProgramActivity">

    <TextView
        android:id="@+id/day_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="MIÉRCOLES"
        android:textSize="22sp"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintTop_toBottomOf="@id/nineteen_hour_group" />

    <Button
        android:id="@+id/save_button"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Guardar"
        app:layout_constraintStart_toStartOf="@+id/nineteen_hour_group"
        app:layout_constraintTop_toBottomOf="@+id/day_text" />

    <Button
        android:id="@+id/update_button"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginStart="12dp"
        android:text="Ver actual"
        app:layout_constraintStart_toEndOf="@+id/save_button"
        app:layout_constraintTop_toTopOf="@id/save_button" />

    <View
        android:id="@+id/divider_h_up"
        android:layout_width="0dp"
        android:layout_height="2dp"
        android:layout_marginTop="-3dp"
        android:background="?attr/colorPrimaryVariant"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="@+id/eight_hour_text" />

    <View
        android:id="@+id/divider_v"
        android:layout_width="2dp"
        android:layout_height="0dp"
        android:layout_marginStart="5dp"
```

```
        android:background="?attr/colorPrimaryVariant"
        app:layout_constraintBottom_toBottomOf="@+id/divider_h_down"
        app:layout_constraintStart_toEndOf="@+id/eight_hour_group"
        app:layout_constraintTop_toTopOf="@id/divider_h_up" />

    <View
        android:id="@+id/divider_h_down"
        android:layout_width="0dp"
        android:layout_height="2dp"
        android:layout_marginTop="2dp"
        android:background="?attr/colorPrimaryVariant"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@id/nineteen_hour_text"
/>

    <TextView
        android:id="@+id/eight_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginStart="6dp"
        android:layout_marginTop="14dp"
        android:text="08:00-09:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

    <RadioGroup
        android:id="@+id/eight_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/eight_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/eight_hour_text"
        app:layout_constraintStart_toEndOf="@+id/eight_hour_text"
        app:layout_constraintTop_toTopOf="@+id/eight_hour_text">

        <RadioButton
            android:id="@+id/eight_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/eight_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/nine_hour_text"
        android:layout_width="wrap_content"
```

```
        android:layout_height="wrap_content"
        android:layout_marginTop="30dp"
        android:text="09:00-10:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@id/eight_hour_text"
        app:layout_constraintTop_toBottomOf="@id/eight_hour_text" />

    <RadioGroup
        android:id="@+id/nine_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/nine_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/nine_hour_text"
        app:layout_constraintStart_toStartOf="@+id/eight_hour_group"
        app:layout_constraintTop_toTopOf="@+id/nine_hour_text">

        <RadioButton
            android:id="@+id/nine_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/nine_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/ten_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="30dp"
        android:text="10:00-11:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@+id/eight_hour_text"
        app:layout_constraintTop_toBottomOf="@id/nine_hour_text" />

    <RadioGroup
        android:id="@+id/ten_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/ten_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/ten_hour_text"
        app:layout_constraintStart_toStartOf="@+id/eight_hour_group"
        app:layout_constraintTop_toTopOf="@+id/ten_hour_text">

        <RadioButton
            android:id="@+id/ten_button_T1"
            android:layout_width="wrap_content"
```

```
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/ten_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/eleven_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="30dp"
        android:text="11:00-12:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@id/eight_hour_text"
        app:layout_constraintTop_toBottomOf="@id/ten_hour_text" />

    <RadioGroup
        android:id="@+id/eleven_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/eleven_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/eleven_hour_text
"
        app:layout_constraintStart_toStartOf="@+id/eight_hour_group"
        app:layout_constraintTop_toTopOf="@+id/eleven_hour_text">

        <RadioButton
            android:id="@+id/eleven_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/eleven_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/twelve_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="30dp"
        android:text="12:00-13:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@id/eight_hour_text"
        app:layout_constraintTop_toBottomOf="@id/eleven_hour_text" />
```

```
208
209     <RadioGroup
210         android:id="@+id/twelve_hour_group"
211         android:layout_width="wrap_content"
212         android:layout_height="wrap_content"
213         android:checkedButton="@id/twelve_button_T1"
214         android:orientation="horizontal"
215         app:layout_constraintBottom_toBottomOf="@+id/twelve_hour_text
    "
216         app:layout_constraintStart_toStartOf="@+id/eight_hour_group"
217         app:layout_constraintTop_toTopOf="@+id/twelve_hour_text">
218
219         <RadioButton
220             android:id="@+id/twelve_button_T1"
221             android:layout_width="wrap_content"
222             android:layout_height="wrap_content"
223             android:text="T1" />
224
225         <RadioButton
226             android:id="@+id/twelve_button_T2"
227             android:layout_width="wrap_content"
228             android:layout_height="wrap_content"
229             android:text="T2" />
230     </RadioGroup>
231
232     <TextView
233         android:id="@+id/thirteen_hour_text"
234         android:layout_width="wrap_content"
235         android:layout_height="wrap_content"
236         android:layout_marginTop="30dp"
237         android:text="13:00-14:00"
238         android:textSize="16sp"
239         app:layout_constraintStart_toStartOf="@id/eight_hour_text"
240         app:layout_constraintTop_toBottomOf="@id/twelve_hour_text" />
241
242     <RadioGroup
243         android:id="@+id/thirteen_hour_group"
244         android:layout_width="wrap_content"
245         android:layout_height="wrap_content"
246         android:checkedButton="@id/thirteen_button_T1"
247         android:orientation="horizontal"
248         app:layout_constraintBottom_toBottomOf="@+id/
    thirteen_hour_text"
249         app:layout_constraintStart_toStartOf="@+id/eight_hour_group"
250         app:layout_constraintTop_toTopOf="@+id/thirteen_hour_text">
251
252         <RadioButton
253             android:id="@+id/thirteen_button_T1"
254             android:layout_width="wrap_content"
255             android:layout_height="wrap_content"
256             android:text="T1" />
257
258         <RadioButton
```

```
            android:id="@+id/thirteen_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/fourteen_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="30dp"
        android:text="14:00-15:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@id/eight_hour_text"
        app:layout_constraintTop_toBottomOf="@id/thirteen_hour_text"
    />

    <RadioGroup
        android:id="@+id/fourteen_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/fourteen_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/
    fourteen_hour_text"
        app:layout_constraintStart_toStartOf="@+id/eight_hour_group"
        app:layout_constraintTop_toTopOf="@+id/fourteen_hour_text">

        <RadioButton
            android:id="@+id/fourteen_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/fourteen_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/fifteen_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="30dp"
        android:text="15:00-16:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@id/eight_hour_text"
        app:layout_constraintTop_toBottomOf="@id/fourteen_hour_text"
    />

    <RadioGroup
```

```
309         android:id="@+id/fifteen_hour_group"
310         android:layout_width="wrap_content"
311         android:layout_height="wrap_content"
312         android:checkedButton="@id/fifteen_button_T1"
313         android:orientation="horizontal"
314         app:layout_constraintBottom_toBottomOf="@+id/
    fifteen_hour_text"
315         app:layout_constraintStart_toStartOf="@+id/eight_hour_group"
316         app:layout_constraintTop_toTopOf="@+id/fifteen_hour_text">
317 
318         <RadioButton
319             android:id="@+id/fifteen_button_T1"
320             android:layout_width="wrap_content"
321             android:layout_height="wrap_content"
322             android:text="T1" />
323 
324         <RadioButton
325             android:id="@+id/fifteen_button_T2"
326             android:layout_width="wrap_content"
327             android:layout_height="wrap_content"
328             android:text="T2" />
329     </RadioGroup>
330 
331     <TextView
332         android:id="@+id/sixteen_hour_text"
333         android:layout_width="wrap_content"
334         android:layout_height="wrap_content"
335         android:layout_marginTop="30dp"
336         android:text="16:00-17:00"
337         android:textSize="16sp"
338         app:layout_constraintStart_toStartOf="@id/eight_hour_text"
339         app:layout_constraintTop_toBottomOf="@id/fifteen_hour_text"
     />
340 
341     <RadioGroup
342         android:id="@+id/sixteen_hour_group"
343         android:layout_width="wrap_content"
344         android:layout_height="wrap_content"
345         android:checkedButton="@id/sixteen_button_T1"
346         android:orientation="horizontal"
347         app:layout_constraintBottom_toBottomOf="@+id/
    sixteen_hour_text"
348         app:layout_constraintStart_toStartOf="@+id/eight_hour_group"
349         app:layout_constraintTop_toTopOf="@+id/sixteen_hour_text">
350 
351         <RadioButton
352             android:id="@+id/sixteen_button_T1"
353             android:layout_width="wrap_content"
354             android:layout_height="wrap_content"
355             android:text="T1" />
356 
357         <RadioButton
358             android:id="@+id/sixteen_button_T2"
```

```
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/seventeen_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="30dp"
        android:text="17:00-18:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@id/eight_hour_text"
        app:layout_constraintTop_toBottomOf="@id/sixteen_hour_text"
    />

    <RadioGroup
        android:id="@+id/seventeen_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/seventeen_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/
seventeen_hour_text"
        app:layout_constraintStart_toStartOf="@+id/eight_hour_group"
        app:layout_constraintTop_toTopOf="@+id/seventeen_hour_text">

        <RadioButton
            android:id="@+id/seventeen_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/seventeen_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/eighteen_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="30dp"
        android:text="18:00-19:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@id/eight_hour_text"
        app:layout_constraintTop_toBottomOf="@id/seventeen_hour_text"
    />

    <RadioGroup
        android:id="@+id/eighteen_hour_group"
```

```
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/eighteen_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/
eighteen_hour_text"
        app:layout_constraintStart_toStartOf="@+id/eight_hour_group"
        app:layout_constraintTop_toTopOf="@+id/eighteen_hour_text">

        <RadioButton
            android:id="@+id/eighteen_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/eighteen_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/nineteen_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="30dp"
        android:text="19:00-20:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@id/eight_hour_text"
        app:layout_constraintTop_toBottomOf="@id/eighteen_hour_text"
 />

    <RadioGroup
        android:id="@+id/nineteen_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/nineteen_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/
nineteen_hour_text"
        app:layout_constraintStart_toStartOf="@+id/eight_hour_group"
        app:layout_constraintTop_toTopOf="@+id/nineteen_hour_text">

        <RadioButton
            android:id="@+id/nineteen_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/nineteen_button_T2"
            android:layout_width="wrap_content"
```

```
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/twenty_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginStart="10dp"
        android:text="20:00-21:00"
        android:textSize="16sp"
        app:layout_constraintStart_toEndOf="@+id/eight_hour_group"
        app:layout_constraintTop_toTopOf="@+id/eight_hour_text" />

    <RadioGroup
        android:id="@+id/twenty_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/twenty_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/twenty_hour_text
"
        app:layout_constraintStart_toEndOf="@+id/twenty_hour_text"
        app:layout_constraintTop_toTopOf="@+id/twenty_hour_text">

        <RadioButton
            android:id="@+id/twenty_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/twenty_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/twenty_one_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="21:00-22:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_text"
        app:layout_constraintTop_toTopOf="@id/nine_hour_text" />

    <RadioGroup
        android:id="@+id/twenty_one_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/twenty_one_button_T1"
        android:orientation="horizontal"
```

```
        app:layout_constraintBottom_toBottomOf="@+id/
    twenty_one_hour_text"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_group"
        app:layout_constraintTop_toTopOf="@+id/twenty_one_hour_text">

        <RadioButton
            android:id="@+id/twenty_one_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/twenty_one_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/twenty_two_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="22:00-23:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_text"
        app:layout_constraintTop_toTopOf="@id/ten_hour_text" />

    <RadioGroup
        android:id="@+id/twenty_two_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/twenty_two_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/
    twenty_two_hour_text"
        app:layout_constraintStart_toEndOf="@+id/twenty_two_hour_text
    "
        app:layout_constraintTop_toTopOf="@+id/twenty_two_hour_text">

        <RadioButton
            android:id="@+id/twenty_two_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/twenty_two_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
```

```
        android:id="@+id/twenty_three_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="23:00-00:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_text"
        app:layout_constraintTop_toTopOf="@id/eleven_hour_text" />

    <RadioGroup
        android:id="@+id/twenty_three_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/twenty_three_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/
twenty_three_hour_text"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_group"
        app:layout_constraintTop_toTopOf="@+id/twenty_three_hour_text
">

        <RadioButton
            android:id="@+id/twenty_three_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/twenty_three_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/zero_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="00:00-01:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_text"
        app:layout_constraintTop_toTopOf="@id/twelve_hour_text" />

    <RadioGroup
        android:id="@+id/zero_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/zero_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/zero_hour_text"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_group"
        app:layout_constraintTop_toTopOf="@+id/zero_hour_text">

        <RadioButton
```

```
            android:id="@+id/zero_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/zero_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/one_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="01:00-02:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_text"
        app:layout_constraintTop_toTopOf="@id/thirteen_hour_text" />

    <RadioGroup
        android:id="@+id/one_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/one_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/one_hour_text"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_group"
        app:layout_constraintTop_toTopOf="@+id/one_hour_text">

        <RadioButton
            android:id="@+id/one_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/one_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/two_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="02:00-03:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_text"
        app:layout_constraintTop_toTopOf="@id/fourteen_hour_text" />

```

```
    <RadioGroup
        android:id="@+id/two_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/two_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/two_hour_text"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_group"
        app:layout_constraintTop_toTopOf="@+id/two_hour_text">

        <RadioButton
            android:id="@+id/two_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/two_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/three_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="03:00-04:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_text"
        app:layout_constraintTop_toTopOf="@id/fifteen_hour_text" />

    <RadioGroup
        android:id="@+id/three_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/three_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/three_hour_text"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_group"
        app:layout_constraintTop_toTopOf="@+id/three_hour_text">

        <RadioButton
            android:id="@+id/three_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/three_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
```

```
    </RadioGroup>

    <TextView
        android:id="@+id/four_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="04:00-05:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_text"
        app:layout_constraintTop_toTopOf="@id/sixteen_hour_text" />

    <RadioGroup
        android:id="@+id/four_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/four_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/four_hour_text"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_group"
        app:layout_constraintTop_toTopOf="@+id/four_hour_text">

        <RadioButton
            android:id="@+id/four_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/four_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/five_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="05:00-06:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_text"
        app:layout_constraintTop_toTopOf="@id/seventeen_hour_text" />

    <RadioGroup
        android:id="@+id/five_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/five_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/five_hour_text"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_group"
        app:layout_constraintTop_toTopOf="@+id/five_hour_text">

```

```
        <RadioButton
            android:id="@+id/five_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/five_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/six_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="06:00-07:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_text"
        app:layout_constraintTop_toTopOf="@id/eighteen_hour_text" />

    <RadioGroup
        android:id="@+id/six_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/six_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/six_hour_text"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_group"
        app:layout_constraintTop_toTopOf="@+id/six_hour_text">

        <RadioButton
            android:id="@+id/six_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/six_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/seven_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="07:00-08:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_text"
        app:layout_constraintTop_toTopOf="@id/nineteen_hour_text" />
```

```
824
825     <RadioGroup
826         android:id="@+id/seven_hour_group"
827         android:layout_width="wrap_content"
828         android:layout_height="wrap_content"
829         android:checkedButton="@id/seven_button_T1"
830         android:orientation="horizontal"
831         app:layout_constraintBottom_toBottomOf="@+id/seven_hour_text"
832         app:layout_constraintStart_toStartOf="@+id/twenty_hour_group"
833         app:layout_constraintTop_toTopOf="@+id/seven_hour_text">
834
835         <RadioButton
836             android:id="@+id/seven_button_T1"
837             android:layout_width="wrap_content"
838             android:layout_height="wrap_content"
839             android:text="T1" />
840
841         <RadioButton
842             android:id="@+id/seven_button_T2"
843             android:layout_width="wrap_content"
844             android:layout_height="wrap_content"
845             android:text="T2" />
846     </RadioGroup>
847
848 </androidx.constraintlayout.widget.ConstraintLayout>
```

```xml
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".ThursdayProgramActivity">

    <TextView
        android:id="@+id/day_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="JUEVES"
        android:textSize="22sp"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintTop_toBottomOf="@id/nineteen_hour_group" />

    <Button
        android:id="@+id/save_button"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Guardar"
        app:layout_constraintStart_toStartOf="@+id/nineteen_hour_group"
        app:layout_constraintTop_toBottomOf="@+id/day_text" />

    <Button
        android:id="@+id/update_button"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginStart="12dp"
        android:text="Ver actual"
        app:layout_constraintStart_toEndOf="@+id/save_button"
        app:layout_constraintTop_toTopOf="@id/save_button" />

    <View
        android:id="@+id/divider_h_up"
        android:layout_width="0dp"
        android:layout_height="2dp"
        android:layout_marginTop="-3dp"
        android:background="?attr/colorPrimaryVariant"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="@+id/eight_hour_text" />

    <View
        android:id="@+id/divider_v"
        android:layout_width="2dp"
        android:layout_height="0dp"
        android:layout_marginStart="5dp"
```

```
        android:background="?attr/colorPrimaryVariant"
        app:layout_constraintBottom_toBottomOf="@+id/divider_h_down"
        app:layout_constraintStart_toEndOf="@+id/eight_hour_group"
        app:layout_constraintTop_toTopOf="@id/divider_h_up" />

    <View
        android:id="@+id/divider_h_down"
        android:layout_width="0dp"
        android:layout_height="2dp"
        android:layout_marginTop="2dp"
        android:background="?attr/colorPrimaryVariant"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@id/nineteen_hour_text"
/>

    <TextView
        android:id="@+id/eight_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginStart="6dp"
        android:layout_marginTop="14dp"
        android:text="08:00-09:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

    <RadioGroup
        android:id="@+id/eight_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/eight_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/eight_hour_text"
        app:layout_constraintStart_toEndOf="@+id/eight_hour_text"
        app:layout_constraintTop_toTopOf="@+id/eight_hour_text">

        <RadioButton
            android:id="@+id/eight_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/eight_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/nine_hour_text"
        android:layout_width="wrap_content"
```

```
        android:layout_height="wrap_content"
        android:layout_marginTop="30dp"
        android:text="09:00-10:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@id/eight_hour_text"
        app:layout_constraintTop_toBottomOf="@id/eight_hour_text" />

    <RadioGroup
        android:id="@+id/nine_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/nine_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/nine_hour_text"
        app:layout_constraintStart_toStartOf="@+id/eight_hour_group"
        app:layout_constraintTop_toTopOf="@+id/nine_hour_text">

        <RadioButton
            android:id="@+id/nine_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/nine_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/ten_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="30dp"
        android:text="10:00-11:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@+id/eight_hour_text"
        app:layout_constraintTop_toBottomOf="@id/nine_hour_text" />

    <RadioGroup
        android:id="@+id/ten_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/ten_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/ten_hour_text"
        app:layout_constraintStart_toStartOf="@+id/eight_hour_group"
        app:layout_constraintTop_toTopOf="@+id/ten_hour_text">

        <RadioButton
            android:id="@+id/ten_button_T1"
            android:layout_width="wrap_content"
```

```
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/ten_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/eleven_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="30dp"
        android:text="11:00-12:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@id/eight_hour_text"
        app:layout_constraintTop_toBottomOf="@id/ten_hour_text" />

    <RadioGroup
        android:id="@+id/eleven_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/eleven_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/eleven_hour_text
"
        app:layout_constraintStart_toStartOf="@+id/eight_hour_group"
        app:layout_constraintTop_toTopOf="@+id/eleven_hour_text">

        <RadioButton
            android:id="@+id/eleven_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/eleven_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/twelve_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="30dp"
        android:text="12:00-13:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@id/eight_hour_text"
        app:layout_constraintTop_toBottomOf="@id/eleven_hour_text" />
```

```
208
209     <RadioGroup
210         android:id="@+id/twelve_hour_group"
211         android:layout_width="wrap_content"
212         android:layout_height="wrap_content"
213         android:checkedButton="@id/twelve_button_T1"
214         android:orientation="horizontal"
215         app:layout_constraintBottom_toBottomOf="@+id/twelve_hour_text
    "
216         app:layout_constraintStart_toStartOf="@+id/eight_hour_group"
217         app:layout_constraintTop_toTopOf="@+id/twelve_hour_text">
218
219         <RadioButton
220             android:id="@+id/twelve_button_T1"
221             android:layout_width="wrap_content"
222             android:layout_height="wrap_content"
223             android:text="T1" />
224
225         <RadioButton
226             android:id="@+id/twelve_button_T2"
227             android:layout_width="wrap_content"
228             android:layout_height="wrap_content"
229             android:text="T2" />
230     </RadioGroup>
231
232     <TextView
233         android:id="@+id/thirteen_hour_text"
234         android:layout_width="wrap_content"
235         android:layout_height="wrap_content"
236         android:layout_marginTop="30dp"
237         android:text="13:00-14:00"
238         android:textSize="16sp"
239         app:layout_constraintStart_toStartOf="@id/eight_hour_text"
240         app:layout_constraintTop_toBottomOf="@id/twelve_hour_text" />
241
242     <RadioGroup
243         android:id="@+id/thirteen_hour_group"
244         android:layout_width="wrap_content"
245         android:layout_height="wrap_content"
246         android:checkedButton="@id/thirteen_button_T1"
247         android:orientation="horizontal"
248         app:layout_constraintBottom_toBottomOf="@+id/
    thirteen_hour_text"
249         app:layout_constraintStart_toStartOf="@+id/eight_hour_group"
250         app:layout_constraintTop_toTopOf="@+id/thirteen_hour_text">
251
252         <RadioButton
253             android:id="@+id/thirteen_button_T1"
254             android:layout_width="wrap_content"
255             android:layout_height="wrap_content"
256             android:text="T1" />
257
258         <RadioButton
```

```
            android:id="@+id/thirteen_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/fourteen_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="30dp"
        android:text="14:00-15:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@id/eight_hour_text"
        app:layout_constraintTop_toBottomOf="@id/thirteen_hour_text"
    />

    <RadioGroup
        android:id="@+id/fourteen_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/fourteen_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/
    fourteen_hour_text"
        app:layout_constraintStart_toStartOf="@+id/eight_hour_group"
        app:layout_constraintTop_toTopOf="@+id/fourteen_hour_text">

        <RadioButton
            android:id="@+id/fourteen_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/fourteen_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/fifteen_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="30dp"
        android:text="15:00-16:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@id/eight_hour_text"
        app:layout_constraintTop_toBottomOf="@id/fourteen_hour_text"
    />

    <RadioGroup
```

```
309         android:id="@+id/fifteen_hour_group"
310         android:layout_width="wrap_content"
311         android:layout_height="wrap_content"
312         android:checkedButton="@id/fifteen_button_T1"
313         android:orientation="horizontal"
314         app:layout_constraintBottom_toBottomOf="@+id/
    fifteen_hour_text"
315         app:layout_constraintStart_toStartOf="@+id/eight_hour_group"
316         app:layout_constraintTop_toTopOf="@+id/fifteen_hour_text">
317 
318         <RadioButton
319             android:id="@+id/fifteen_button_T1"
320             android:layout_width="wrap_content"
321             android:layout_height="wrap_content"
322             android:text="T1" />
323 
324         <RadioButton
325             android:id="@+id/fifteen_button_T2"
326             android:layout_width="wrap_content"
327             android:layout_height="wrap_content"
328             android:text="T2" />
329     </RadioGroup>
330 
331     <TextView
332         android:id="@+id/sixteen_hour_text"
333         android:layout_width="wrap_content"
334         android:layout_height="wrap_content"
335         android:layout_marginTop="30dp"
336         android:text="16:00-17:00"
337         android:textSize="16sp"
338         app:layout_constraintStart_toStartOf="@id/eight_hour_text"
339         app:layout_constraintTop_toBottomOf="@id/fifteen_hour_text"
     />
340 
341     <RadioGroup
342         android:id="@+id/sixteen_hour_group"
343         android:layout_width="wrap_content"
344         android:layout_height="wrap_content"
345         android:checkedButton="@id/sixteen_button_T1"
346         android:orientation="horizontal"
347         app:layout_constraintBottom_toBottomOf="@+id/
    sixteen_hour_text"
348         app:layout_constraintStart_toStartOf="@+id/eight_hour_group"
349         app:layout_constraintTop_toTopOf="@+id/sixteen_hour_text">
350 
351         <RadioButton
352             android:id="@+id/sixteen_button_T1"
353             android:layout_width="wrap_content"
354             android:layout_height="wrap_content"
355             android:text="T1" />
356 
357         <RadioButton
358             android:id="@+id/sixteen_button_T2"
```

```
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/seventeen_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="30dp"
        android:text="17:00-18:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@id/eight_hour_text"
        app:layout_constraintTop_toBottomOf="@id/sixteen_hour_text"
    />

    <RadioGroup
        android:id="@+id/seventeen_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/seventeen_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/
seventeen_hour_text"
        app:layout_constraintStart_toStartOf="@+id/eight_hour_group"
        app:layout_constraintTop_toTopOf="@+id/seventeen_hour_text">

        <RadioButton
            android:id="@+id/seventeen_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/seventeen_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/eighteen_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="30dp"
        android:text="18:00-19:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@id/eight_hour_text"
        app:layout_constraintTop_toBottomOf="@id/seventeen_hour_text"
    />

    <RadioGroup
        android:id="@+id/eighteen_hour_group"
```

```
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/eighteen_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/
eighteen_hour_text"
        app:layout_constraintStart_toStartOf="@+id/eight_hour_group"
        app:layout_constraintTop_toTopOf="@+id/eighteen_hour_text">

        <RadioButton
            android:id="@+id/eighteen_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/eighteen_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/nineteen_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="30dp"
        android:text="19:00-20:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@id/eight_hour_text"
        app:layout_constraintTop_toBottomOf="@id/eighteen_hour_text"
 />

    <RadioGroup
        android:id="@+id/nineteen_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/nineteen_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/
nineteen_hour_text"
        app:layout_constraintStart_toStartOf="@+id/eight_hour_group"
        app:layout_constraintTop_toTopOf="@+id/nineteen_hour_text">

        <RadioButton
            android:id="@+id/nineteen_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/nineteen_button_T2"
            android:layout_width="wrap_content"
```

```
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/twenty_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginStart="10dp"
        android:text="20:00-21:00"
        android:textSize="16sp"
        app:layout_constraintStart_toEndOf="@+id/eight_hour_group"
        app:layout_constraintTop_toTopOf="@+id/eight_hour_text" />

    <RadioGroup
        android:id="@+id/twenty_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/twenty_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/twenty_hour_text
"
        app:layout_constraintStart_toEndOf="@+id/twenty_hour_text"
        app:layout_constraintTop_toTopOf="@+id/twenty_hour_text">

        <RadioButton
            android:id="@+id/twenty_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/twenty_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/twenty_one_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="21:00-22:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_text"
        app:layout_constraintTop_toTopOf="@id/nine_hour_text" />

    <RadioGroup
        android:id="@+id/twenty_one_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/twenty_one_button_T1"
        android:orientation="horizontal"
```

```
511         app:layout_constraintBottom_toBottomOf="@+id/
    twenty_one_hour_text"
512         app:layout_constraintStart_toStartOf="@+id/twenty_hour_group"
513         app:layout_constraintTop_toTopOf="@+id/twenty_one_hour_text">
514 
515         <RadioButton
516             android:id="@+id/twenty_one_button_T1"
517             android:layout_width="wrap_content"
518             android:layout_height="wrap_content"
519             android:text="T1" />
520 
521         <RadioButton
522             android:id="@+id/twenty_one_button_T2"
523             android:layout_width="wrap_content"
524             android:layout_height="wrap_content"
525             android:text="T2" />
526     </RadioGroup>
527 
528     <TextView
529         android:id="@+id/twenty_two_hour_text"
530         android:layout_width="wrap_content"
531         android:layout_height="wrap_content"
532         android:text="22:00-23:00"
533         android:textSize="16sp"
534         app:layout_constraintStart_toStartOf="@+id/twenty_hour_text"
535         app:layout_constraintTop_toTopOf="@id/ten_hour_text" />
536 
537     <RadioGroup
538         android:id="@+id/twenty_two_hour_group"
539         android:layout_width="wrap_content"
540         android:layout_height="wrap_content"
541         android:checkedButton="@id/twenty_two_button_T1"
542         android:orientation="horizontal"
543         app:layout_constraintBottom_toBottomOf="@+id/
    twenty_two_hour_text"
544         app:layout_constraintStart_toEndOf="@+id/twenty_two_hour_text
    "
545         app:layout_constraintTop_toTopOf="@+id/twenty_two_hour_text">
546 
547         <RadioButton
548             android:id="@+id/twenty_two_button_T1"
549             android:layout_width="wrap_content"
550             android:layout_height="wrap_content"
551             android:text="T1" />
552 
553         <RadioButton
554             android:id="@+id/twenty_two_button_T2"
555             android:layout_width="wrap_content"
556             android:layout_height="wrap_content"
557             android:text="T2" />
558     </RadioGroup>
559 
560     <TextView
```

```
        android:id="@+id/twenty_three_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="23:00-00:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_text"
        app:layout_constraintTop_toTopOf="@id/eleven_hour_text" />

    <RadioGroup
        android:id="@+id/twenty_three_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/twenty_three_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/
twenty_three_hour_text"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_group"
        app:layout_constraintTop_toTopOf="@+id/twenty_three_hour_text
">

        <RadioButton
            android:id="@+id/twenty_three_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/twenty_three_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/zero_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="00:00-01:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_text"
        app:layout_constraintTop_toTopOf="@id/twelve_hour_text" />

    <RadioGroup
        android:id="@+id/zero_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/zero_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/zero_hour_text"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_group"
        app:layout_constraintTop_toTopOf="@+id/zero_hour_text">

        <RadioButton
```

```
            android:id="@+id/zero_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/zero_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/one_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="01:00-02:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_text"
        app:layout_constraintTop_toTopOf="@id/thirteen_hour_text" />

    <RadioGroup
        android:id="@+id/one_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/one_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/one_hour_text"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_group"
        app:layout_constraintTop_toTopOf="@+id/one_hour_text">

        <RadioButton
            android:id="@+id/one_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/one_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/two_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="02:00-03:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_text"
        app:layout_constraintTop_toTopOf="@id/fourteen_hour_text" />

```

```
    <RadioGroup
        android:id="@+id/two_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/two_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/two_hour_text"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_group"
        app:layout_constraintTop_toTopOf="@+id/two_hour_text">

        <RadioButton
            android:id="@+id/two_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/two_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/three_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="03:00-04:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_text"
        app:layout_constraintTop_toTopOf="@id/fifteen_hour_text" />

    <RadioGroup
        android:id="@+id/three_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/three_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/three_hour_text"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_group"
        app:layout_constraintTop_toTopOf="@+id/three_hour_text">

        <RadioButton
            android:id="@+id/three_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/three_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
```

```
    </RadioGroup>

    <TextView
        android:id="@+id/four_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="04:00-05:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_text"
        app:layout_constraintTop_toTopOf="@id/sixteen_hour_text" />

    <RadioGroup
        android:id="@+id/four_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/four_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/four_hour_text"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_group"
        app:layout_constraintTop_toTopOf="@+id/four_hour_text">

        <RadioButton
            android:id="@+id/four_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/four_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/five_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="05:00-06:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_text"
        app:layout_constraintTop_toTopOf="@id/seventeen_hour_text" />

    <RadioGroup
        android:id="@+id/five_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/five_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/five_hour_text"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_group"
        app:layout_constraintTop_toTopOf="@+id/five_hour_text">

```

```
        <RadioButton
            android:id="@+id/five_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/five_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/six_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="06:00-07:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_text"
        app:layout_constraintTop_toTopOf="@id/eighteen_hour_text" />

    <RadioGroup
        android:id="@+id/six_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/six_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/six_hour_text"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_group"
        app:layout_constraintTop_toTopOf="@+id/six_hour_text">

        <RadioButton
            android:id="@+id/six_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/six_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/seven_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="07:00-08:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_text"
        app:layout_constraintTop_toTopOf="@id/nineteen_hour_text" />
```

```
824
825     <RadioGroup
826         android:id="@+id/seven_hour_group"
827         android:layout_width="wrap_content"
828         android:layout_height="wrap_content"
829         android:checkedButton="@id/seven_button_T1"
830         android:orientation="horizontal"
831         app:layout_constraintBottom_toBottomOf="@+id/seven_hour_text"
832         app:layout_constraintStart_toStartOf="@+id/twenty_hour_group"
833         app:layout_constraintTop_toTopOf="@+id/seven_hour_text">
834
835         <RadioButton
836             android:id="@+id/seven_button_T1"
837             android:layout_width="wrap_content"
838             android:layout_height="wrap_content"
839             android:text="T1" />
840
841         <RadioButton
842             android:id="@+id/seven_button_T2"
843             android:layout_width="wrap_content"
844             android:layout_height="wrap_content"
845             android:text="T2" />
846     </RadioGroup>
847
848 </androidx.constraintlayout.widget.ConstraintLayout>
```

```xml
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".FridayProgramActivity">

    <TextView
        android:id="@+id/day_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="VIERNES"
        android:textSize="22sp"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintTop_toBottomOf="@id/nineteen_hour_group" />

    <Button
        android:id="@+id/save_button"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Guardar"
        app:layout_constraintStart_toStartOf="@+id/nineteen_hour_group"
        app:layout_constraintTop_toBottomOf="@+id/day_text" />

    <Button
        android:id="@+id/update_button"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginStart="12dp"
        android:text="Ver actual"
        app:layout_constraintStart_toEndOf="@+id/save_button"
        app:layout_constraintTop_toTopOf="@id/save_button" />

    <View
        android:id="@+id/divider_h_up"
        android:layout_width="0dp"
        android:layout_height="2dp"
        android:layout_marginTop="-3dp"
        android:background="?attr/colorPrimaryVariant"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="@+id/eight_hour_text" />

    <View
        android:id="@+id/divider_v"
        android:layout_width="2dp"
        android:layout_height="0dp"
        android:layout_marginStart="5dp"
```

```
        android:background="?attr/colorPrimaryVariant"
        app:layout_constraintBottom_toBottomOf="@+id/divider_h_down"
        app:layout_constraintStart_toEndOf="@+id/eight_hour_group"
        app:layout_constraintTop_toTopOf="@id/divider_h_up" />

    <View
        android:id="@+id/divider_h_down"
        android:layout_width="0dp"
        android:layout_height="2dp"
        android:layout_marginTop="2dp"
        android:background="?attr/colorPrimaryVariant"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@id/nineteen_hour_text"
/>

    <TextView
        android:id="@+id/eight_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginStart="6dp"
        android:layout_marginTop="14dp"
        android:text="08:00-09:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

    <RadioGroup
        android:id="@+id/eight_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/eight_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/eight_hour_text"
        app:layout_constraintStart_toEndOf="@+id/eight_hour_text"
        app:layout_constraintTop_toTopOf="@+id/eight_hour_text">

        <RadioButton
            android:id="@+id/eight_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/eight_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/nine_hour_text"
        android:layout_width="wrap_content"
```

```
        android:layout_height="wrap_content"
        android:layout_marginTop="30dp"
        android:text="09:00-10:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@id/eight_hour_text"
        app:layout_constraintTop_toBottomOf="@id/eight_hour_text" />

    <RadioGroup
        android:id="@+id/nine_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/nine_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/nine_hour_text"
        app:layout_constraintStart_toStartOf="@+id/eight_hour_group"
        app:layout_constraintTop_toTopOf="@+id/nine_hour_text">

        <RadioButton
            android:id="@+id/nine_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/nine_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/ten_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="30dp"
        android:text="10:00-11:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@+id/eight_hour_text"
        app:layout_constraintTop_toBottomOf="@id/nine_hour_text" />

    <RadioGroup
        android:id="@+id/ten_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/ten_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/ten_hour_text"
        app:layout_constraintStart_toStartOf="@+id/eight_hour_group"
        app:layout_constraintTop_toTopOf="@+id/ten_hour_text">

        <RadioButton
            android:id="@+id/ten_button_T1"
            android:layout_width="wrap_content"
```

```
156             android:layout_height="wrap_content"
157             android:text="T1" />
158 
159         <RadioButton
160             android:id="@+id/ten_button_T2"
161             android:layout_width="wrap_content"
162             android:layout_height="wrap_content"
163             android:text="T2" />
164     </RadioGroup>
165 
166     <TextView
167         android:id="@+id/eleven_hour_text"
168         android:layout_width="wrap_content"
169         android:layout_height="wrap_content"
170         android:layout_marginTop="30dp"
171         android:text="11:00-12:00"
172         android:textSize="16sp"
173         app:layout_constraintStart_toStartOf="@id/eight_hour_text"
174         app:layout_constraintTop_toBottomOf="@id/ten_hour_text" />
175 
176     <RadioGroup
177         android:id="@+id/eleven_hour_group"
178         android:layout_width="wrap_content"
179         android:layout_height="wrap_content"
180         android:checkedButton="@id/eleven_button_T1"
181         android:orientation="horizontal"
182         app:layout_constraintBottom_toBottomOf="@+id/eleven_hour_text
    "
183         app:layout_constraintStart_toStartOf="@+id/eight_hour_group"
184         app:layout_constraintTop_toTopOf="@+id/eleven_hour_text">
185 
186         <RadioButton
187             android:id="@+id/eleven_button_T1"
188             android:layout_width="wrap_content"
189             android:layout_height="wrap_content"
190             android:text="T1" />
191 
192         <RadioButton
193             android:id="@+id/eleven_button_T2"
194             android:layout_width="wrap_content"
195             android:layout_height="wrap_content"
196             android:text="T2" />
197     </RadioGroup>
198 
199     <TextView
200         android:id="@+id/twelve_hour_text"
201         android:layout_width="wrap_content"
202         android:layout_height="wrap_content"
203         android:layout_marginTop="30dp"
204         android:text="12:00-13:00"
205         android:textSize="16sp"
206         app:layout_constraintStart_toStartOf="@id/eight_hour_text"
207         app:layout_constraintTop_toBottomOf="@id/eleven_hour_text" />
```

```
208
209     <RadioGroup
210         android:id="@+id/twelve_hour_group"
211         android:layout_width="wrap_content"
212         android:layout_height="wrap_content"
213         android:checkedButton="@id/twelve_button_T1"
214         android:orientation="horizontal"
215         app:layout_constraintBottom_toBottomOf="@+id/twelve_hour_text
    "
216         app:layout_constraintStart_toStartOf="@+id/eight_hour_group"
217         app:layout_constraintTop_toTopOf="@+id/twelve_hour_text">
218
219         <RadioButton
220             android:id="@+id/twelve_button_T1"
221             android:layout_width="wrap_content"
222             android:layout_height="wrap_content"
223             android:text="T1" />
224
225         <RadioButton
226             android:id="@+id/twelve_button_T2"
227             android:layout_width="wrap_content"
228             android:layout_height="wrap_content"
229             android:text="T2" />
230     </RadioGroup>
231
232     <TextView
233         android:id="@+id/thirteen_hour_text"
234         android:layout_width="wrap_content"
235         android:layout_height="wrap_content"
236         android:layout_marginTop="30dp"
237         android:text="13:00-14:00"
238         android:textSize="16sp"
239         app:layout_constraintStart_toStartOf="@id/eight_hour_text"
240         app:layout_constraintTop_toBottomOf="@id/twelve_hour_text" />
241
242     <RadioGroup
243         android:id="@+id/thirteen_hour_group"
244         android:layout_width="wrap_content"
245         android:layout_height="wrap_content"
246         android:checkedButton="@id/thirteen_button_T1"
247         android:orientation="horizontal"
248         app:layout_constraintBottom_toBottomOf="@+id/
    thirteen_hour_text"
249         app:layout_constraintStart_toStartOf="@+id/eight_hour_group"
250         app:layout_constraintTop_toTopOf="@+id/thirteen_hour_text">
251
252         <RadioButton
253             android:id="@+id/thirteen_button_T1"
254             android:layout_width="wrap_content"
255             android:layout_height="wrap_content"
256             android:text="T1" />
257
258         <RadioButton
```

```
            android:id="@+id/thirteen_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/fourteen_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="30dp"
        android:text="14:00-15:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@id/eight_hour_text"
        app:layout_constraintTop_toBottomOf="@id/thirteen_hour_text"
    />

    <RadioGroup
        android:id="@+id/fourteen_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/fourteen_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/
    fourteen_hour_text"
        app:layout_constraintStart_toStartOf="@+id/eight_hour_group"
        app:layout_constraintTop_toTopOf="@+id/fourteen_hour_text">

        <RadioButton
            android:id="@+id/fourteen_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/fourteen_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/fifteen_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="30dp"
        android:text="15:00-16:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@id/eight_hour_text"
        app:layout_constraintTop_toBottomOf="@id/fourteen_hour_text"
    />

    <RadioGroup
```

```
309         android:id="@+id/fifteen_hour_group"
310         android:layout_width="wrap_content"
311         android:layout_height="wrap_content"
312         android:checkedButton="@id/fifteen_button_T1"
313         android:orientation="horizontal"
314         app:layout_constraintBottom_toBottomOf="@+id/
    fifteen_hour_text"
315         app:layout_constraintStart_toStartOf="@+id/eight_hour_group"
316         app:layout_constraintTop_toTopOf="@+id/fifteen_hour_text">
317 
318         <RadioButton
319             android:id="@+id/fifteen_button_T1"
320             android:layout_width="wrap_content"
321             android:layout_height="wrap_content"
322             android:text="T1" />
323 
324         <RadioButton
325             android:id="@+id/fifteen_button_T2"
326             android:layout_width="wrap_content"
327             android:layout_height="wrap_content"
328             android:text="T2" />
329     </RadioGroup>
330 
331     <TextView
332         android:id="@+id/sixteen_hour_text"
333         android:layout_width="wrap_content"
334         android:layout_height="wrap_content"
335         android:layout_marginTop="30dp"
336         android:text="16:00-17:00"
337         android:textSize="16sp"
338         app:layout_constraintStart_toStartOf="@id/eight_hour_text"
339         app:layout_constraintTop_toBottomOf="@id/fifteen_hour_text"
     />
340 
341     <RadioGroup
342         android:id="@+id/sixteen_hour_group"
343         android:layout_width="wrap_content"
344         android:layout_height="wrap_content"
345         android:checkedButton="@id/sixteen_button_T1"
346         android:orientation="horizontal"
347         app:layout_constraintBottom_toBottomOf="@+id/
    sixteen_hour_text"
348         app:layout_constraintStart_toStartOf="@+id/eight_hour_group"
349         app:layout_constraintTop_toTopOf="@+id/sixteen_hour_text">
350 
351         <RadioButton
352             android:id="@+id/sixteen_button_T1"
353             android:layout_width="wrap_content"
354             android:layout_height="wrap_content"
355             android:text="T1" />
356 
357         <RadioButton
358             android:id="@+id/sixteen_button_T2"
```

```
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/seventeen_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="30dp"
        android:text="17:00-18:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@id/eight_hour_text"
        app:layout_constraintTop_toBottomOf="@id/sixteen_hour_text"
    />

    <RadioGroup
        android:id="@+id/seventeen_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/seventeen_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/
seventeen_hour_text"
        app:layout_constraintStart_toStartOf="@+id/eight_hour_group"
        app:layout_constraintTop_toTopOf="@+id/seventeen_hour_text">

        <RadioButton
            android:id="@+id/seventeen_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/seventeen_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/eighteen_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="30dp"
        android:text="18:00-19:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@id/eight_hour_text"
        app:layout_constraintTop_toBottomOf="@id/seventeen_hour_text"
    />

    <RadioGroup
        android:id="@+id/eighteen_hour_group"
```

```
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/eighteen_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/
eighteen_hour_text"
        app:layout_constraintStart_toStartOf="@+id/eight_hour_group"
        app:layout_constraintTop_toTopOf="@+id/eighteen_hour_text">

        <RadioButton
            android:id="@+id/eighteen_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/eighteen_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/nineteen_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="30dp"
        android:text="19:00-20:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@id/eight_hour_text"
        app:layout_constraintTop_toBottomOf="@id/eighteen_hour_text"
 />

    <RadioGroup
        android:id="@+id/nineteen_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/nineteen_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/
nineteen_hour_text"
        app:layout_constraintStart_toStartOf="@+id/eight_hour_group"
        app:layout_constraintTop_toTopOf="@+id/nineteen_hour_text">

        <RadioButton
            android:id="@+id/nineteen_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/nineteen_button_T2"
            android:layout_width="wrap_content"
```

```
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/twenty_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginStart="10dp"
        android:text="20:00-21:00"
        android:textSize="16sp"
        app:layout_constraintStart_toEndOf="@+id/eight_hour_group"
        app:layout_constraintTop_toTopOf="@+id/eight_hour_text" />

    <RadioGroup
        android:id="@+id/twenty_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/twenty_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/twenty_hour_text
"
        app:layout_constraintStart_toEndOf="@+id/twenty_hour_text"
        app:layout_constraintTop_toTopOf="@+id/twenty_hour_text">

        <RadioButton
            android:id="@+id/twenty_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/twenty_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/twenty_one_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="21:00-22:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_text"
        app:layout_constraintTop_toTopOf="@id/nine_hour_text" />

    <RadioGroup
        android:id="@+id/twenty_one_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/twenty_one_button_T1"
        android:orientation="horizontal"
```

```
        app:layout_constraintBottom_toBottomOf="@+id/
    twenty_one_hour_text"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_group"
        app:layout_constraintTop_toTopOf="@+id/twenty_one_hour_text">

        <RadioButton
            android:id="@+id/twenty_one_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/twenty_one_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/twenty_two_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="22:00-23:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_text"
        app:layout_constraintTop_toTopOf="@id/ten_hour_text" />

    <RadioGroup
        android:id="@+id/twenty_two_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/twenty_two_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/
    twenty_two_hour_text"
        app:layout_constraintStart_toEndOf="@+id/twenty_two_hour_text
    "
        app:layout_constraintTop_toTopOf="@+id/twenty_two_hour_text">

        <RadioButton
            android:id="@+id/twenty_two_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/twenty_two_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
```

```
        android:id="@+id/twenty_three_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="23:00-00:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_text"
        app:layout_constraintTop_toTopOf="@id/eleven_hour_text" />

    <RadioGroup
        android:id="@+id/twenty_three_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/twenty_three_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/
twenty_three_hour_text"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_group"
        app:layout_constraintTop_toTopOf="@+id/twenty_three_hour_text
">

        <RadioButton
            android:id="@+id/twenty_three_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/twenty_three_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/zero_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="00:00-01:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_text"
        app:layout_constraintTop_toTopOf="@id/twelve_hour_text" />

    <RadioGroup
        android:id="@+id/zero_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/zero_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/zero_hour_text"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_group"
        app:layout_constraintTop_toTopOf="@+id/zero_hour_text">

        <RadioButton
```

```
            android:id="@+id/zero_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/zero_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/one_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="01:00-02:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_text"
        app:layout_constraintTop_toTopOf="@id/thirteen_hour_text" />

    <RadioGroup
        android:id="@+id/one_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/one_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/one_hour_text"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_group"
        app:layout_constraintTop_toTopOf="@+id/one_hour_text">

        <RadioButton
            android:id="@+id/one_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/one_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/two_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="02:00-03:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_text"
        app:layout_constraintTop_toTopOf="@id/fourteen_hour_text" />

```

```
    <RadioGroup
        android:id="@+id/two_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/two_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/two_hour_text"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_group"
        app:layout_constraintTop_toTopOf="@+id/two_hour_text">

        <RadioButton
            android:id="@+id/two_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/two_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/three_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="03:00-04:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_text"
        app:layout_constraintTop_toTopOf="@id/fifteen_hour_text" />

    <RadioGroup
        android:id="@+id/three_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/three_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/three_hour_text"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_group"
        app:layout_constraintTop_toTopOf="@+id/three_hour_text">

        <RadioButton
            android:id="@+id/three_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/three_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
```

```
    </RadioGroup>

    <TextView
        android:id="@+id/four_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="04:00-05:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_text"
        app:layout_constraintTop_toTopOf="@id/sixteen_hour_text" />

    <RadioGroup
        android:id="@+id/four_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/four_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/four_hour_text"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_group"
        app:layout_constraintTop_toTopOf="@+id/four_hour_text">

        <RadioButton
            android:id="@+id/four_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/four_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/five_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="05:00-06:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_text"
        app:layout_constraintTop_toTopOf="@id/seventeen_hour_text" />

    <RadioGroup
        android:id="@+id/five_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/five_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/five_hour_text"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_group"
        app:layout_constraintTop_toTopOf="@+id/five_hour_text">

```

```
        <RadioButton
            android:id="@+id/five_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/five_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/six_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="06:00-07:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_text"
        app:layout_constraintTop_toTopOf="@id/eighteen_hour_text" />

    <RadioGroup
        android:id="@+id/six_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/six_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/six_hour_text"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_group"
        app:layout_constraintTop_toTopOf="@+id/six_hour_text">

        <RadioButton
            android:id="@+id/six_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/six_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/seven_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="07:00-08:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_text"
        app:layout_constraintTop_toTopOf="@id/nineteen_hour_text" />
```

```
824
825     <RadioGroup
826         android:id="@+id/seven_hour_group"
827         android:layout_width="wrap_content"
828         android:layout_height="wrap_content"
829         android:checkedButton="@id/seven_button_T1"
830         android:orientation="horizontal"
831         app:layout_constraintBottom_toBottomOf="@+id/seven_hour_text"
832         app:layout_constraintStart_toStartOf="@+id/twenty_hour_group"
833         app:layout_constraintTop_toTopOf="@+id/seven_hour_text">
834
835         <RadioButton
836             android:id="@+id/seven_button_T1"
837             android:layout_width="wrap_content"
838             android:layout_height="wrap_content"
839             android:text="T1" />
840
841         <RadioButton
842             android:id="@+id/seven_button_T2"
843             android:layout_width="wrap_content"
844             android:layout_height="wrap_content"
845             android:text="T2" />
846     </RadioGroup>
847
848 </androidx.constraintlayout.widget.ConstraintLayout>
```

```xml
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".SaturdayProgramActivity">

    <TextView
        android:id="@+id/day_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="SÁBADO"
        android:textSize="22sp"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintTop_toBottomOf="@id/nineteen_hour_group" />

    <Button
        android:id="@+id/save_button"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Guardar"
        app:layout_constraintStart_toStartOf="@+id/nineteen_hour_group"
        app:layout_constraintTop_toBottomOf="@+id/day_text" />

    <Button
        android:id="@+id/update_button"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginStart="12dp"
        android:text="Ver actual"
        app:layout_constraintStart_toEndOf="@+id/save_button"
        app:layout_constraintTop_toTopOf="@id/save_button" />

    <View
        android:id="@+id/divider_h_up"
        android:layout_width="0dp"
        android:layout_height="2dp"
        android:layout_marginTop="-3dp"
        android:background="?attr/colorPrimaryVariant"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="@+id/eight_hour_text" />

    <View
        android:id="@+id/divider_v"
        android:layout_width="2dp"
        android:layout_height="0dp"
        android:layout_marginStart="5dp"
```

```
        android:background="?attr/colorPrimaryVariant"
        app:layout_constraintBottom_toBottomOf="@+id/divider_h_down"
        app:layout_constraintStart_toEndOf="@+id/eight_hour_group"
        app:layout_constraintTop_toTopOf="@id/divider_h_up" />

    <View
        android:id="@+id/divider_h_down"
        android:layout_width="0dp"
        android:layout_height="2dp"
        android:layout_marginTop="2dp"
        android:background="?attr/colorPrimaryVariant"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@id/nineteen_hour_text"
/>

    <TextView
        android:id="@+id/eight_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginStart="6dp"
        android:layout_marginTop="14dp"
        android:text="08:00-09:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

    <RadioGroup
        android:id="@+id/eight_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/eight_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/eight_hour_text"
        app:layout_constraintStart_toEndOf="@+id/eight_hour_text"
        app:layout_constraintTop_toTopOf="@+id/eight_hour_text">

        <RadioButton
            android:id="@+id/eight_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/eight_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/nine_hour_text"
        android:layout_width="wrap_content"
```

```
        android:layout_height="wrap_content"
        android:layout_marginTop="30dp"
        android:text="09:00-10:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@id/eight_hour_text"
        app:layout_constraintTop_toBottomOf="@id/eight_hour_text" />

    <RadioGroup
        android:id="@+id/nine_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/nine_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/nine_hour_text"
        app:layout_constraintStart_toStartOf="@+id/eight_hour_group"
        app:layout_constraintTop_toTopOf="@+id/nine_hour_text">

        <RadioButton
            android:id="@+id/nine_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/nine_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/ten_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="30dp"
        android:text="10:00-11:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@+id/eight_hour_text"
        app:layout_constraintTop_toBottomOf="@id/nine_hour_text" />

    <RadioGroup
        android:id="@+id/ten_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/ten_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/ten_hour_text"
        app:layout_constraintStart_toStartOf="@+id/eight_hour_group"
        app:layout_constraintTop_toTopOf="@+id/ten_hour_text">

        <RadioButton
            android:id="@+id/ten_button_T1"
            android:layout_width="wrap_content"
```

```
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/ten_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/eleven_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="30dp"
        android:text="11:00-12:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@id/eight_hour_text"
        app:layout_constraintTop_toBottomOf="@id/ten_hour_text" />

    <RadioGroup
        android:id="@+id/eleven_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/eleven_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/eleven_hour_text
"
        app:layout_constraintStart_toStartOf="@+id/eight_hour_group"
        app:layout_constraintTop_toTopOf="@+id/eleven_hour_text">

        <RadioButton
            android:id="@+id/eleven_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/eleven_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/twelve_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="30dp"
        android:text="12:00-13:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@id/eight_hour_text"
        app:layout_constraintTop_toBottomOf="@id/eleven_hour_text" />
```

```
208
209     <RadioGroup
210         android:id="@+id/twelve_hour_group"
211         android:layout_width="wrap_content"
212         android:layout_height="wrap_content"
213         android:checkedButton="@id/twelve_button_T1"
214         android:orientation="horizontal"
215         app:layout_constraintBottom_toBottomOf="@+id/twelve_hour_text
    "
216         app:layout_constraintStart_toStartOf="@+id/eight_hour_group"
217         app:layout_constraintTop_toTopOf="@+id/twelve_hour_text">
218
219         <RadioButton
220             android:id="@+id/twelve_button_T1"
221             android:layout_width="wrap_content"
222             android:layout_height="wrap_content"
223             android:text="T1" />
224
225         <RadioButton
226             android:id="@+id/twelve_button_T2"
227             android:layout_width="wrap_content"
228             android:layout_height="wrap_content"
229             android:text="T2" />
230     </RadioGroup>
231
232     <TextView
233         android:id="@+id/thirteen_hour_text"
234         android:layout_width="wrap_content"
235         android:layout_height="wrap_content"
236         android:layout_marginTop="30dp"
237         android:text="13:00-14:00"
238         android:textSize="16sp"
239         app:layout_constraintStart_toStartOf="@id/eight_hour_text"
240         app:layout_constraintTop_toBottomOf="@id/twelve_hour_text" />
241
242     <RadioGroup
243         android:id="@+id/thirteen_hour_group"
244         android:layout_width="wrap_content"
245         android:layout_height="wrap_content"
246         android:checkedButton="@id/thirteen_button_T1"
247         android:orientation="horizontal"
248         app:layout_constraintBottom_toBottomOf="@+id/
    thirteen_hour_text"
249         app:layout_constraintStart_toStartOf="@+id/eight_hour_group"
250         app:layout_constraintTop_toTopOf="@+id/thirteen_hour_text">
251
252         <RadioButton
253             android:id="@+id/thirteen_button_T1"
254             android:layout_width="wrap_content"
255             android:layout_height="wrap_content"
256             android:text="T1" />
257
258         <RadioButton
```

```
            android:id="@+id/thirteen_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/fourteen_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="30dp"
        android:text="14:00-15:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@id/eight_hour_text"
        app:layout_constraintTop_toBottomOf="@id/thirteen_hour_text"
    />

    <RadioGroup
        android:id="@+id/fourteen_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/fourteen_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/
    fourteen_hour_text"
        app:layout_constraintStart_toStartOf="@+id/eight_hour_group"
        app:layout_constraintTop_toTopOf="@+id/fourteen_hour_text">

        <RadioButton
            android:id="@+id/fourteen_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/fourteen_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/fifteen_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="30dp"
        android:text="15:00-16:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@id/eight_hour_text"
        app:layout_constraintTop_toBottomOf="@id/fourteen_hour_text"
    />

    <RadioGroup
```

```
309         android:id="@+id/fifteen_hour_group"
310         android:layout_width="wrap_content"
311         android:layout_height="wrap_content"
312         android:checkedButton="@id/fifteen_button_T1"
313         android:orientation="horizontal"
314         app:layout_constraintBottom_toBottomOf="@+id/
    fifteen_hour_text"
315         app:layout_constraintStart_toStartOf="@+id/eight_hour_group"
316         app:layout_constraintTop_toTopOf="@+id/fifteen_hour_text">
317 
318         <RadioButton
319             android:id="@+id/fifteen_button_T1"
320             android:layout_width="wrap_content"
321             android:layout_height="wrap_content"
322             android:text="T1" />
323 
324         <RadioButton
325             android:id="@+id/fifteen_button_T2"
326             android:layout_width="wrap_content"
327             android:layout_height="wrap_content"
328             android:text="T2" />
329     </RadioGroup>
330 
331     <TextView
332         android:id="@+id/sixteen_hour_text"
333         android:layout_width="wrap_content"
334         android:layout_height="wrap_content"
335         android:layout_marginTop="30dp"
336         android:text="16:00-17:00"
337         android:textSize="16sp"
338         app:layout_constraintStart_toStartOf="@id/eight_hour_text"
339         app:layout_constraintTop_toBottomOf="@id/fifteen_hour_text"
     />
340 
341     <RadioGroup
342         android:id="@+id/sixteen_hour_group"
343         android:layout_width="wrap_content"
344         android:layout_height="wrap_content"
345         android:checkedButton="@id/sixteen_button_T1"
346         android:orientation="horizontal"
347         app:layout_constraintBottom_toBottomOf="@+id/
    sixteen_hour_text"
348         app:layout_constraintStart_toStartOf="@+id/eight_hour_group"
349         app:layout_constraintTop_toTopOf="@+id/sixteen_hour_text">
350 
351         <RadioButton
352             android:id="@+id/sixteen_button_T1"
353             android:layout_width="wrap_content"
354             android:layout_height="wrap_content"
355             android:text="T1" />
356 
357         <RadioButton
358             android:id="@+id/sixteen_button_T2"
```

```
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/seventeen_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="30dp"
        android:text="17:00-18:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@id/eight_hour_text"
        app:layout_constraintTop_toBottomOf="@id/sixteen_hour_text"
    />

    <RadioGroup
        android:id="@+id/seventeen_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/seventeen_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/
seventeen_hour_text"
        app:layout_constraintStart_toStartOf="@+id/eight_hour_group"
        app:layout_constraintTop_toTopOf="@+id/seventeen_hour_text">

        <RadioButton
            android:id="@+id/seventeen_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/seventeen_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/eighteen_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="30dp"
        android:text="18:00-19:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@id/eight_hour_text"
        app:layout_constraintTop_toBottomOf="@id/seventeen_hour_text"
    />

    <RadioGroup
        android:id="@+id/eighteen_hour_group"
```

```
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/eighteen_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/
eighteen_hour_text"
        app:layout_constraintStart_toStartOf="@+id/eight_hour_group"
        app:layout_constraintTop_toTopOf="@+id/eighteen_hour_text">

        <RadioButton
            android:id="@+id/eighteen_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/eighteen_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/nineteen_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="30dp"
        android:text="19:00-20:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@id/eight_hour_text"
        app:layout_constraintTop_toBottomOf="@id/eighteen_hour_text"
 />

    <RadioGroup
        android:id="@+id/nineteen_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/nineteen_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/
nineteen_hour_text"
        app:layout_constraintStart_toStartOf="@+id/eight_hour_group"
        app:layout_constraintTop_toTopOf="@+id/nineteen_hour_text">

        <RadioButton
            android:id="@+id/nineteen_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/nineteen_button_T2"
            android:layout_width="wrap_content"
```

```
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/twenty_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginStart="10dp"
        android:text="20:00-21:00"
        android:textSize="16sp"
        app:layout_constraintStart_toEndOf="@+id/eight_hour_group"
        app:layout_constraintTop_toTopOf="@+id/eight_hour_text" />

    <RadioGroup
        android:id="@+id/twenty_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/twenty_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/twenty_hour_text
"
        app:layout_constraintStart_toEndOf="@+id/twenty_hour_text"
        app:layout_constraintTop_toTopOf="@+id/twenty_hour_text">

        <RadioButton
            android:id="@+id/twenty_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/twenty_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/twenty_one_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="21:00-22:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_text"
        app:layout_constraintTop_toTopOf="@id/nine_hour_text" />

    <RadioGroup
        android:id="@+id/twenty_one_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/twenty_one_button_T1"
        android:orientation="horizontal"
```

```
        app:layout_constraintBottom_toBottomOf="@+id/
    twenty_one_hour_text"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_group"
        app:layout_constraintTop_toTopOf="@+id/twenty_one_hour_text">

        <RadioButton
            android:id="@+id/twenty_one_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/twenty_one_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/twenty_two_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="22:00-23:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_text"
        app:layout_constraintTop_toTopOf="@id/ten_hour_text" />

    <RadioGroup
        android:id="@+id/twenty_two_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/twenty_two_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/
    twenty_two_hour_text"
        app:layout_constraintStart_toEndOf="@+id/twenty_two_hour_text
    "
        app:layout_constraintTop_toTopOf="@+id/twenty_two_hour_text">

        <RadioButton
            android:id="@+id/twenty_two_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/twenty_two_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
```

```
        android:id="@+id/twenty_three_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="23:00-00:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_text"
        app:layout_constraintTop_toTopOf="@id/eleven_hour_text" />

    <RadioGroup
        android:id="@+id/twenty_three_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/twenty_three_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/
twenty_three_hour_text"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_group"
        app:layout_constraintTop_toTopOf="@+id/twenty_three_hour_text
">

        <RadioButton
            android:id="@+id/twenty_three_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/twenty_three_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/zero_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="00:00-01:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_text"
        app:layout_constraintTop_toTopOf="@id/twelve_hour_text" />

    <RadioGroup
        android:id="@+id/zero_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/zero_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/zero_hour_text"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_group"
        app:layout_constraintTop_toTopOf="@+id/zero_hour_text">

        <RadioButton
```

```
            android:id="@+id/zero_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/zero_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/one_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="01:00-02:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_text"
        app:layout_constraintTop_toTopOf="@id/thirteen_hour_text" />

    <RadioGroup
        android:id="@+id/one_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/one_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/one_hour_text"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_group"
        app:layout_constraintTop_toTopOf="@+id/one_hour_text">

        <RadioButton
            android:id="@+id/one_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/one_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/two_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="02:00-03:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_text"
        app:layout_constraintTop_toTopOf="@id/fourteen_hour_text" />

```

```
    <RadioGroup
        android:id="@+id/two_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/two_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/two_hour_text"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_group"
        app:layout_constraintTop_toTopOf="@+id/two_hour_text">

        <RadioButton
            android:id="@+id/two_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/two_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/three_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="03:00-04:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_text"
        app:layout_constraintTop_toTopOf="@id/fifteen_hour_text" />

    <RadioGroup
        android:id="@+id/three_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/three_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/three_hour_text"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_group"
        app:layout_constraintTop_toTopOf="@+id/three_hour_text">

        <RadioButton
            android:id="@+id/three_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/three_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
```

```
    </RadioGroup>

    <TextView
        android:id="@+id/four_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="04:00-05:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_text"
        app:layout_constraintTop_toTopOf="@id/sixteen_hour_text" />

    <RadioGroup
        android:id="@+id/four_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/four_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/four_hour_text"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_group"
        app:layout_constraintTop_toTopOf="@+id/four_hour_text">

        <RadioButton
            android:id="@+id/four_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/four_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/five_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="05:00-06:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_text"
        app:layout_constraintTop_toTopOf="@id/seventeen_hour_text" />

    <RadioGroup
        android:id="@+id/five_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/five_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/five_hour_text"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_group"
        app:layout_constraintTop_toTopOf="@+id/five_hour_text">

```

```
        <RadioButton
            android:id="@+id/five_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/five_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/six_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="06:00-07:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_text"
        app:layout_constraintTop_toTopOf="@id/eighteen_hour_text" />

    <RadioGroup
        android:id="@+id/six_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/six_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/six_hour_text"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_group"
        app:layout_constraintTop_toTopOf="@+id/six_hour_text">

        <RadioButton
            android:id="@+id/six_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/six_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/seven_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="07:00-08:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_text"
        app:layout_constraintTop_toTopOf="@id/nineteen_hour_text" />
```

```
824
825     <RadioGroup
826         android:id="@+id/seven_hour_group"
827         android:layout_width="wrap_content"
828         android:layout_height="wrap_content"
829         android:checkedButton="@id/seven_button_T1"
830         android:orientation="horizontal"
831         app:layout_constraintBottom_toBottomOf="@+id/seven_hour_text"
832         app:layout_constraintStart_toStartOf="@+id/twenty_hour_group"
833         app:layout_constraintTop_toTopOf="@+id/seven_hour_text">
834
835         <RadioButton
836             android:id="@+id/seven_button_T1"
837             android:layout_width="wrap_content"
838             android:layout_height="wrap_content"
839             android:text="T1" />
840
841         <RadioButton
842             android:id="@+id/seven_button_T2"
843             android:layout_width="wrap_content"
844             android:layout_height="wrap_content"
845             android:text="T2" />
846     </RadioGroup>
847
848 </androidx.constraintlayout.widget.ConstraintLayout>
```

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".SundayProgramActivity">

    <TextView
        android:id="@+id/day_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="DOMINGO"
        android:textSize="22sp"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintTop_toBottomOf="@id/nineteen_hour_group" />

    <Button
        android:id="@+id/save_button"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Guardar"
        app:layout_constraintStart_toStartOf="@+id/nineteen_hour_group"
        app:layout_constraintTop_toBottomOf="@+id/day_text" />

    <Button
        android:id="@+id/update_button"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginStart="12dp"
        android:text="Ver actual"
        app:layout_constraintStart_toEndOf="@+id/save_button"
        app:layout_constraintTop_toTopOf="@id/save_button" />

    <View
        android:id="@+id/divider_h_up"
        android:layout_width="0dp"
        android:layout_height="2dp"
        android:layout_marginTop="-3dp"
        android:background="?attr/colorPrimaryVariant"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="@+id/eight_hour_text" />

    <View
        android:id="@+id/divider_v"
        android:layout_width="2dp"
        android:layout_height="0dp"
        android:layout_marginStart="5dp"
```

```
        android:background="?attr/colorPrimaryVariant"
        app:layout_constraintBottom_toBottomOf="@+id/divider_h_down"
        app:layout_constraintStart_toEndOf="@+id/eight_hour_group"
        app:layout_constraintTop_toTopOf="@id/divider_h_up" />

    <View
        android:id="@+id/divider_h_down"
        android:layout_width="0dp"
        android:layout_height="2dp"
        android:layout_marginTop="2dp"
        android:background="?attr/colorPrimaryVariant"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@id/nineteen_hour_text"
/>

    <TextView
        android:id="@+id/eight_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginStart="6dp"
        android:layout_marginTop="14dp"
        android:text="08:00-09:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

    <RadioGroup
        android:id="@+id/eight_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/eight_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/eight_hour_text"
        app:layout_constraintStart_toEndOf="@+id/eight_hour_text"
        app:layout_constraintTop_toTopOf="@+id/eight_hour_text">

        <RadioButton
            android:id="@+id/eight_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/eight_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/nine_hour_text"
        android:layout_width="wrap_content"
```

```
        android:layout_height="wrap_content"
        android:layout_marginTop="30dp"
        android:text="09:00-10:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@id/eight_hour_text"
        app:layout_constraintTop_toBottomOf="@id/eight_hour_text" />

    <RadioGroup
        android:id="@+id/nine_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/nine_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/nine_hour_text"
        app:layout_constraintStart_toStartOf="@+id/eight_hour_group"
        app:layout_constraintTop_toTopOf="@+id/nine_hour_text">

        <RadioButton
            android:id="@+id/nine_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/nine_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/ten_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="30dp"
        android:text="10:00-11:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@+id/eight_hour_text"
        app:layout_constraintTop_toBottomOf="@id/nine_hour_text" />

    <RadioGroup
        android:id="@+id/ten_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/ten_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/ten_hour_text"
        app:layout_constraintStart_toStartOf="@+id/eight_hour_group"
        app:layout_constraintTop_toTopOf="@+id/ten_hour_text">

        <RadioButton
            android:id="@+id/ten_button_T1"
            android:layout_width="wrap_content"
```

```
156             android:layout_height="wrap_content"
157             android:text="T1" />
158 
159         <RadioButton
160             android:id="@+id/ten_button_T2"
161             android:layout_width="wrap_content"
162             android:layout_height="wrap_content"
163             android:text="T2" />
164     </RadioGroup>
165 
166     <TextView
167         android:id="@+id/eleven_hour_text"
168         android:layout_width="wrap_content"
169         android:layout_height="wrap_content"
170         android:layout_marginTop="30dp"
171         android:text="11:00-12:00"
172         android:textSize="16sp"
173         app:layout_constraintStart_toStartOf="@id/eight_hour_text"
174         app:layout_constraintTop_toBottomOf="@id/ten_hour_text" />
175 
176     <RadioGroup
177         android:id="@+id/eleven_hour_group"
178         android:layout_width="wrap_content"
179         android:layout_height="wrap_content"
180         android:checkedButton="@id/eleven_button_T1"
181         android:orientation="horizontal"
182         app:layout_constraintBottom_toBottomOf="@+id/eleven_hour_text
    "
183         app:layout_constraintStart_toStartOf="@+id/eight_hour_group"
184         app:layout_constraintTop_toTopOf="@+id/eleven_hour_text">
185 
186         <RadioButton
187             android:id="@+id/eleven_button_T1"
188             android:layout_width="wrap_content"
189             android:layout_height="wrap_content"
190             android:text="T1" />
191 
192         <RadioButton
193             android:id="@+id/eleven_button_T2"
194             android:layout_width="wrap_content"
195             android:layout_height="wrap_content"
196             android:text="T2" />
197     </RadioGroup>
198 
199     <TextView
200         android:id="@+id/twelve_hour_text"
201         android:layout_width="wrap_content"
202         android:layout_height="wrap_content"
203         android:layout_marginTop="30dp"
204         android:text="12:00-13:00"
205         android:textSize="16sp"
206         app:layout_constraintStart_toStartOf="@id/eight_hour_text"
207         app:layout_constraintTop_toBottomOf="@id/eleven_hour_text" />
```

```
208
209     <RadioGroup
210         android:id="@+id/twelve_hour_group"
211         android:layout_width="wrap_content"
212         android:layout_height="wrap_content"
213         android:checkedButton="@id/twelve_button_T1"
214         android:orientation="horizontal"
215         app:layout_constraintBottom_toBottomOf="@+id/twelve_hour_text
    "
216         app:layout_constraintStart_toStartOf="@+id/eight_hour_group"
217         app:layout_constraintTop_toTopOf="@+id/twelve_hour_text">
218
219         <RadioButton
220             android:id="@+id/twelve_button_T1"
221             android:layout_width="wrap_content"
222             android:layout_height="wrap_content"
223             android:text="T1" />
224
225         <RadioButton
226             android:id="@+id/twelve_button_T2"
227             android:layout_width="wrap_content"
228             android:layout_height="wrap_content"
229             android:text="T2" />
230     </RadioGroup>
231
232     <TextView
233         android:id="@+id/thirteen_hour_text"
234         android:layout_width="wrap_content"
235         android:layout_height="wrap_content"
236         android:layout_marginTop="30dp"
237         android:text="13:00-14:00"
238         android:textSize="16sp"
239         app:layout_constraintStart_toStartOf="@id/eight_hour_text"
240         app:layout_constraintTop_toBottomOf="@id/twelve_hour_text" />
241
242     <RadioGroup
243         android:id="@+id/thirteen_hour_group"
244         android:layout_width="wrap_content"
245         android:layout_height="wrap_content"
246         android:checkedButton="@id/thirteen_button_T1"
247         android:orientation="horizontal"
248         app:layout_constraintBottom_toBottomOf="@+id/
    thirteen_hour_text"
249         app:layout_constraintStart_toStartOf="@+id/eight_hour_group"
250         app:layout_constraintTop_toTopOf="@+id/thirteen_hour_text">
251
252         <RadioButton
253             android:id="@+id/thirteen_button_T1"
254             android:layout_width="wrap_content"
255             android:layout_height="wrap_content"
256             android:text="T1" />
257
258         <RadioButton
```

```
            android:id="@+id/thirteen_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/fourteen_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="30dp"
        android:text="14:00-15:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@id/eight_hour_text"
        app:layout_constraintTop_toBottomOf="@id/thirteen_hour_text"
    />

    <RadioGroup
        android:id="@+id/fourteen_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/fourteen_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/
    fourteen_hour_text"
        app:layout_constraintStart_toStartOf="@+id/eight_hour_group"
        app:layout_constraintTop_toTopOf="@+id/fourteen_hour_text">

        <RadioButton
            android:id="@+id/fourteen_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/fourteen_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/fifteen_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="30dp"
        android:text="15:00-16:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@id/eight_hour_text"
        app:layout_constraintTop_toBottomOf="@id/fourteen_hour_text"
    />

    <RadioGroup
```

```
309         android:id="@+id/fifteen_hour_group"
310         android:layout_width="wrap_content"
311         android:layout_height="wrap_content"
312         android:checkedButton="@id/fifteen_button_T1"
313         android:orientation="horizontal"
314         app:layout_constraintBottom_toBottomOf="@+id/
    fifteen_hour_text"
315         app:layout_constraintStart_toStartOf="@+id/eight_hour_group"
316         app:layout_constraintTop_toTopOf="@+id/fifteen_hour_text">
317 
318         <RadioButton
319             android:id="@+id/fifteen_button_T1"
320             android:layout_width="wrap_content"
321             android:layout_height="wrap_content"
322             android:text="T1" />
323 
324         <RadioButton
325             android:id="@+id/fifteen_button_T2"
326             android:layout_width="wrap_content"
327             android:layout_height="wrap_content"
328             android:text="T2" />
329     </RadioGroup>
330 
331     <TextView
332         android:id="@+id/sixteen_hour_text"
333         android:layout_width="wrap_content"
334         android:layout_height="wrap_content"
335         android:layout_marginTop="30dp"
336         android:text="16:00-17:00"
337         android:textSize="16sp"
338         app:layout_constraintStart_toStartOf="@id/eight_hour_text"
339         app:layout_constraintTop_toBottomOf="@id/fifteen_hour_text"
     />
340 
341     <RadioGroup
342         android:id="@+id/sixteen_hour_group"
343         android:layout_width="wrap_content"
344         android:layout_height="wrap_content"
345         android:checkedButton="@id/sixteen_button_T1"
346         android:orientation="horizontal"
347         app:layout_constraintBottom_toBottomOf="@+id/
    sixteen_hour_text"
348         app:layout_constraintStart_toStartOf="@+id/eight_hour_group"
349         app:layout_constraintTop_toTopOf="@+id/sixteen_hour_text">
350 
351         <RadioButton
352             android:id="@+id/sixteen_button_T1"
353             android:layout_width="wrap_content"
354             android:layout_height="wrap_content"
355             android:text="T1" />
356 
357         <RadioButton
358             android:id="@+id/sixteen_button_T2"
```

```
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/seventeen_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="30dp"
        android:text="17:00-18:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@id/eight_hour_text"
        app:layout_constraintTop_toBottomOf="@id/sixteen_hour_text"
    />

    <RadioGroup
        android:id="@+id/seventeen_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/seventeen_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/
seventeen_hour_text"
        app:layout_constraintStart_toStartOf="@+id/eight_hour_group"
        app:layout_constraintTop_toTopOf="@+id/seventeen_hour_text">

        <RadioButton
            android:id="@+id/seventeen_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/seventeen_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/eighteen_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="30dp"
        android:text="18:00-19:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@id/eight_hour_text"
        app:layout_constraintTop_toBottomOf="@id/seventeen_hour_text"
    />

    <RadioGroup
        android:id="@+id/eighteen_hour_group"
```

```
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/eighteen_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/
eighteen_hour_text"
        app:layout_constraintStart_toStartOf="@+id/eight_hour_group"
        app:layout_constraintTop_toTopOf="@+id/eighteen_hour_text">

        <RadioButton
            android:id="@+id/eighteen_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/eighteen_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/nineteen_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="30dp"
        android:text="19:00-20:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@id/eight_hour_text"
        app:layout_constraintTop_toBottomOf="@id/eighteen_hour_text"
 />

    <RadioGroup
        android:id="@+id/nineteen_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/nineteen_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/
nineteen_hour_text"
        app:layout_constraintStart_toStartOf="@+id/eight_hour_group"
        app:layout_constraintTop_toTopOf="@+id/nineteen_hour_text">

        <RadioButton
            android:id="@+id/nineteen_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/nineteen_button_T2"
            android:layout_width="wrap_content"
```

```
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/twenty_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginStart="10dp"
        android:text="20:00-21:00"
        android:textSize="16sp"
        app:layout_constraintStart_toEndOf="@+id/eight_hour_group"
        app:layout_constraintTop_toTopOf="@+id/eight_hour_text" />

    <RadioGroup
        android:id="@+id/twenty_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/twenty_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/twenty_hour_text
"
        app:layout_constraintStart_toEndOf="@+id/twenty_hour_text"
        app:layout_constraintTop_toTopOf="@+id/twenty_hour_text">

        <RadioButton
            android:id="@+id/twenty_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/twenty_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/twenty_one_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="21:00-22:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_text"
        app:layout_constraintTop_toTopOf="@id/nine_hour_text" />

    <RadioGroup
        android:id="@+id/twenty_one_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/twenty_one_button_T1"
        android:orientation="horizontal"
```

```
        app:layout_constraintBottom_toBottomOf="@+id/
    twenty_one_hour_text"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_group"
        app:layout_constraintTop_toTopOf="@+id/twenty_one_hour_text">

        <RadioButton
            android:id="@+id/twenty_one_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/twenty_one_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/twenty_two_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="22:00-23:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_text"
        app:layout_constraintTop_toTopOf="@id/ten_hour_text" />

    <RadioGroup
        android:id="@+id/twenty_two_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/twenty_two_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/
    twenty_two_hour_text"
        app:layout_constraintStart_toEndOf="@+id/twenty_two_hour_text
    "
        app:layout_constraintTop_toTopOf="@+id/twenty_two_hour_text">

        <RadioButton
            android:id="@+id/twenty_two_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/twenty_two_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
```

```
        android:id="@+id/twenty_three_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="23:00-00:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_text"
        app:layout_constraintTop_toTopOf="@id/eleven_hour_text" />

    <RadioGroup
        android:id="@+id/twenty_three_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/twenty_three_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/
twenty_three_hour_text"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_group"
        app:layout_constraintTop_toTopOf="@+id/twenty_three_hour_text
">

        <RadioButton
            android:id="@+id/twenty_three_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/twenty_three_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/zero_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="00:00-01:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_text"
        app:layout_constraintTop_toTopOf="@id/twelve_hour_text" />

    <RadioGroup
        android:id="@+id/zero_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/zero_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/zero_hour_text"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_group"
        app:layout_constraintTop_toTopOf="@+id/zero_hour_text">

        <RadioButton
```

```
            android:id="@+id/zero_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/zero_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/one_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="01:00-02:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_text"
        app:layout_constraintTop_toTopOf="@id/thirteen_hour_text" />

    <RadioGroup
        android:id="@+id/one_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/one_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/one_hour_text"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_group"
        app:layout_constraintTop_toTopOf="@+id/one_hour_text">

        <RadioButton
            android:id="@+id/one_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/one_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/two_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="02:00-03:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_text"
        app:layout_constraintTop_toTopOf="@id/fourteen_hour_text" />

```

```
    <RadioGroup
        android:id="@+id/two_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/two_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/two_hour_text"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_group"
        app:layout_constraintTop_toTopOf="@+id/two_hour_text">

        <RadioButton
            android:id="@+id/two_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/two_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/three_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="03:00-04:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_text"
        app:layout_constraintTop_toTopOf="@id/fifteen_hour_text" />

    <RadioGroup
        android:id="@+id/three_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/three_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/three_hour_text"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_group"
        app:layout_constraintTop_toTopOf="@+id/three_hour_text">

        <RadioButton
            android:id="@+id/three_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/three_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
```

```
    </RadioGroup>

    <TextView
        android:id="@+id/four_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="04:00-05:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_text"
        app:layout_constraintTop_toTopOf="@id/sixteen_hour_text" />

    <RadioGroup
        android:id="@+id/four_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/four_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/four_hour_text"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_group"
        app:layout_constraintTop_toTopOf="@+id/four_hour_text">

        <RadioButton
            android:id="@+id/four_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/four_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/five_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="05:00-06:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_text"
        app:layout_constraintTop_toTopOf="@id/seventeen_hour_text" />

    <RadioGroup
        android:id="@+id/five_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/five_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/five_hour_text"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_group"
        app:layout_constraintTop_toTopOf="@+id/five_hour_text">

```

```
        <RadioButton
            android:id="@+id/five_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/five_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/six_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="06:00-07:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_text"
        app:layout_constraintTop_toTopOf="@id/eighteen_hour_text" />

    <RadioGroup
        android:id="@+id/six_hour_group"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:checkedButton="@id/six_button_T1"
        android:orientation="horizontal"
        app:layout_constraintBottom_toBottomOf="@+id/six_hour_text"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_group"
        app:layout_constraintTop_toTopOf="@+id/six_hour_text">

        <RadioButton
            android:id="@+id/six_button_T1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T1" />

        <RadioButton
            android:id="@+id/six_button_T2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="T2" />
    </RadioGroup>

    <TextView
        android:id="@+id/seven_hour_text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="07:00-08:00"
        android:textSize="16sp"
        app:layout_constraintStart_toStartOf="@+id/twenty_hour_text"
        app:layout_constraintTop_toTopOf="@id/nineteen_hour_text" />
```

```
824
825     <RadioGroup
826         android:id="@+id/seven_hour_group"
827         android:layout_width="wrap_content"
828         android:layout_height="wrap_content"
829         android:checkedButton="@id/seven_button_T1"
830         android:orientation="horizontal"
831         app:layout_constraintBottom_toBottomOf="@+id/seven_hour_text"
832         app:layout_constraintStart_toStartOf="@+id/twenty_hour_group"
833         app:layout_constraintTop_toTopOf="@+id/seven_hour_text">
834
835         <RadioButton
836             android:id="@+id/seven_button_T1"
837             android:layout_width="wrap_content"
838             android:layout_height="wrap_content"
839             android:text="T1" />
840
841         <RadioButton
842             android:id="@+id/seven_button_T2"
843             android:layout_width="wrap_content"
844             android:layout_height="wrap_content"
845             android:text="T2" />
846     </RadioGroup>
847
848 </androidx.constraintlayout.widget.ConstraintLayout>
```

```xml
<?xml version="1.0" encoding="utf-8"?>
<menu xmlns:android="http://schemas.android.com/apk/res/android">
    <item
        android:id="@+id/homeFragment"
        android:enabled="true"
        android:icon="@drawable/ic_home"
        android:title="Inicio" />
    <item
        android:id="@+id/manualModeFragment"
        android:enabled="true"
        android:icon="@drawable/ic_manual_mode"
        android:title="Modo manual" />
    <item
        android:id="@+id/programModeFragment"
        android:enabled="true"
        android:icon="@drawable/ic_program_mode"
        android:title="Modo programa" />
</menu>
```

```xml
<?xml version="1.0" encoding="utf-8"?>
<navigation xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/my_navigation"
    app:startDestination="@id/homeFragment">

    <fragment
        android:id="@+id/homeFragment"
        android:name="com.example.home_t.fragments.menu.HomeFragment"
        android:label="fragment_home"
        tools:layout="@layout/fragment_home" />
    <fragment
        android:id="@+id/manualModeFragment"
        android:name="com.example.home_t.fragments.menu.
ManualModeFragment"
        android:label="fragment_manual_mode"
        tools:layout="@layout/fragment_manual_mode" />
    <fragment
        android:id="@+id/programModeFragment"
        android:name="com.example.home_t.fragments.menu.
ProgramModeFragment"
        android:label="fragment_program_mode"
        tools:layout="@layout/fragment_program_mode" />
</navigation>
```

Datasheets

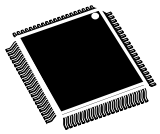
STM32F427xx STM32F429xx

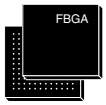
32b Arm® Cortex®-M4 MCU+FPU, 225DMIPS, up to 2MB Flash/256+4KB RAM, USB OTG HS/FS, Ethernet, 17 TIMs, 3 ADCs, 20 com. interfaces, camera & LCD-TFT

Datasheet - production data

Features

- Core: Arm® 32-bit Cortex®-M4 CPU with FPU, Adaptive real-time accelerator (ART Accelerator™) allowing 0-wait state execution from Flash memory, frequency up to 180 MHz, MPU, 225 DMIPS/1.25 DMIPS/MHz (Dhrystone 2.1), and DSP instructions
- Memories
  - Up to 2 MB of Flash memory organized into two banks allowing read-while-write
  - Up to 256+4 KB of SRAM including 64-KB of CCM (core coupled memory) data RAM
  - Flexible external memory controller with up to 32-bit data bus: SRAM, PSRAM, SDRAM/LPSDR SDRAM, Compact Flash/NOR/NAND memories
- LCD parallel interface, 8080/6800 modes
- LCD-TFT controller with fully programmable resolution (total width up to 4096 pixels, total height up to 2048 lines and pixel clock up to 83 MHz)
- Chrom-ART Accelerator™ for enhanced graphic content creation (DMA2D)
- Clock, reset and supply management
  - 1.7 V to 3.6 V application supply and I/Os
  - POR, PDR, PVD and BOR
  - 4-to-26 MHz crystal oscillator
  - Internal 16 MHz factory-trimmed RC (1% accuracy)
  - 32 kHz oscillator for RTC with calibration
  - Internal 32 kHz RC with calibration
- Low power
  - Sleep, Stop and Standby modes
  - V_{BAT} supply for RTC, 20×32 bit backup registers + optional 4 KB backup SRAM
- 3×12-bit, 2.4 MSPS ADC: up to 24 channels and 7.2 MSPS in triple interleaved mode
- 2×12-bit D/A converters
- General-purpose DMA: 16-stream DMA controller with FIFOs and burst support
- Up to 17 timers: up to twelve 16-bit and two 32-bit timers up to 180 MHz, each with up to 4 IC/OC/PWM or pulse counter and quadrature (incremental) encoder input





LQFP100 (14 × 14 mm)
LQFP144 (20 × 20 mm)
LQFP176 (24 × 24 mm)
LQFP208 (28 × 28 mm)

UFBGA176 (10 x 10 mm)
UFBGA169 (7 × 7 mm)
TFBGA216 (13 x 13 mm)

WLCSP143

- Debug mode
  - SWD & JTAG interfaces
  - Cortex-M4 Trace Macrocell™
- Up to 168 I/O ports with interrupt capability
  - Up to 164 fast I/Os up to 90 MHz
  - Up to 166 5 V-tolerant I/Os
- Up to 21 communication interfaces
  - Up to 3 × I^2C interfaces (SMBus/PMBus)
  - Up to 4 USARTs/4 UARTs (11.25 Mbit/s, ISO7816 interface, LIN, IrDA, modem control)
  - Up to 6 SPIs (45 Mbits/s), 2 with muxed full-duplex I^2S for audio class accuracy via internal audio PLL or external clock
  - 1 x SAI (serial audio interface)
  - 2 × CAN (2.0B Active) and SDIO interface
- Advanced connectivity
  - USB 2.0 full-speed device/host/OTG controller with on-chip PHY
  - USB 2.0 high-speed/full-speed device/host/OTG controller with dedicated DMA, on-chip full-speed PHY and ULPI
  - 10/100 Ethernet MAC with dedicated DMA: supports IEEE 1588v2 hardware, MII/RMII
- 8- to 14-bit parallel camera interface up to 54 Mbytes/s
- True random number generator
- CRC calculation unit
- RTC: subsecond accuracy, hardware calendar
- 96-bit unique ID

This is information on a product in full production. www.st.com

Table 1. Device summary

Reference	Part number
STM32F427xx	STM32F427VG, STM32F427ZG, STM32F427IG, STM32F427AG, STM32F427VI, STM32F427ZI, STM32F427II, STM32F427AI
STM32F429xx	STM32F429VG, STM32F429ZG, STM32F429IG, STM32F429BG, STM32F429NG, STM32F429AG, STM32F429VI, STM32F429ZI, STM32F429II,, STM32F429BI, STM32F429NI,STM32F429AI, STM32F429VE, STM32F429ZE, STM32F429IE, STM32F429BE, STM32F429NE

Contents

1 **Introduction** **13**

2 **Description** **14**

2.1 Full compatibility throughout the family 18

3 **Functional overview** **21**

3.1 Arm® Cortex®-M4 with FPU and embedded Flash and SRAM 21

3.2 Adaptive real-time memory accelerator (ART Accelerator™) 21

3.3 Memory protection unit 21

3.4 Embedded Flash memory 22

3.5 CRC (cyclic redundancy check) calculation unit 22

3.6 Embedded SRAM 22

3.7 Multi-AHB bus matrix 22

3.8 DMA controller (DMA) 23

3.9 Flexible memory controller (FMC) 24

3.10 LCD-TFT controller (available only on STM32F429xx) 24

3.11 Chrom-ART Accelerator™ (DMA2D) 25

3.12 Nested vectored interrupt controller (NVIC) 25

3.13 External interrupt/event controller (EXTI) 25

3.14 Clocks and startup 25

3.15 Boot modes 26

3.16 Power supply schemes 26

3.17 Power supply supervisor 26

3.17.1 Internal reset ON 26

3.17.2 Internal reset OFF 27

3.18 Voltage regulator 28

3.18.1 Regulator ON 28

3.18.2 Regulator OFF 29

3.18.3 Regulator ON/OFF and internal reset ON/OFF availability 32

3.19 Real-time clock (RTC), backup SRAM and backup registers 32

3.20 Low-power modes 33

3.21 V_{BAT} operation 34

3.22 Timers and watchdogs 34
3.22.1 Advanced-control timers (TIM1, TIM8) 36
3.22.2 General-purpose timers (TIMx) 36
3.22.3 Basic timers TIM6 and TIM7 36
3.22.4 Independent watchdog 37
3.22.5 Window watchdog 37
3.22.6 SysTick timer 37
3.23 Inter-integrated circuit interface (I^2C) 37
3.24 Universal synchronous/asynchronous receiver transmitters (USART) . . 37
3.25 Serial peripheral interface (SPI) 38
3.26 Inter-integrated sound (I^2S) 39
3.27 Serial Audio interface (SAI1) 39
3.28 Audio PLL (PLLI2S) 39
3.29 Audio and LCD PLL(PLLSAI) 39
3.30 Secure digital input/output interface (SDIO) 40
3.31 Ethernet MAC interface with dedicated DMA and IEEE 1588 support . . . 40
3.32 Controller area network (bxCAN) 40
3.33 Universal serial bus on-the-go full-speed (OTG_FS) 41
3.34 Universal serial bus on-the-go high-speed (OTG_HS) 41
3.35 Digital camera interface (DCMI) 42
3.36 Random number generator (RNG) 42
3.37 General-purpose input/outputs (GPIOs) 42
3.38 Analog-to-digital converters (ADCs) 42
3.39 Temperature sensor 43
3.40 Digital-to-analog converter (DAC) 43
3.41 Serial wire JTAG debug port (SWJ-DP) 43
3.42 Embedded Trace Macrocell™ 44

4 Pinouts and pin description 45

5 Memory mapping 86

6 Electrical characteristics 91
6.1 Parameter conditions 91
6.1.1 Minimum and maximum values 91

6.1.2 Typical values . . . 91
6.1.3 Typical curves . . . 91
6.1.4 Loading capacitor . . . 91
6.1.5 Pin input voltage . . . 91
6.1.6 Power supply scheme . . . 92
6.1.7 Current consumption measurement . . . 93
6.2 Absolute maximum ratings . . . 93
6.3 Operating conditions . . . 95
6.3.1 General operating conditions . . . 95
6.3.2 VCAP1/VCAP2 external capacitor . . . 97
6.3.3 Operating conditions at power-up / power-down (regulator ON) . . . 98
6.3.4 Operating conditions at power-up / power-down (regulator OFF) . . . 98
6.3.5 Reset and power control block characteristics . . . 99
6.3.6 Over-drive switching characteristics . . . 100
6.3.7 Supply current characteristics . . . 101
6.3.8 Wakeup time from low-power modes . . . 117
6.3.9 External clock source characteristics . . . 118
6.3.10 Internal clock source characteristics . . . 122
6.3.11 PLL characteristics . . . 124
6.3.12 PLL spread spectrum clock generation (SSCG) characteristics . . . 127
6.3.13 Memory characteristics . . . 129
6.3.14 EMC characteristics . . . 131
6.3.15 Absolute maximum ratings (electrical sensitivity) . . . 133
6.3.16 I/O current injection characteristics . . . 134
6.3.17 I/O port characteristics . . . 135
6.3.18 NRST pin characteristics . . . 141
6.3.19 TIM timer characteristics . . . 142
6.3.20 Communications interfaces . . . 142
6.3.21 12-bit ADC characteristics . . . 157
6.3.22 Temperature sensor characteristics . . . 163
6.3.23 V_{BAT} monitoring characteristics . . . 164
6.3.24 Reference voltage . . . 164
6.3.25 DAC electrical characteristics . . . 165
6.3.26 FMC characteristics . . . 168
6.3.27 Camera interface (DCMI) timing specifications . . . 193
6.3.28 LCD-TFT controller (LTDC) characteristics . . . 194
6.3.29 SD/SDIO MMC card host interface (SDIO) characteristics . . . 196

6.3.30 RTC characteristics 197

7 Package information 198

7.1 LQFP100 package information 198

7.2 WLCSP143 package information 202

7.3 LQFP144 package information 205

7.4 LQFP176 package information 209

7.5 LQFP208 package information 213

7.6 UFBGA169 package information 217

7.7 UFBGA176+25 package information 220

7.8 TFBGA216 package information 223

7.9 Thermal characteristics 225

8 Part numbering 226

Appendix A Recommendations when using internal reset OFF 227

A.1 Operating conditions 227

Appendix B Application block diagrams 228

B.1 USB OTG full speed (FS) interface solutions 228

B.2 USB OTG high speed (HS) interface solutions 230

B.3 Ethernet interface solutions 231

9 Revision history 233

List of tables

Table 1. Device summary . . . 2
Table 2. STM32F427xx and STM32F429xx features and peripheral counts . . . 16
Table 3. Voltage regulator configuration mode versus device operating mode . . . 29
Table 4. Regulator ON/OFF and internal reset ON/OFF availability . . . 32
Table 5. Voltage regulator modes in stop mode . . . 33
Table 6. Timer feature comparison . . . 35
Table 7. Comparison of I2C analog and digital filters . . . 37
Table 8. USART feature comparison . . . 38
Table 9. Legend/abbreviations used in the pinout table . . . 53
Table 10. STM32F427xx and STM32F429xx pin and ball definitions . . . 53
Table 11. FMC pin definition . . . 72
Table 12. STM32F427xx and STM32F429xx alternate function mapping . . . 75
Table 13. STM32F427xx and STM32F429xx register boundary addresses . . . 87
Table 14. Voltage characteristics . . . 93
Table 15. Current characteristics . . . 94
Table 16. Thermal characteristics . . . 94
Table 17. General operating conditions . . . 95
Table 18. Limitations depending on the operating power supply range . . . 97
Table 19. VCAP1/VCAP2 operating conditions . . . 97
Table 20. Operating conditions at power-up / power-down (regulator ON) . . . 98
Table 21. Operating conditions at power-up / power-down (regulator OFF) . . . 98
Table 22. reset and power control block characteristics . . . 99
Table 23. Over-drive switching characteristics . . . 100
Table 24. Typical and maximum current consumption in Run mode, code with data processing running from Flash memory (ART accelerator enabled except prefetch) or RAM . . . 102
Table 25. Typical and maximum current consumption in Run mode, code with data processing running from Flash memory (ART accelerator disabled) . . . 103
Table 26. Typical and maximum current consumption in Sleep mode . . . 104
Table 27. Typical and maximum current consumptions in Stop mode . . . 105
Table 28. Typical and maximum current consumptions in Standby mode . . . 106
Table 29. Typical and maximum current consumptions in V_{BAT} mode . . . 106
Table 30. Typical current consumption in Run mode, code with data processing running from Flash memory or RAM, regulator ON (ART accelerator enabled except prefetch), VDD=1.7 V . . . 108
Table 31. Typical current consumption in Run mode, code with data processing running from Flash memory, regulator OFF (ART accelerator enabled except prefetch) . . . 109
Table 32. Typical current consumption in Sleep mode, regulator ON, VDD=1.7 V . . . 110
Table 33. Tyical current consumption in Sleep mode, regulator OFF . . . 111
Table 34. Switching output I/O current consumption . . . 113
Table 35. Peripheral current consumption . . . 114
Table 36. Low-power mode wakeup timings . . . 117
Table 37. High-speed external user clock characteristics . . . 118
Table 38. Low-speed external user clock characteristics . . . 119
Table 39. HSE 4-26 MHz oscillator characteristics . . . 120
Table 40. LSE oscillator characteristics (f_{LSE} = 32.768 kHz) . . . 121
Table 41. HSI oscillator characteristics . . . 122
Table 42. LSI oscillator characteristics . . . 123
Table 43. Main PLL characteristics . . . 124

Table 44. PLLI2S (audio PLL) characteristics . . . 125
Table 45. PLLISAI (audio and LCD-TFT PLL) characteristics . . . 126
Table 46. SSCG parameters constraint . . . 127
Table 47. Flash memory characteristics . . . 129
Table 48. Flash memory programming . . . 129
Table 49. Flash memory programming with V_{PP} . . . 130
Table 50. Flash memory endurance and data retention . . . 131
Table 51. EMS characteristics . . . 131
Table 52. EMI characteristics . . . 132
Table 53. ESD absolute maximum ratings . . . 133
Table 54. Electrical sensitivities . . . 133
Table 55. I/O current injection susceptibility . . . 134
Table 56. I/O static characteristics . . . 135
Table 57. Output voltage characteristics . . . 138
Table 58. I/O AC characteristics . . . 139
Table 59. NRST pin characteristics . . . 141
Table 60. TIMx characteristics . . . 142
Table 61. I2C analog filter characteristics . . . 143
Table 62. SPI dynamic characteristics . . . 143
Table 63. I^2S dynamic characteristics . . . 146
Table 64. SAI characteristics . . . 148
Table 65. USB OTG full speed startup time . . . 150
Table 66. USB OTG full speed DC electrical characteristics . . . 150
Table 67. USB OTG full speed electrical characteristics . . . 151
Table 68. USB HS DC electrical characteristics . . . 151
Table 69. USB HS clock timing parameters . . . 152
Table 70. Dynamic characteristics: USB ULPI . . . 153
Table 71. Dynamics characteristics: Ethernet MAC signals for SMI . . . 154
Table 72. Dynamics characteristics: Ethernet MAC signals for RMII . . . 155
Table 73. Dynamics characteristics: Ethernet MAC signals for MII . . . 156
Table 74. ADC characteristics . . . 157
Table 75. ADC static accuracy at f_{ADC} = 18 MHz . . . 158
Table 76. ADC static accuracy at f_{ADC} = 30 MHz . . . 159
Table 77. ADC static accuracy at f_{ADC} = 36 MHz . . . 159
Table 78. ADC dynamic accuracy at f_{ADC} = 18 MHz - limited test conditions . . . 159
Table 79. ADC dynamic accuracy at f_{ADC} = 36 MHz - limited test conditions . . . 159
Table 80. Temperature sensor characteristics . . . 163
Table 81. Temperature sensor calibration values . . . 163
Table 82. V_{BAT} monitoring characteristics . . . 164
Table 83. internal reference voltage . . . 164
Table 84. Internal reference voltage calibration values . . . 164
Table 85. DAC characteristics . . . 165
Table 86. Asynchronous non-multiplexed SRAM/PSRAM/NOR - read timings . . . 169
Table 87. Asynchronous non-multiplexed SRAM/PSRAM/NOR read - NWAIT timings . . . 170
Table 88. Asynchronous non-multiplexed SRAM/PSRAM/NOR write timings . . . 171
Table 89. Asynchronous non-multiplexed SRAM/PSRAM/NOR write - NWAIT timings . . . 172
Table 90. Asynchronous multiplexed PSRAM/NOR read timings . . . 173
Table 91. Asynchronous multiplexed PSRAM/NOR read-NWAIT timings . . . 173
Table 92. Asynchronous multiplexed PSRAM/NOR write timings . . . 174

Table 93. Asynchronous multiplexed PSRAM/NOR write-NWAIT timings . . . 175
Table 94. Synchronous multiplexed NOR/PSRAM read timings . . . 176
Table 95. Synchronous multiplexed PSRAM write timings . . . 178
Table 96. Synchronous non-multiplexed NOR/PSRAM read timings . . . 179
Table 97. Synchronous non-multiplexed PSRAM write timings . . . 180
Table 98. Switching characteristics for PC Card/CF read and write cycles in attribute/common space . . . 185
Table 99. Switching characteristics for PC Card/CF read and write cycles in I/O space . . . 186
Table 100. Switching characteristics for NAND Flash read cycles . . . 188
Table 101. Switching characteristics for NAND Flash write cycles . . . 189
Table 102. SDRAM read timings . . . 190
Table 103. LPSDR SDRAM read timings . . . 190
Table 104. SDRAM write timings . . . 192
Table 105. LPSDR SDRAM write timings . . . 192
Table 106. DCMI characteristics . . . 193
Table 107. LTDC characteristics . . . 194
Table 108. Dynamic characteristics: SD / MMC characteristics . . . 197
Table 109. RTC characteristics . . . 197
Table 110. LQPF100 100-pin, 14 x 14 mm low-profile quad flat package mechanical data . . . 199
Table 111. WLCSP143 - 143-ball, 4.521x 5.547 mm, 0.4 mm pitch wafer level chip scale package mechanical data . . . 203
Table 112. WLCSP143 recommended PCB design rules (0.4 mm pitch) . . . 204
Table 113. LQFP144 - 144-pin, 20 x 20 mm low-profile quad flat package mechanical data . . . 206
Table 114. LQFP176 - 176-pin, 24 x 24 mm low-profile quad flat package mechanical data . . . 209
Table 115. LQFP208 - 208-pin, 28 x 28 mm low-profile quad flat package mechanical data . . . 214
Table 116. UFBGA169 - 169-ball 7 x 7 mm 0.50 mm pitch, ultra fine pitch ball grid array package mechanical data . . . 217
Table 117. UFBGA169 recommended PCB design rules (0.5 mm pitch BGA) . . . 218
Table 118. UFBGA176+25 - ball, 10 x 10 mm, 0.65 mm pitch, ultra fine pitch ball grid array package mechanical data . . . 220
Table 119. UFBGA176+25 recommended PCB design rules (0.65 mm pitch BGA) . . . 221
Table 120. TFBGA216 - 216 ball 13 × 13 mm 0.8 mm pitch thin fine pitch ball grid array package mechanical data . . . 223
Table 121. Package thermal characteristics . . . 225
Table 122. Ordering information scheme . . . 226
Table 123. Limitations depending on the operating power supply range . . . 227
Table 124. Document revision history . . . 233

List of figures

Figure 1. Compatible board design STM32F10xx/STM32F2xx/STM32F4xx for LQFP100 package . . . 18
Figure 2. Compatible board design between STM32F10xx/STM32F2xx/STM32F4xx for LQFP144 package . . . 19
Figure 3. Compatible board design between STM32F2xx and STM32F4xx for LQFP176 and UFBGA176 packages . . . 19
Figure 4. STM32F427xx and STM32F429xx block diagram . . . 20
Figure 5. STM32F427xx and STM32F429xx Multi-AHB matrix . . . 23
Figure 6. Power supply supervisor interconnection with internal reset OFF . . . 27
Figure 7. PDR_ON control with internal reset OFF . . . 28
Figure 8. Regulator OFF . . . 30
Figure 9. Startup in regulator OFF: slow V_{DD} slope - power-down reset risen after V_{CAP_1}/V_{CAP_2} stabilization . . . 31
Figure 10. Startup in regulator OFF mode: fast V_{DD} slope - power-down reset risen before V_{CAP_1}/V_{CAP_2} stabilization . . . 31
Figure 11. STM32F42x LQFP100 pinout . . . 45
Figure 12. STM32F42x WLCSP143 ballout . . . 46
Figure 13. STM32F42x LQFP144 pinout . . . 47
Figure 14. STM32F42x LQFP176 pinout . . . 48
Figure 15. STM32F42x LQFP208 pinout . . . 49
Figure 16. STM32F42x UFBGA169 ballout . . . 50
Figure 17. STM32F42x UFBGA176 ballout . . . 51
Figure 18. STM32F42x TFBGA216 ballout . . . 52
Figure 19. Memory map . . . 86
Figure 20. Pin loading conditions . . . 91
Figure 21. Pin input voltage . . . 91
Figure 22. Power supply scheme . . . 92
Figure 23. Current consumption measurement scheme . . . 93
Figure 24. External capacitor C_{EXT} . . . 97
Figure 25. Typical V_{BAT} current consumption (LSE and RTC ON/backup RAM OFF) . . . 107
Figure 26. Typical V_{BAT} current consumption (LSE and RTC ON/backup RAM ON) . . . 107
Figure 27. High-speed external clock source AC timing diagram . . . 119
Figure 28. Low-speed external clock source AC timing diagram . . . 120
Figure 29. Typical application with an 8 MHz crystal . . . 121
Figure 30. Typical application with a 32.768 kHz crystal . . . 122
Figure 31. ACCHSI accuracy versus temperature . . . 123
Figure 32. ACC_{LSI} versus temperature . . . 124
Figure 33. PLL output clock waveforms in center spread mode . . . 128
Figure 34. PLL output clock waveforms in down spread mode . . . 128
Figure 35. FT I/O input characteristics . . . 137
Figure 36. I/O AC characteristics definition . . . 140
Figure 37. Recommended NRST pin protection . . . 141
Figure 38. SPI timing diagram - slave mode and CPHA = 0 . . . 144
Figure 39. SPI timing diagram - slave mode and CPHA = 1 . . . 145
Figure 40. SPI timing diagram - master mode . . . 145
Figure 41. I^2S slave timing diagram (Philips protocol)(1) . . . 147
Figure 42. I^2S master timing diagram (Philips protocol)(1) . . . 147
Figure 43. SAI master timing waveforms . . . 149

Figure 44. SAI slave timing waveforms . . . 149
Figure 45. USB OTG full speed timings: definition of data signal rise and fall time . . . 151
Figure 46. ULPI timing diagram . . . 152
Figure 47. Ethernet SMI timing diagram . . . 154
Figure 48. Ethernet RMII timing diagram . . . 155
Figure 49. Ethernet MII timing diagram . . . 156
Figure 50. ADC accuracy characteristics . . . 160
Figure 51. Typical connection diagram using the ADC . . . 161
Figure 52. Power supply and reference decoupling (V_{REF+} not connected to V_{DDA}) . . . 162
Figure 53. Power supply and reference decoupling (V_{REF+} connected to V_{DDA}) . . . 163
Figure 54. 12-bit buffered /non-buffered DAC . . . 167
Figure 55. Asynchronous non-multiplexed SRAM/PSRAM/NOR read waveforms . . . 169
Figure 56. Asynchronous non-multiplexed SRAM/PSRAM/NOR write waveforms . . . 171
Figure 57. Asynchronous multiplexed PSRAM/NOR read waveforms . . . 172
Figure 58. Asynchronous multiplexed PSRAM/NOR write waveforms . . . 174
Figure 59. Synchronous multiplexed NOR/PSRAM read timings . . . 176
Figure 60. Synchronous multiplexed PSRAM write timings . . . 177
Figure 61. Synchronous non-multiplexed NOR/PSRAM read timings . . . 179
Figure 62. Synchronous non-multiplexed PSRAM write timings . . . 180
Figure 63. PC Card/CompactFlash controller waveforms for common memory read access . . . 182
Figure 64. PC Card/CompactFlash controller waveforms for common memory write access . . . 182
Figure 65. PC Card/CompactFlash controller waveforms for attribute memory read access . . . 183
Figure 66. PC Card/CompactFlash controller waveforms for attribute memory write access . . . 184
Figure 67. PC Card/CompactFlash controller waveforms for I/O space read access . . . 184
Figure 68. PC Card/CompactFlash controller waveforms for I/O space write access . . . 185
Figure 69. NAND controller waveforms for read access . . . 187
Figure 70. NAND controller waveforms for write access . . . 187
Figure 71. NAND controller waveforms for common memory read access . . . 188
Figure 72. NAND controller waveforms for common memory write access . . . 188
Figure 73. SDRAM read access waveforms (CL = 1) . . . 189
Figure 74. SDRAM write access waveforms . . . 191
Figure 75. DCMI timing diagram . . . 193
Figure 76. LCD-TFT horizontal timing diagram . . . 195
Figure 77. LCD-TFT vertical timing diagram . . . 195
Figure 78. SDIO high-speed mode . . . 196
Figure 79. SD default mode . . . 196
Figure 80. LQFP100 -100-pin, 14 x 14 mm low-profile quad flat package outline . . . 198
Figure 81. LQPF100 - 100-pin, 14 x 14 mm low-profile quad flat recommended footprint . . . 200
Figure 82. LQFP100 marking example (package top view) . . . 201
Figure 83. WLCSP143 - 143-ball, 4.521x 5.547 mm, 0.4 mm pitch wafer level chip scale package outline . . . 202
Figure 84. WLCSP143 - 143-ball, 4.521x 5.547 mm, 0.4 mm pitch wafer level chip scale recommended footprint . . . 203
Figure 85. WLCSP143 marking example (package top view) . . . 204
Figure 86. LQFP144-144-pin, 20 x 20 mm low-profile quad flat package outline . . . 205
Figure 87. LQPF144- 144-pin,20 x 20 mm low-profile quad flat package recommended footprint . . . 207
Figure 88. LQFP144 marking example (package top view) . . . 208
Figure 89. LQFP176 - 176-pin, 24 x 24 mm low-profile quad flat package outline . . . 209

Figure 90. LQFP176 - 176-pin, 24 x 24 mm low profile quad flat recommended footprint. 211
Figure 91. LQFP176 marking (package top view) 212
Figure 92. LQFP208 - 208-pin, 28 x 28 mm low-profile quad flat package outline 213
Figure 93. LQFP208 - 208-pin, 28 x 28 mm low-profile quad flat package recommended footprint 215
Figure 94. LQFP208 marking example (package top view) 216
Figure 95. UFBGA169 - 169-ball 7 x 7 mm 0.50 mm pitch, ultra fine pitch ball grid array package outline 217
Figure 96. UFBGA169 - 169-ball, 7 x 7 mm, 0.50 mm pitch, ultra fine pitch ball grid array recommended footprint 218
Figure 97. UFBGA169 marking example (package top view) 219
Figure 98. UFBGA176+25 - ball 10 x 10 mm, 0.65 mm pitch ultra thin fine pitch ball grid array package outline 220
Figure 99. UFBGA176+25-ball, 10 x 10 mm, 0.65 mm pitch, ultra fine pitch ball grid array package recommended footprint 221
Figure 100. UFBGA176+25 marking example (package top view) 222
Figure 101. TFBGA216 - 216 ball 13 × 13 mm 0.8 mm pitch thin fine pitch ball grid array package outline 223
Figure 102. TFBGA176 marking example (package top view) 224
Figure 103. USB controller configured as peripheral-only and used in Full speed mode 228
Figure 104. USB controller configured as host-only and used in full speed mode. 228
Figure 105. USB controller configured in dual mode and used in full speed mode 229
Figure 106. USB controller configured as peripheral, host, or dual-mode and used in high speed mode. 230
Figure 107. MII mode using a 25 MHz crystal 231
Figure 108. RMII with a 50 MHz oscillator 231
Figure 109. RMII with a 25 MHz crystal and PHY with PLL. 232

1 Introduction

This datasheet provides the description of the STM32F427xx and STM32F429xx line of microcontrollers. For more details on the whole STMicroelectronics STM32 family, please refer to *Section 2.1: Full compatibility throughout the family*.

The STM32F427xx and STM32F429xx datasheet should be read in conjunction with the STM32F4xx reference manual.

For information on the Cortex®-M4 core, please refer to the Cortex®-M4 programming manual (PM0214), available from *www.st.com*.

2 Description

The STM32F427xx and STM32F429xx devices are based on the high-performance Arm® Cortex®-M4 32-bit RISC core operating at a frequency of up to 180 MHz. The Cortex-M4 core features a Floating point unit (FPU) single precision which supports all Arm® single-precision data-processing instructions and data types. It also implements a full set of DSP instructions and a memory protection unit (MPU) which enhances application security.

The STM32F427xx and STM32F429xx devices incorporate high-speed embedded memories (Flash memory up to 2 Mbyte, up to 256 Kbytes of SRAM), up to 4 Kbytes of backup SRAM, and an extensive range of enhanced I/Os and peripherals connected to two APB buses, two AHB buses and a 32-bit multi-AHB bus matrix.

All devices offer three 12-bit ADCs, two DACs, a low-power RTC, twelve general-purpose 16-bit timers including two PWM timers for motor control, two general-purpose 32-bit timers. They also feature standard and advanced communication interfaces.

- Up to three I^2Cs
- Six SPIs, two I^2Ss full duplex. To achieve audio class accuracy, the I^2S peripherals can be clocked via a dedicated internal audio PLL or via an external clock to allow synchronization.
- Four USARTs plus four UARTs
- An USB OTG full-speed and a USB OTG high-speed with full-speed capability (with the ULPI),
- Two CANs
- One SAI serial audio interface
- An SDIO/MMC interface
- Ethernet and camera interface
- LCD-TFT display controller
- Chrom-ART Accelerator™.

Advanced peripherals include an SDIO, a flexible memory control (FMC) interface, a camera interface for CMOS sensors. Refer to *Table 2: STM32F427xx and STM32F429xx features and peripheral counts* for the list of peripherals available on each part number.

The STM32F427xx and STM32F429xx devices operates in the –40 to +105 °C temperature range from a 1.7 to 3.6 V power supply.

The supply voltage can drop to 1.7 V with the use of an external power supply supervisor (refer to *Section 3.17.2: Internal reset OFF*). A comprehensive set of power-saving mode allows the design of low-power applications.

The STM32F427xx and STM32F429xx devices offer devices in 8 packages ranging from 100 pins to 216 pins. The set of included peripherals changes with the device chosen.

These features make the STM32F427xx and STM32F429xx microcontrollers suitable for a wide range of applications:

- Motor drive and application control
- Medical equipment
- Industrial applications: PLC, inverters, circuit breakers
- Printers, and scanners
- Alarm systems, video intercom, and HVAC
- Home audio appliances

Figure 4 shows the general block diagram of the device family.

arm

Table 2. STM32F427xx and STM32F429xx features and peripheral counts

Peripherals		STM32F427 Vx		STM32F429Vx			STM32F427 Zx		STM32F429Zx			STM32F427 Ax		STM32F429 Ax		STM32F427 Ix		STM32F429Ix			STM32F429Bx			STM32F429Nx		
Flash memory in Kbytes		1024	2048	512	1024	2048	1024	2048	512	1024	2048	1024	2048	1024	2048	1024	2048	512	1024	2048	512	1024	2048	512	1024	2048
SRAM in Kbytes	System	256(112+16+64+64)																								
	Backup	4																								
FMC memory controller		Yes(1)																								
Ethernet		Yes																								
Timers	General-purpose	10																								
	Advanced-control	2																								
	Basic	2																								
Random number generator		Yes																								
Communication interfaces	SPI / I^2S	4/2 (full duplex)(2)					6/2 (full duplex)(2)																			
	I^2C	3																								
	USART/ UART	4/4																								
	USB OTG FS	Yes																								
	USB OTG HS	Yes																								
	CAN	2																								
	SAI	1																								
	SDIO	Yes																								
Camera interface		Yes																								
LCD-TFT (STM32F429xx only)		No		Yes			No		Yes			No		Yes		No		Yes								
Chrom-ART Accelerator™		Yes																								
GPIOs		82					114					130				140					168					
12-bit ADC Number of channels		3																								
		16					24																			

Table 2. STM32F427xx and STM32F429xx features and peripheral counts (continued)

Peripherals	STM32F427 Vx	STM32F429Vx	STM32F427 Zx	STM32F429Zx	STM32F427 Ax	STM32F429 Ax	STM32F427 Ix	STM32F429Ix	STM32F429Bx	STM32F429Nx
12-bit DAC Number of channels	Yes 2									
Maximum CPU frequency	180 MHz									
Operating voltage	1.8 to 3.6 V(3)									
Operating temperatures	Ambient temperatures: –40 to +85 °C /–40 to +105 °C									
	Junction temperature: –40 to + 125 °C									
Packages	LQFP100		WLCSP143 LQFP144		UFBGA169		UFBGA176 LQFP176		LQFP208	TFBGA216

1. For the LQFP100 package, only FMC Bank1 or Bank2 are available. Bank1 can only support a multiplexed NOR/PSRAM memory using the NE1 Chip Select. Bank2 can only support a 16- or 8-bit NAND Flash memory using the NCE2 Chip Select. The interrupt line cannot be used since Port G is not available in this package. For UFBGA169 package, only SDRAM, NAND and multiplexed static memories are supported.
2. The SPI2 and SPI3 interfaces give the flexibility to work in an exclusive way in either the SPI mode or the I2S audio mode.
3. V_{DD}/V_{DDA} minimum value of 1.7 V is obtained when the device operates in reduced temperature range, and with the use of an external power supply supervisor (refer to *Section 3.17.2: Internal reset OFF)*.

2.1 Full compatibility throughout the family

The STM32F427xx and STM32F429xx devices are part of the STM32F4 family. They are fully pin-to-pin, software and feature compatible with the STM32F2xx devices, allowing the user to try different memory densities, peripherals, and performances (FPU, higher frequency) for a greater degree of freedom during the development cycle.

The STM32F427xx and STM32F429xx devices maintain a close compatibility with the whole STM32F10xx family. All functional pins are pin-to-pin compatible. The STM32F427xx and STM32F429xx, however, are not drop-in replacements for the STM32F10xx devices: the two families do not have the same power scheme, and so their power pins are different. Nonetheless, transition from the STM32F10xx to the STM32F42x family remains simple as only a few pins are impacted.

Figure 1, *Figure 2*, and *Figure 3*, give compatible board designs between the STM32F4xx, STM32F2xx, and STM32F10xx families.

Figure 1. Compatible board design STM32F10xx/STM32F2xx/STM32F4xx for LQFP100 package

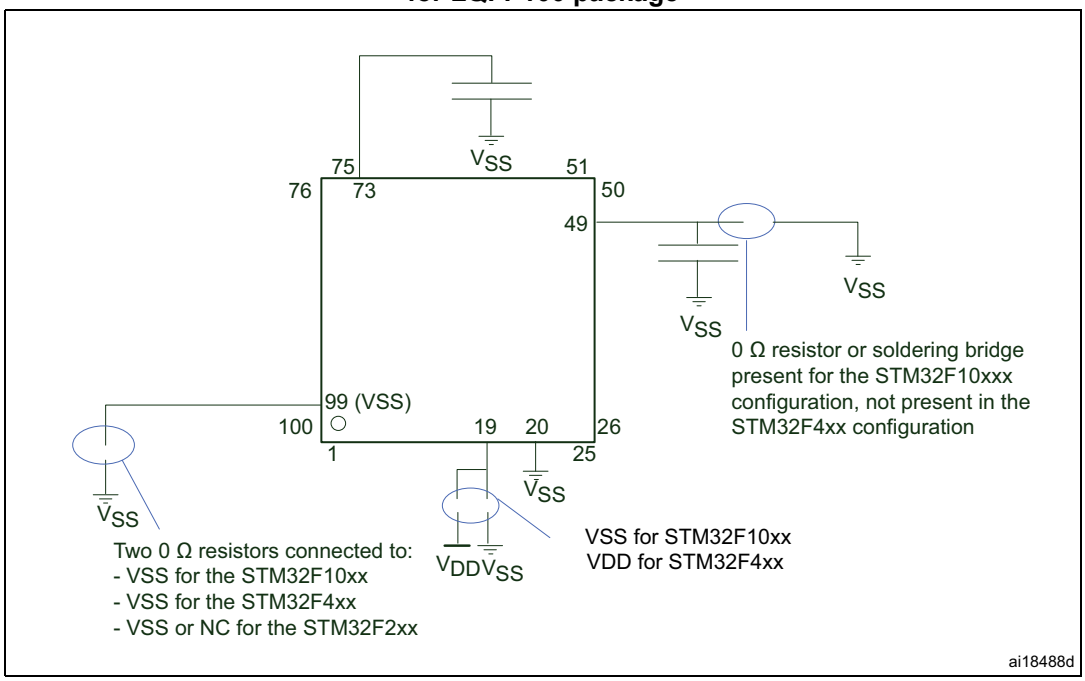


Figure 2. Compatible board design between STM32F10xx/STM32F2xx/STM32F4xx for LQFP144 package

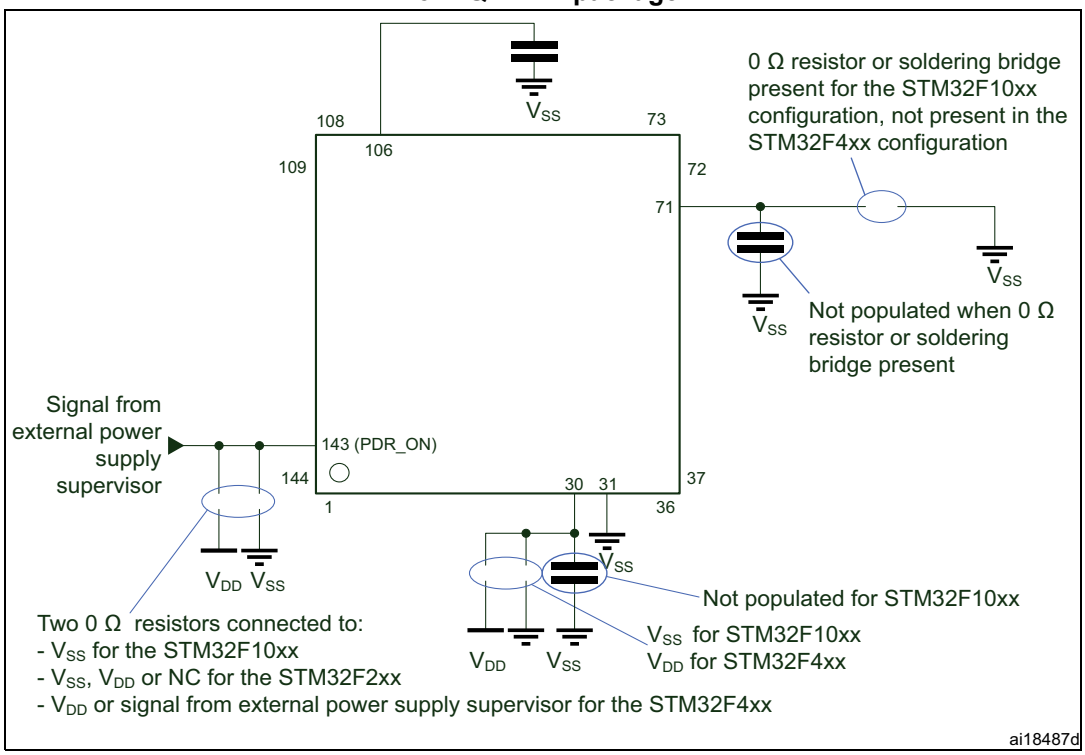


Figure 3. Compatible board design between STM32F2xx and STM32F4xx for LQFP176 and UFBGA176 packages

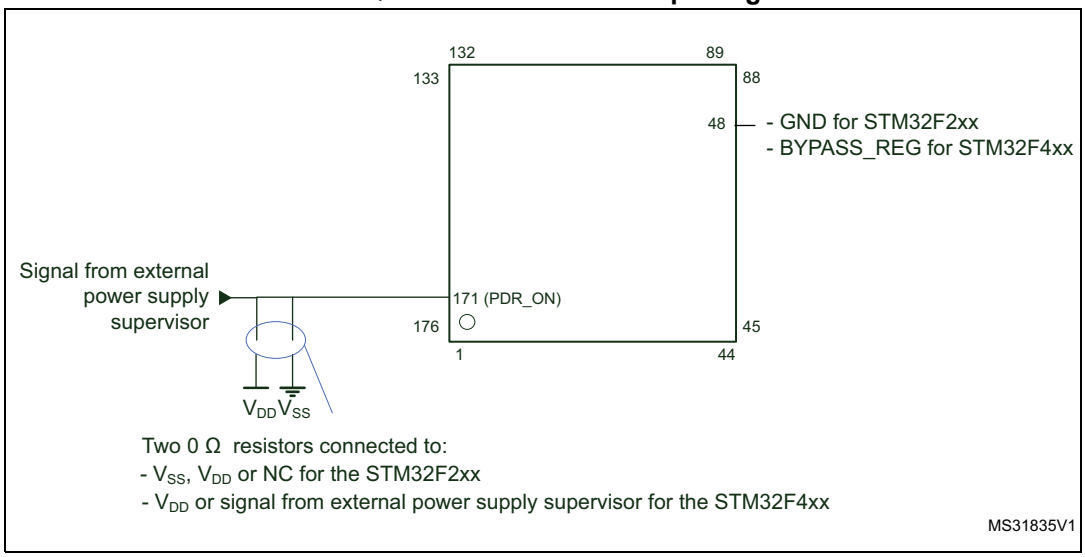
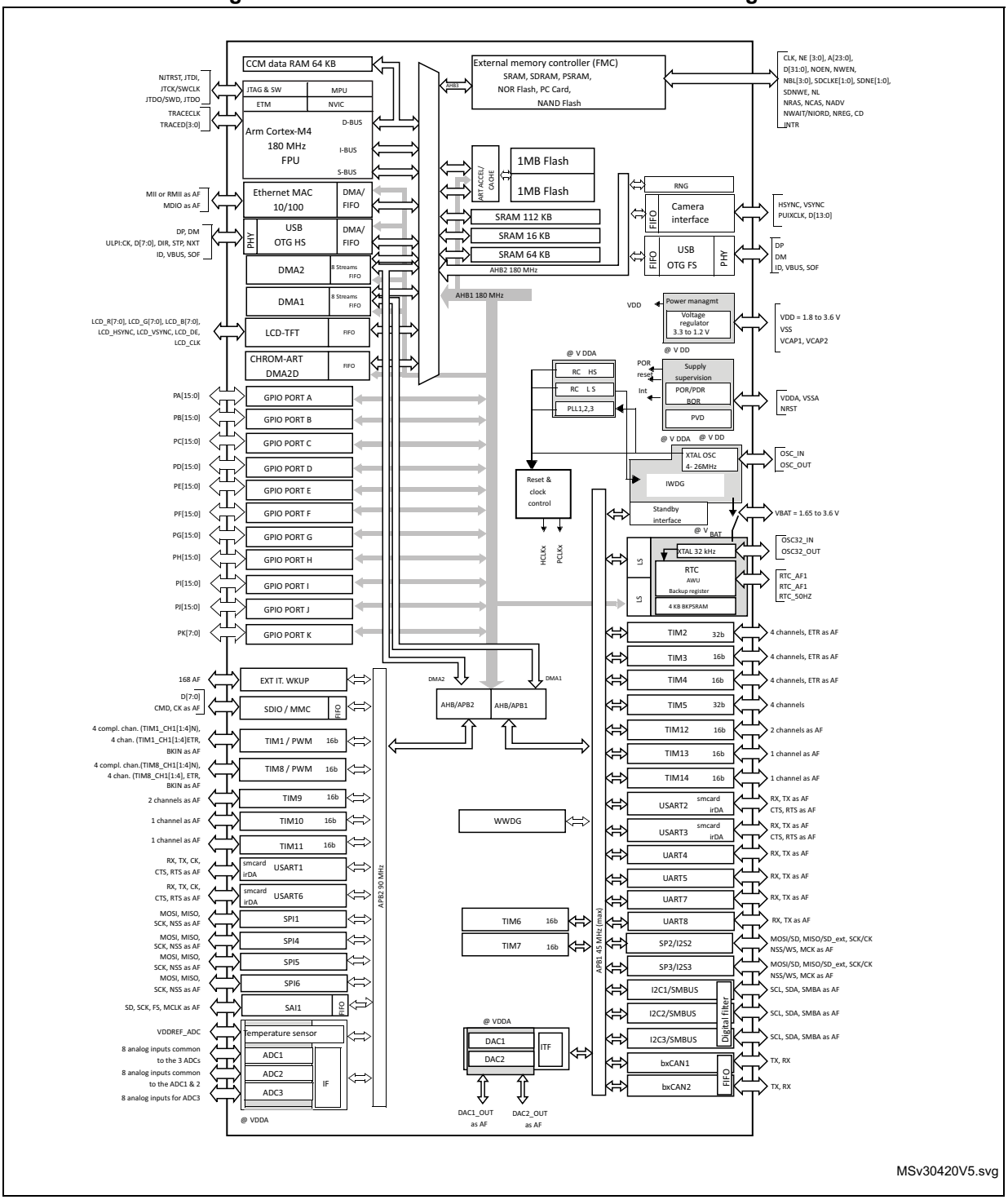


Figure 4. STM32F427xx and STM32F429xx block diagram



1. The timers connected to APB2 are clocked from TIMxCLK up to 180 MHz, while the timers connected to APB1 are clocked from TIMxCLK either up to 90 MHz or 180 MHz depending on TIMPRE bit configuration in the RCC_DCKCFGR register.
2. The LCD-TFT is available only on STM32F429xx devices.

3 Functional overview

3.1 Arm® Cortex®-M4 with FPU and embedded Flash and SRAM

The Arm® Cortex®-M4 with FPU processor is the latest generation of Arm processors for embedded systems. It was developed to provide a low-cost platform that meets the needs of MCU implementation, with a reduced pin count and low-power consumption, while delivering outstanding computational performance and an advanced response to interrupts.

The Arm® Cortex®-M4 with FPU core is a 32-bit RISC processor that features exceptional code-efficiency, delivering the high-performance expected from an Arm core in the memory size usually associated with 8- and 16-bit devices.

The processor supports a set of DSP instructions which allow efficient signal processing and complex algorithm execution.

Its single precision FPU (floating point unit) speeds up software development by using metalanguage development tools, while avoiding saturation.

The STM32F42x family is compatible with all Arm tools and software.

Figure 4 shows the general block diagram of the STM32F42x family.

Note: Cortex-M4 with FPU core is binary compatible with the Cortex-M3 core.

3.2 Adaptive real-time memory accelerator (ART Accelerator™)

The ART Accelerator™ is a memory accelerator which is optimized for STM32 industry-standard Arm® Cortex®-M4 with FPU processors. It balances the inherent performance advantage of the Arm® Cortex®-M4 with FPU over Flash memory technologies, which normally requires the processor to wait for the Flash memory at higher frequencies.

To release the processor full 225 DMIPS performance at this frequency, the accelerator implements an instruction prefetch queue and branch cache, which increases program execution speed from the 128-bit Flash memory. Based on CoreMark benchmark, the performance achieved thanks to the ART Accelerator is equivalent to 0 wait state program execution from Flash memory at a CPU frequency up to 180 MHz.

3.3 Memory protection unit

The memory protection unit (MPU) is used to manage the CPU accesses to memory to prevent one task to accidentally corrupt the memory or resources used by any other active task. This memory area is organized into up to 8 protected areas that can in turn be divided up into 8 subareas. The protection area sizes are between 32 bytes and the whole 4 gigabytes of addressable memory.

The MPU is especially helpful for applications where some critical or certified code has to be protected against the misbehavior of other tasks. It is usually managed by an RTOS (real-time operating system). If a program accesses a memory location that is prohibited by the MPU, the RTOS can detect it and take action. In an RTOS environment, the kernel can dynamically update the MPU area setting, based on the process to be executed.

The MPU is optional and can be bypassed for applications that do not need it.

3.4 Embedded Flash memory

The devices embed a Flash memory of up to 2 Mbytes available for storing programs and data.

3.5 CRC (cyclic redundancy check) calculation unit

The CRC (cyclic redundancy check) calculation unit is used to get a CRC code from a 32-bit data word and a fixed generator polynomial.

Among other applications, CRC-based techniques are used to verify data transmission or storage integrity. In the scope of the EN/IEC 60335-1 standard, they offer a means of verifying the Flash memory integrity. The CRC calculation unit helps compute a software signature during runtime, to be compared with a reference signature generated at link-time and stored at a given memory location.

3.6 Embedded SRAM

All devices embed:

- Up to 256Kbytes of system SRAM including 64 Kbytes of CCM (core coupled memory) data RAM

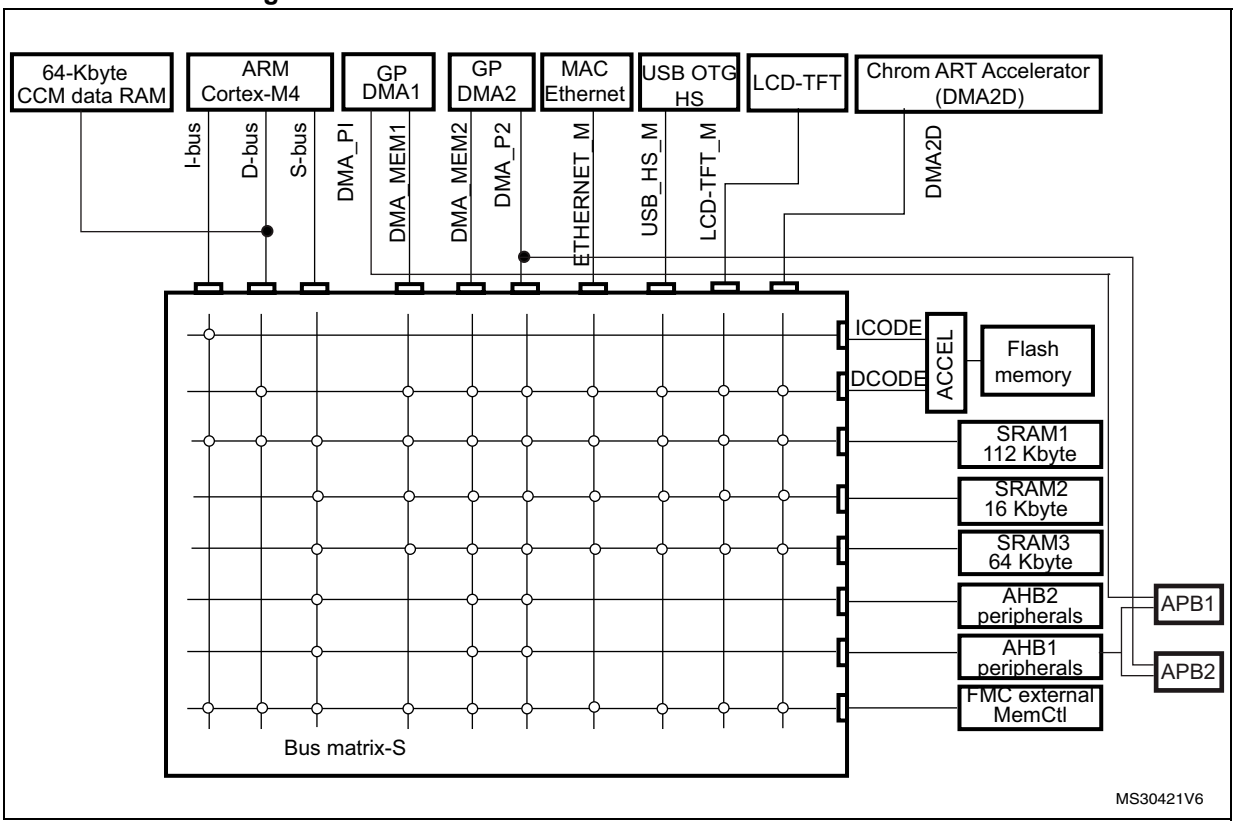
  RAM memory is accessed (read/write) at CPU clock speed with 0 wait states.
- 4 Kbytes of backup SRAM

  This area is accessible only from the CPU. Its content is protected against possible unwanted write accesses, and is retained in Standby or V_{BAT} mode.

3.7 Multi-AHB bus matrix

The 32-bit multi-AHB bus matrix interconnects all the masters (CPU, DMAs, Ethernet, USB HS, LCD-TFT, and DMA2D) and the slaves (Flash memory, RAM, FMC, AHB and APB peripherals) and ensures a seamless and efficient operation even when several high-speed peripherals work simultaneously.

Figure 5. STM32F427xx and STM32F429xx Multi-AHB matrix



3.8 DMA controller (DMA)

The devices feature two general-purpose dual-port DMAs (DMA1 and DMA2) with 8 streams each. They are able to manage memory-to-memory, peripheral-to-memory and memory-to-peripheral transfers. They feature dedicated FIFOs for APB/AHB peripherals, support burst transfer and are designed to provide the maximum peripheral bandwidth (AHB/APB).

The two DMA controllers support circular buffer management, so that no specific code is needed when the controller reaches the end of the buffer. The two DMA controllers also have a double buffering feature, which automates the use and switching of two memory buffers without requiring any special code.

Each stream is connected to dedicated hardware DMA requests, with support for software trigger on each stream. Configuration is made by software and transfer sizes between source and destination are independent.

The DMA can be used with the main peripherals:

- SPI and I^2S
- I^2C
- USART
- General-purpose, basic and advanced-control timers TIMx
- DAC
- SDIO
- Camera interface (DCMI)
- ADC
- SAI1.

3.9 Flexible memory controller (FMC)

All devices embed an FMC. It has four Chip Select outputs supporting the following modes: PCCard/Compact Flash, SDRAM/LPSDR SDRAM, SRAM, PSRAM, NOR Flash and NAND Flash.

Functionality overview:

- 8-,16-, 32-bit data bus width
- Read FIFO for SDRAM controller
- Write FIFO
- Maximum FMC_CLK/FMC_SDCLK frequency for synchronous accesses is 90 MHz.

LCD parallel interface

The FMC can be configured to interface seamlessly with most graphic LCD controllers. It supports the Intel 8080 and Motorola 6800 modes, and is flexible enough to adapt to specific LCD interfaces. This LCD parallel interface capability makes it easy to build cost-effective graphic applications using LCD modules with embedded controllers or high performance solutions using external controllers with dedicated acceleration.

3.10 LCD-TFT controller (available only on STM32F429xx)

The LCD-TFT display controller provides a 24-bit parallel digital RGB (Red, Green, Blue) and delivers all signals to interface directly to a broad range of LCD and TFT panels up to XGA (1024x768) resolution with the following features:

- 2 displays layers with dedicated FIFO (64x32-bit)
- Color Look-Up table (CLUT) up to 256 colors (256x24-bit) per layer
- Up to 8 Input color formats selectable per layer
- Flexible blending between two layers using alpha value (per pixel or constant)
- Flexible programmable parameters for each layer
- Color keying (transparency color)
- Up to 4 programmable interrupt events.

3.11 Chrom-ART Accelerator™ (DMA2D)

The Chrom-Art Accelerator™ (DMA2D) is a graphic accelerator which offers advanced bit blitting, row data copy and pixel format conversion. It supports the following functions:

- Rectangle filling with a fixed color
- Rectangle copy
- Rectangle copy with pixel format conversion
- Rectangle composition with blending and pixel format conversion.

Various image format coding are supported, from indirect 4bpp color mode up to 32bpp direct color. It embeds dedicated memory to store color lookup tables.

An interrupt can be generated when an operation is complete or at a programmed watermark.

All the operations are fully automatized and are running independently from the CPU or the DMAs.

3.12 Nested vectored interrupt controller (NVIC)

The devices embed a nested vectored interrupt controller able to manage 16 priority levels, and handle up to 91 maskable interrupt channels plus the 16 interrupt lines of the Cortex®-M4 with FPU core.

- Closely coupled NVIC gives low-latency interrupt processing
- Interrupt entry vector table address passed directly to the core
- Allows early processing of interrupts
- Processing of late arriving, higher-priority interrupts
- Support tail chaining
- Processor state automatically saved
- Interrupt entry restored on interrupt exit with no instruction overhead

This hardware block provides flexible interrupt management features with minimum interrupt latency.

3.13 External interrupt/event controller (EXTI)

The external interrupt/event controller consists of 23 edge-detector lines used to generate interrupt/event requests. Each line can be independently configured to select the trigger event (rising edge, falling edge, both) and can be masked independently. A pending register maintains the status of the interrupt requests. The EXTI can detect an external line with a pulse width shorter than the Internal APB2 clock period. Up to 168 GPIOs can be connected to the 16 external interrupt lines.

3.14 Clocks and startup

On reset the 16 MHz internal RC oscillator is selected as the default CPU clock. The 16 MHz internal RC oscillator is factory-trimmed to offer 1% accuracy over the full temperature range. The application can then select as system clock either the RC oscillator or an external 4-26 MHz clock source. This clock can be monitored for failure. If a failure is

detected, the system automatically switches back to the internal RC oscillator and a software interrupt is generated (if enabled). This clock source is input to a PLL thus allowing to increase the frequency up to 180 MHz. Similarly, full interrupt management of the PLL clock entry is available when necessary (for example if an indirectly used external oscillator fails).

Several prescalers allow the configuration of the two AHB buses, the high-speed APB (APB2) and the low-speed APB (APB1) domains. The maximum frequency of the two AHB buses is 180 MHz while the maximum frequency of the high-speed APB domains is 90 MHz. The maximum allowed frequency of the low-speed APB domain is 45 MHz.

The devices embed a dedicated PLL (PLLI2S) and PLLSAI which allows to achieve audio class performance. In this case, the I^2S master clock can generate all standard sampling frequencies from 8 kHz to 192 kHz.

3.15 Boot modes

At startup, boot pins are used to select one out of three boot options:

- Boot from user Flash
- Boot from system memory
- Boot from embedded SRAM

The boot loader is located in system memory. It is used to reprogram the Flash memory through a serial interface. Refer to application note AN2606 for details.

3.16 Power supply schemes

- V_{DD} = 1.7 to 3.6 V: external power supply for I/Os and the internal regulator (when enabled), provided externally through V_{DD} pins.
- V_{SSA}, V_{DDA} = 1.7 to 3.6 V: external analog power supplies for ADC, DAC, Reset blocks, RCs and PLL. V_{DDA} and V_{SSA} must be connected to V_{DD} and V_{SS}, respectively.
- V_{BAT} = 1.65 to 3.6 V: power supply for RTC, external clock 32 kHz oscillator and backup registers (through power switch) when V_{DD} is not present.

Note: V_{DD}/V_{DDA} minimum value of 1.7 V is obtained with the use of an external power supply supervisor (refer to Section 3.17.2: Internal reset OFF). Refer to Table 3: Voltage regulator configuration mode versus device operating mode to identify the packages supporting this option.

3.17 Power supply supervisor

3.17.1 Internal reset ON

On packages embedding the PDR_ON pin, the power supply supervisor is enabled by holding PDR_ON high. On the other package, the power supply supervisor is always enabled.

The device has an integrated power-on reset (POR)/ power-down reset (PDR) circuitry coupled with a Brownout reset (BOR) circuitry. At power-on, POR/PDR is always active and ensures proper operation starting from 1.8 V. After the 1.8 V POR threshold level is

reached, the option byte loading process starts, either to confirm or modify default BOR thresholds, or to disable BOR permanently. Three BOR thresholds are available through option bytes. The device remains in reset mode when V_{DD} is below a specified threshold, $V_{POR/PDR}$ or V_{BOR}, without the need for an external reset circuit.

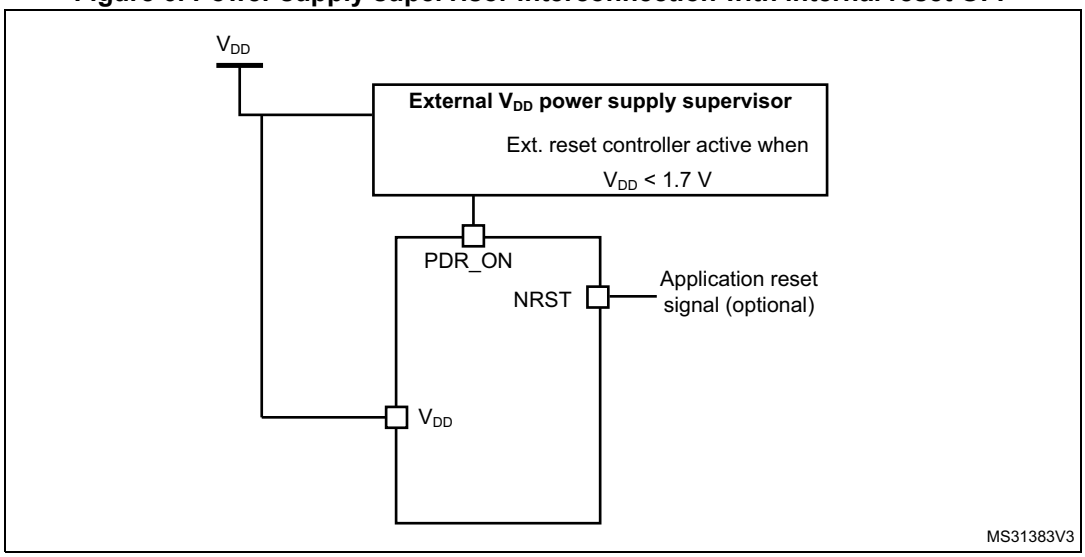
The device also features an embedded programmable voltage detector (PVD) that monitors the V_{DD}/V_{DDA} power supply and compares it to the V_{PVD} threshold. An interrupt can be generated when V_{DD}/V_{DDA} drops below the V_{PVD} threshold and/or when V_{DD}/V_{DDA} is higher than the V_{PVD} threshold. The interrupt service routine can then generate a warning message and/or put the MCU into a safe state. The PVD is enabled by software.

3.17.2 Internal reset OFF

This feature is available only on packages featuring the PDR_ON pin. The internal power-on reset (POR) / power-down reset (PDR) circuitry is disabled through the PDR_ON pin.

An external power supply supervisor should monitor V_{DD} and should maintain the device in reset mode as long as V_{DD} is below a specified threshold. PDR_ON should be connected to this external power supply supervisor. Refer to *Figure 6: Power supply supervisor interconnection with internal reset OFF*.

Figure 6. Power supply supervisor interconnection with internal reset OFF



The V_{DD} specified threshold, below which the device must be maintained under reset, is 1.7 V (see *Figure 7*).

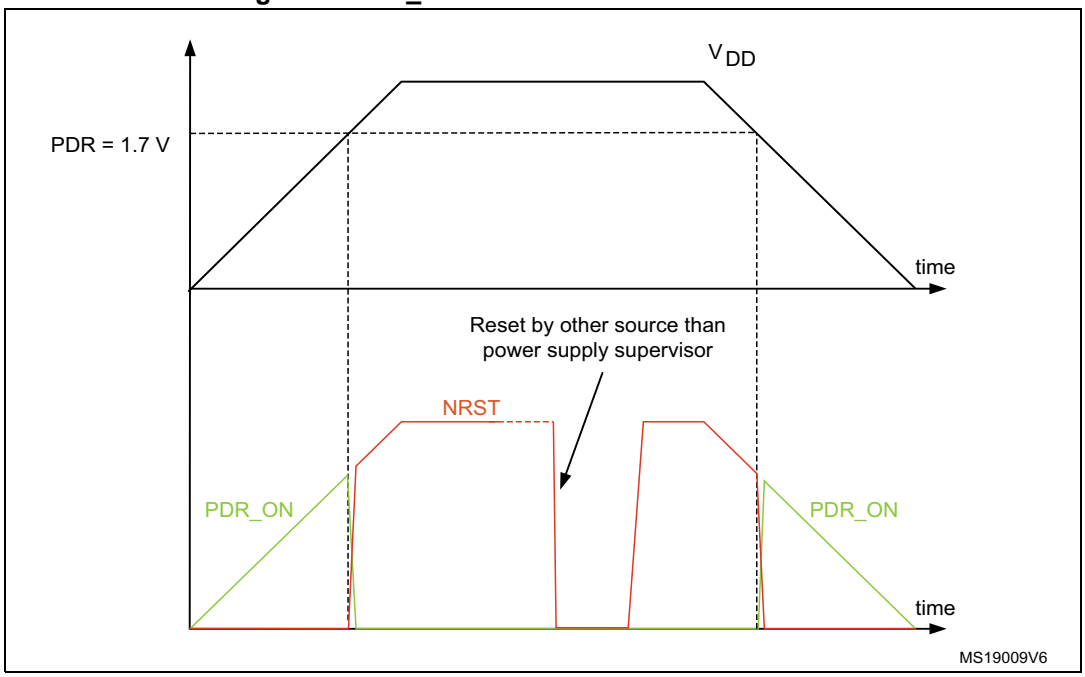
A comprehensive set of power-saving mode allows to design low-power applications.

When the internal reset is OFF, the following integrated features are no more supported:

- The integrated power-on reset (POR) / power-down reset (PDR) circuitry is disabled
- The brownout reset (BOR) circuitry must be disabled
- The embedded programmable voltage detector (PVD) is disabled
- V_{BAT} functionality is no more available and V_{BAT} pin should be connected to V_{DD}.

All packages, except for the LQFP100, allow to disable the internal reset through the PDR_ON signal.

Figure 7. PDR_ON control with internal reset OFF



3.18 Voltage regulator

The regulator has four operating modes:

- Regulator ON
  - Main regulator mode (MR)
  - Low power regulator (LPR)
  - Power-down
- Regulator OFF

3.18.1 Regulator ON

On packages embedding the BYPASS_REG pin, the regulator is enabled by holding BYPASS_REG low. On all other packages, the regulator is always enabled.

There are three power modes configured by software when the regulator is ON:

- MR mode used in Run/sleep modes or in Stop modes
  - In Run/Sleep mode

    The MR mode is used either in the normal mode (default mode) or the over-drive mode (enabled by software). Different voltages scaling are provided to reach the best compromise between maximum frequency and dynamic power consumption.

The over-drive mode allows operating at a higher frequency than the normal mode for a given voltage scaling.

– In Stop modes

The MR can be configured in two ways during stop mode:

MR operates in normal mode (default mode of MR in stop mode)

MR operates in under-drive mode (reduced leakage mode).

- LPR is used in the Stop modes:

  The LP regulator mode is configured by software when entering Stop mode.

  Like the MR mode, the LPR can be configured in two ways during stop mode:

  – LPR operates in normal mode (default mode when LPR is ON)

  – LPR operates in under-drive mode (reduced leakage mode).

- Power-down is used in Standby mode.

  The Power-down mode is activated only when entering in Standby mode. The regulator output is in high impedance and the kernel circuitry is powered down, inducing zero consumption. The contents of the registers and SRAM are lost.

Refer to *Table 3* for a summary of voltage regulator modes versus device operating modes.

Two external ceramic capacitors should be connected on V_{CAP_1} and V_{CAP_2} pin. Refer to *Figure 22: Power supply scheme* and *Table 19: VCAP1/VCAP2 operating conditions*.

All packages have the regulator ON feature.

Table 3. Voltage regulator configuration mode versus device operating mode(1)

Voltage regulator configuration	Run mode	Sleep mode	Stop mode	Standby mode
Normal mode	MR	MR	MR or LPR	-
Over-drive mode(2)	MR	MR	-	-
Under-drive mode	-	-	MR or LPR	-
Power-down mode	-	-	-	Yes

1. '-' means that the corresponding configuration is not available.
2. The over-drive mode is not available when V_{DD} = 1.7 to 2.1 V.

3.18.2 Regulator OFF

This feature is available only on packages featuring the BYPASS_REG pin. The regulator is disabled by holding BYPASS_REG high. The regulator OFF mode allows to supply externally a V_{12} voltage source through V_{CAP_1} and V_{CAP_2} pins.

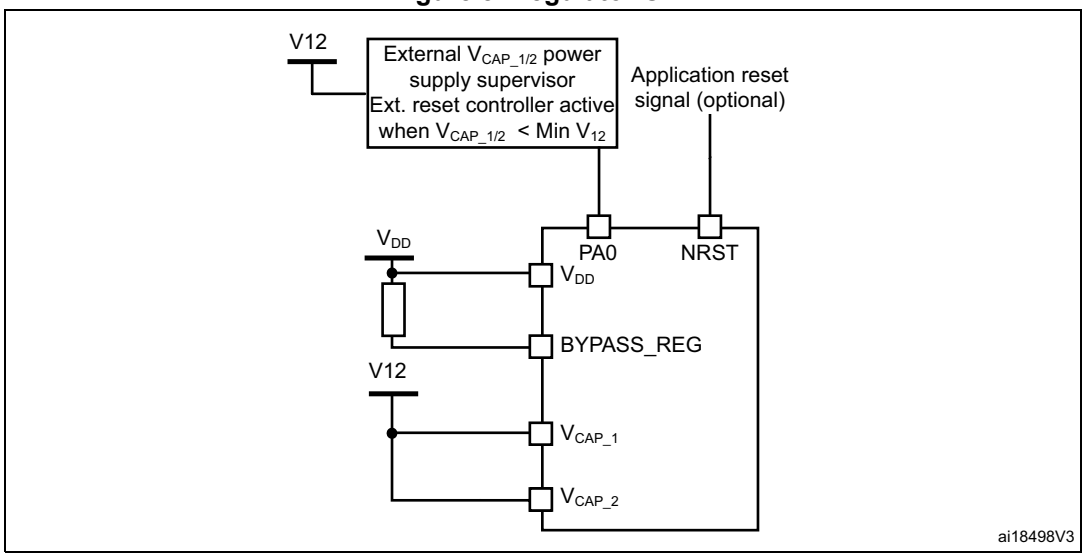
Since the internal voltage scaling is not managed internally, the external voltage value must be aligned with the targeted maximum frequency. Refer to *Table 17: General operating conditions*.The two 2.2 µF ceramic capacitors should be replaced by two 100 nF decoupling capacitors. Refer to *Figure 22: Power supply scheme*.

When the regulator is OFF, there is no more internal monitoring on V_{12}. An external power supply supervisor should be used to monitor the V_{12} of the logic power domain. PA0 pin should be used for this purpose, and act as power-on reset on V_{12} power domain.

In regulator OFF mode, the following features are no more supported:

- PA0 cannot be used as a GPIO pin since it allows to reset a part of the V_{12} logic power domain which is not reset by the NRST pin.
- As long as PA0 is kept low, the debug mode cannot be used under power-on reset. As a consequence, PA0 and NRST pins must be managed separately if the debug connection under reset or pre-reset is required.
- The over-drive and under-drive modes are not available.
- The Standby mode is not available.

Figure 8. Regulator OFF

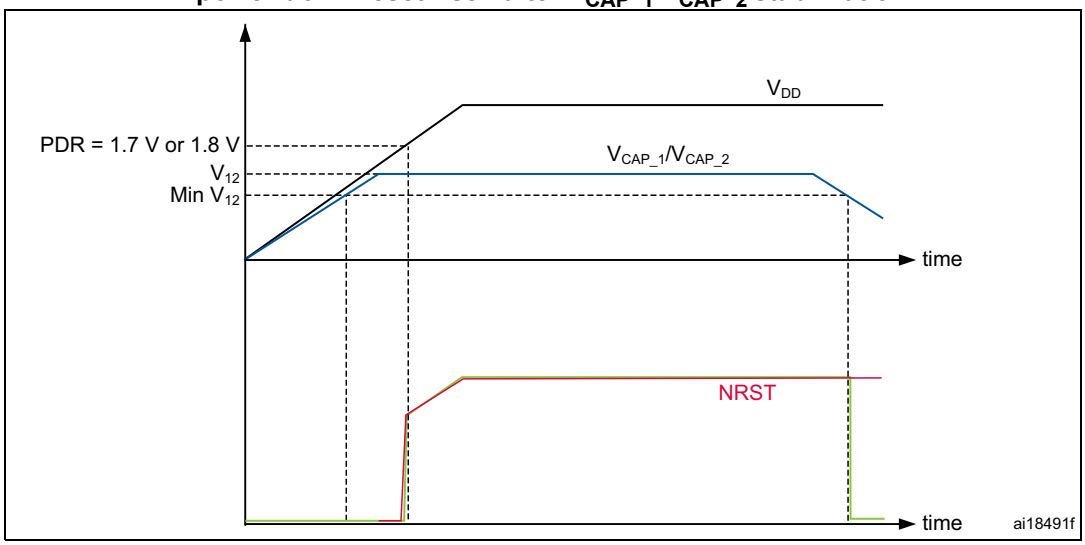


The following conditions must be respected:

- V_{DD} should always be higher than V_{CAP_1} and V_{CAP_2} to avoid current injection between power domains.
- If the time for V_{CAP_1} and V_{CAP_2} to reach V_{12} minimum value is faster than the time for V_{DD} to reach 1.7 V, then PA0 should be kept low to cover both conditions: until V_{CAP_1} and V_{CAP_2} reach V_{12} minimum value and until V_{DD} reaches 1.7 V (see *Figure 9*).
- Otherwise, if the time for V_{CAP_1} and V_{CAP_2} to reach V_{12} minimum value is slower than the time for V_{DD} to reach 1.7 V, then PA0 could be asserted low externally (see *Figure 10*).
- If V_{CAP_1} and V_{CAP_2} go below V_{12} minimum value and V_{DD} is higher than 1.7 V, then a reset must be asserted on PA0 pin.

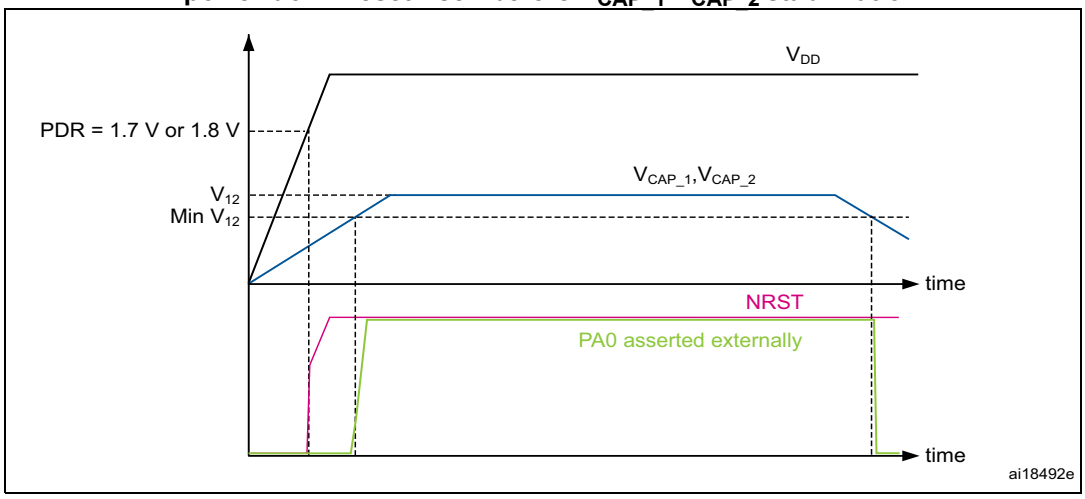
Note: The minimum value of V_{12} depends on the maximum frequency targeted in the application (see Table 17: General operating conditions).

Figure 9. Startup in regulator OFF: slow V_{DD} slope - power-down reset risen after V_{CAP_1}/V_{CAP_2} stabilization



1. This figure is valid whatever the internal reset mode (ON or OFF).

Figure 10. Startup in regulator OFF mode: fast V_{DD} slope - power-down reset risen before V_{CAP_1}/V_{CAP_2} stabilization



1. This figure is valid whatever the internal reset mode (ON or OFF).

3.18.3 Regulator ON/OFF and internal reset ON/OFF availability

Table 4. Regulator ON/OFF and internal reset ON/OFF availability

Package	Regulator ON	Regulator OFF	Internal reset ON	Internal reset OFF
LQFP100	Yes	No	Yes	No
LQFP144, LQFP208			Yes PDR_ON set to V_{DD}	Yes PDR_ON connected to an external power supply supervisor
WLCSP143, LQFP176, UFBGA169, UFBGA176, TFBGA216	Yes BYPASS_REG set to V_{SS}	Yes BYPASS_REG set to V_{DD}		

3.19 Real-time clock (RTC), backup SRAM and backup registers

The backup domain includes:

- The real-time clock (RTC)
- 4 Kbytes of backup SRAM
- 20 backup registers

The real-time clock (RTC) is an independent BCD timer/counter. Dedicated registers contain the second, minute, hour (in 12/24 hour), week day, date, month, year, in BCD (binary-coded decimal) format. Correction for 28, 29 (leap year), 30, and 31 day of the month are performed automatically. The RTC provides a programmable alarm and programmable periodic interrupts with wakeup from Stop and Standby modes. The sub-seconds value is also available in binary format.

It is clocked by a 32.768 kHz external crystal, resonator or oscillator, the internal low-power RC oscillator or the high-speed external clock divided by 128. The internal low-speed RC has a typical frequency of 32 kHz. The RTC can be calibrated using an external 512 Hz output to compensate for any natural quartz deviation.

Two alarm registers are used to generate an alarm at a specific time and calendar fields can be independently masked for alarm comparison. To generate a periodic interrupt, a 16-bit programmable binary auto-reload downcounter with programmable resolution is available and allows automatic wakeup and periodic alarms from every 120 µs to every 36 hours.

A 20-bit prescaler is used for the time base clock. It is by default configured to generate a time base of 1 second from a clock at 32.768 kHz.

The 4-Kbyte backup SRAM is an EEPROM-like memory area. It can be used to store data which need to be retained in VBAT and standby mode. This memory area is disabled by default to minimize power consumption (see *Section 3.20: Low-power modes*). It can be enabled by software.

The backup registers are 32-bit registers used to store 80 bytes of user application data when V_{DD} power is not present. Backup registers are not reset by a system, a power reset, or when the device wakes up from the Standby mode (see *Section 3.20: Low-power modes*).

Additional 32-bit registers contain the programmable alarm subseconds, seconds, minutes, hours, day, and date.

Like backup SRAM, the RTC and backup registers are supplied through a switch that is powered either from the V_{DD} supply when present or from the V_{BAT} pin.

3.20 Low-power modes

The devices support three low-power modes to achieve the best compromise between low power consumption, short startup time and available wakeup sources:

- **Sleep mode**

  In Sleep mode, only the CPU is stopped. All peripherals continue to operate and can wake up the CPU when an interrupt/event occurs.

- **Stop mode**

  The Stop mode achieves the lowest power consumption while retaining the contents of SRAM and registers. All clocks in the 1.2 V domain are stopped, the PLL, the HSI RC and the HSE crystal oscillators are disabled.

  The voltage regulator can be put either in main regulator mode (MR) or in low-power mode (LPR). Both modes can be configured as follows (see *Table 5: Voltage regulator modes in stop mode*):

  – Normal mode (default mode when MR or LPR is enabled)

  – Under-drive mode.

  The device can be woken up from the Stop mode by any of the EXTI line (the EXTI line source can be one of the 16 external lines, the PVD output, the RTC alarm / wakeup / tamper / time stamp events, the USB OTG FS/HS wakeup or the Ethernet wakeup).

Table 5. Voltage regulator modes in stop mode

Voltage regulator configuration	Main regulator (MR)	Low-power regulator (LPR)
Normal mode	MR ON	LPR ON
Under-drive mode	MR in under-drive mode	LPR in under-drive mode

- **Standby mode**

  The Standby mode is used to achieve the lowest power consumption. The internal voltage regulator is switched off so that the entire 1.2 V domain is powered off. The PLL, the HSI RC and the HSE crystal oscillators are also switched off. After entering Standby mode, the SRAM and register contents are lost except for registers in the backup domain and the backup SRAM when selected.

  The device exits the Standby mode when an external reset (NRST pin), an IWDG reset, a rising edge on the WKUP pin, or an RTC alarm / wakeup / tamper /time stamp event occurs.

  The standby mode is not supported when the embedded voltage regulator is bypassed and the 1.2 V domain is controlled by an external power.

3.21 V_{BAT} operation

The V_{BAT} pin allows to power the device V_{BAT} domain from an external battery, an external supercapacitor, or from V_{DD} when no external battery and an external supercapacitor are present.

V_{BAT} operation is activated when V_{DD} is not present.

The V_{BAT} pin supplies the RTC, the backup registers and the backup SRAM.

Note: When the microcontroller is supplied from V_{BAT}, external interrupts and RTC alarm/events do not exit it from V_{BAT} operation.

When PDR_ON pin is not connected to V_{DD} (Internal Reset OFF), the V_{BAT} functionality is no more available and V_{BAT} pin should be connected to V_{DD}.

3.22 Timers and watchdogs

The devices include two advanced-control timers, eight general-purpose timers, two basic timers and two watchdog timers.

All timer counters can be frozen in debug mode.

Table 6 compares the features of the advanced-control, general-purpose and basic timers.

Table 6. Timer feature comparison

Timer type	Timer	Counter resolution	Counter type	Prescaler factor	DMA request generation	Capture/ compare channels	Complementary output	Max interface clock (MHz)	Max timer clock (MHz) (1)
Advanced -control	TIM1, TIM8	16-bit	Up, Down, Up/down	Any integer between 1 and 65536	Yes	4	Yes	90	180
General purpose	TIM2, TIM5	32-bit	Up, Down, Up/down	Any integer between 1 and 65536	Yes	4	No	45	90/180
	TIM3, TIM4	16-bit	Up, Down, Up/down	Any integer between 1 and 65536	Yes	4	No	45	90/180
	TIM9	16-bit	Up	Any integer between 1 and 65536	No	2	No	90	180
	TIM10, TIM11	16-bit	Up	Any integer between 1 and 65536	No	1	No	90	180
	TIM12	16-bit	Up	Any integer between 1 and 65536	No	2	No	45	90/180
	TIM13, TIM14	16-bit	Up	Any integer between 1 and 65536	No	1	No	45	90/180
Basic	TIM6, TIM7	16-bit	Up	Any integer between 1 and 65536	Yes	0	No	45	90/180

1. The maximum timer clock is either 90 or 180 MHz depending on TIMPRE bit configuration in the RCC_DCKCFGR register.

3.22.1 Advanced-control timers (TIM1, TIM8)

The advanced-control timers (TIM1, TIM8) can be seen as three-phase PWM generators multiplexed on 6 channels. They have complementary PWM outputs with programmable inserted dead times. They can also be considered as complete general-purpose timers. Their 4 independent channels can be used for:

- Input capture
- Output compare
- PWM generation (edge- or center-aligned modes)
- One-pulse mode output

If configured as standard 16-bit timers, they have the same features as the general-purpose TIMx timers. If configured as 16-bit PWM generators, they have full modulation capability (0-100%).

The advanced-control timer can work together with the TIMx timers via the Timer Link feature for synchronization or event chaining.

TIM1 and TIM8 support independent DMA request generation.

3.22.2 General-purpose timers (TIMx)

There are ten synchronizable general-purpose timers embedded in the STM32F42x devices (see *Table 6* for differences).

- **TIM2, TIM3, TIM4, TIM5**

  The STM32F42x include 4 full-featured general-purpose timers: TIM2, TIM5, TIM3, and TIM4.The TIM2 and TIM5 timers are based on a 32-bit auto-reload up/downcounter and a 16-bit prescaler. The TIM3 and TIM4 timers are based on a 16-bit auto-reload up/downcounter and a 16-bit prescaler. They all feature 4 independent channels for input capture/output compare, PWM or one-pulse mode output. This gives up to 16 input capture/output compare/PWMs on the largest packages.

  The TIM2, TIM3, TIM4, TIM5 general-purpose timers can work together, or with the other general-purpose timers and the advanced-control timers TIM1 and TIM8 via the Timer Link feature for synchronization or event chaining.

  Any of these general-purpose timers can be used to generate PWM outputs.

  TIM2, TIM3, TIM4, TIM5 all have independent DMA request generation. They are capable of handling quadrature (incremental) encoder signals and the digital outputs from 1 to 4 hall-effect sensors.

- **TIM9, TIM10, TIM11, TIM12, TIM13, and TIM14**

  These timers are based on a 16-bit auto-reload upcounter and a 16-bit prescaler. TIM10, TIM11, TIM13, and TIM14 feature one independent channel, whereas TIM9 and TIM12 have two independent channels for input capture/output compare, PWM or one-pulse mode output. They can be synchronized with the TIM2, TIM3, TIM4, TIM5 full-featured general-purpose timers. They can also be used as simple time bases.

3.22.3 Basic timers TIM6 and TIM7

These timers are mainly used for DAC trigger and waveform generation. They can also be used as a generic 16-bit time base.

TIM6 and TIM7 support independent DMA request generation.

3.22.4 Independent watchdog

The independent watchdog is based on a 12-bit downcounter and 8-bit prescaler. It is clocked from an independent 32 kHz internal RC and as it operates independently from the main clock, it can operate in Stop and Standby modes. It can be used either as a watchdog to reset the device when a problem occurs, or as a free-running timer for application timeout management. It is hardware- or software-configurable through the option bytes.

3.22.5 Window watchdog

The window watchdog is based on a 7-bit downcounter that can be set as free-running. It can be used as a watchdog to reset the device when a problem occurs. It is clocked from the main clock. It has an early warning interrupt capability and the counter can be frozen in debug mode.

3.22.6 SysTick timer

This timer is dedicated to real-time operating systems, but could also be used as a standard downcounter. It features:

- A 24-bit downcounter
- Autoreload capability
- Maskable system interrupt generation when the counter reaches 0
- Programmable clock source.

3.23 Inter-integrated circuit interface (I^2C)

Up to three I²C bus interfaces can operate in multimaster and slave modes. They can support the standard (up to 100 KHz), and fast (up to 400 KHz) modes. They support the 7/10-bit addressing mode and the 7-bit dual addressing mode (as slave). A hardware CRC generation/verification is embedded.

They can be served by DMA and they support SMBus 2.0/PMBus.

The devices also include programmable analog and digital noise filters (see *Table 7*).

Table 7. Comparison of I2C analog and digital filters

	Analog filter	Digital filter
Pulse width of suppressed spikes	≥ 50 ns	Programmable length from 1 to 15 I2C peripheral clocks

3.24 Universal synchronous/asynchronous receiver transmitters (USART)

The devices embed four universal synchronous/asynchronous receiver transmitters (USART1, USART2, USART3 and USART6) and four universal asynchronous receiver transmitters (UART4, UART5, UART7, and UART8).

These six interfaces provide asynchronous communication, IrDA SIR ENDEC support, multiprocessor communication mode, single-wire half-duplex communication mode and have LIN Master/Slave capability. The USART1 and USART6 interfaces are able to

communicate at speeds of up to 11.25 Mbit/s. The other available interfaces communicate at up to 5.62 bit/s.

USART1, USART2, USART3 and USART6 also provide hardware management of the CTS and RTS signals, Smart Card mode (ISO 7816 compliant) and SPI-like communication capability. All interfaces can be served by the DMA controller.

Table 8. USART feature comparison(1)

USART name	Standard features	Modem (RTS/CTS)	LIN	SPI master	irDA	Smartcard (ISO 7816)	Max. baud rate in Mbit/s (oversampling by 16)	Max. baud rate in Mbit/s (oversampling by 8)	APB mapping
USART1	X	X	X	X	X	X	5.62	11.25	APB2 (max. 90 MHz)
USART2	X	X	X	X	X	X	2.81	5.62	APB1 (max. 45 MHz)
USART3	X	X	X	X	X	X	2.81	5.62	APB1 (max. 45 MHz)
UART4	X	-	X	-	X	-	2.81	5.62	APB1 (max. 45 MHz)
UART5	X	-	X	-	X	-	2.81	5.62	APB1 (max. 45 MHz)
USART6	X	X	X	X	X	X	5.62	11.25	APB2 (max. 90 MHz)
UART7	X	-	X	-	X	-	2.81	5.62	APB1 (max. 45 MHz)
UART8	X	-	X	-	X	-	2.81	5.62	APB1 (max. 45 MHz)

1. X = feature supported.

3.25 Serial peripheral interface (SPI)

The devices feature up to six SPIs in slave and master modes in full-duplex and simplex communication modes. SPI1, SPI4, SPI5, and SPI6 can communicate at up to 45 Mbits/s, SPI2 and SPI3 can communicate at up to 22.5 Mbit/s. The 3-bit prescaler gives 8 master mode frequencies and the frame is configurable to 8 bits or 16 bits. The hardware CRC generation/verification supports basic SD Card/MMC modes. All SPIs can be served by the DMA controller.

The SPI interface can be configured to operate in TI mode for communications in master mode and slave mode.

3.26 Inter-integrated sound (I^2S)

Two standard I^2S interfaces (multiplexed with SPI2 and SPI3) are available. They can be operated in master or slave mode, in full duplex and simplex communication modes, and can be configured to operate with a 16-/32-bit resolution as an input or output channel. Audio sampling frequencies from 8 kHz up to 192 kHz are supported. When either or both of the I^2S interfaces is/are configured in master mode, the master clock can be output to the external DAC/CODEC at 256 times the sampling frequency.

All I2Sx can be served by the DMA controller.

Note: For I2S2 *full-duplex mode, I2S2_CK and I2S2_WS signals can be used only on GPIO Port B and GPIO Port D.*

3.27 Serial Audio interface (SAI1)

The serial audio interface (SAI1) is based on two independent audio sub-blocks which can operate as transmitter or receiver with their FIFO. Many audio protocols are supported by each block: I2S standards, LSB or MSB-justified, PCM/DSP, TDM, AC'97 and SPDIF output, supporting audio sampling frequencies from 8 kHz up to 192 kHz. Both sub-blocks can be configured in master or in slave mode.

In master mode, the master clock can be output to the external DAC/CODEC at 256 times of the sampling frequency.

The two sub-blocks can be configured in synchronous mode when full-duplex mode is required.

SAI1 can be served by the DMA controller.

3.28 Audio PLL (PLLI2S)

The devices feature an additional dedicated PLL for audio I^2S and SAI applications. It allows to achieve error-free I^2S sampling clock accuracy without compromising on the CPU performance, while using USB peripherals.

The PLLI2S configuration can be modified to manage an I^2S/SAI sample rate change without disabling the main PLL (PLL) used for CPU, USB and Ethernet interfaces.

The audio PLL can be programmed with very low error to obtain sampling rates ranging from 8 KHz to 192 KHz.

In addition to the audio PLL, a master clock input pin can be used to synchronize the I^2S/SAI flow with an external PLL (or Codec output).

3.29 Audio and LCD PLL(PLLSAI)

An additional PLL dedicated to audio and LCD-TFT is used for SAI1 peripheral in case the PLLI2S is programmed to achieve another audio sampling frequency (49.152 MHz or 11.2896 MHz) and the audio application requires both sampling frequencies simultaneously.

The PLLSAI is also used to generate the LCD-TFT clock.

3.30 Secure digital input/output interface (SDIO)

An SD/SDIO/MMC host interface is available, that supports MultiMediaCard System Specification Version 4.2 in three different databus modes: 1-bit (default), 4-bit and 8-bit.

The interface allows data transfer at up to 48 MHz, and is compliant with the SD Memory Card Specification Version 2.0.

The SDIO Card Specification Version 2.0 is also supported with two different databus modes: 1-bit (default) and 4-bit.

The current version supports only one SD/SDIO/MMC4.2 card at any one time and a stack of MMC4.1 or previous.

In addition to SD/SDIO/MMC, this interface is fully compliant with the CE-ATA digital protocol Rev1.1.

3.31 Ethernet MAC interface with dedicated DMA and IEEE 1588 support

The devices provide an IEEE-802.3-2002-compliant media access controller (MAC) for ethernet LAN communications through an industry-standard medium-independent interface (MII) or a reduced medium-independent interface (RMII). The microcontroller requires an external physical interface device (PHY) to connect to the physical LAN bus (twisted-pair, fiber, etc.). The PHY is connected to the device MII port using 17 signals for MII or 9 signals for RMII, and can be clocked using the 25 MHz (MII) from the microcontroller.

The devices include the following features:

- Supports 10 and 100 Mbit/s rates
- Dedicated DMA controller allowing high-speed transfers between the dedicated SRAM and the descriptors (see the STM32F4xx reference manual for details)
- Tagged MAC frame support (VLAN support)
- Half-duplex (CSMA/CD) and full-duplex operation
- MAC control sublayer (control frames) support
- 32-bit CRC generation and removal
- Several address filtering modes for physical and multicast address (multicast and group addresses)
- 32-bit status code for each transmitted or received frame
- Internal FIFOs to buffer transmit and receive frames. The transmit FIFO and the receive FIFO are both 2 Kbytes.
- Supports hardware PTP (precision time protocol) in accordance with IEEE 1588 2008 (PTP V2) with the time stamp comparator connected to the TIM2 input
- Triggers interrupt when system time becomes greater than target time

3.32 Controller area network (bxCAN)

The two CANs are compliant with the 2.0A and B (active) specifications with a bitrate up to 1 Mbit/s. They can receive and transmit standard frames with 11-bit identifiers as well as extended frames with 29-bit identifiers. Each CAN has three transmit mailboxes, two receive

FIFOS with 3 stages and 28 shared scalable filter banks (all of them can be used even if one CAN is used). 256 bytes of SRAM are allocated for each CAN.

3.33 Universal serial bus on-the-go full-speed (OTG_FS)

The devices embed an USB OTG full-speed device/host/OTG peripheral with integrated transceivers. The USB OTG FS peripheral is compliant with the USB 2.0 specification and with the OTG 1.0 specification. It has software-configurable endpoint setting and supports suspend/resume. The USB OTG full-speed controller requires a dedicated 48 MHz clock that is generated by a PLL connected to the HSE oscillator. The major features are:

- Combined Rx and Tx FIFO size of 320 × 35 bits with dynamic FIFO sizing
- Supports the session request protocol (SRP) and host negotiation protocol (HNP)
- 4 bidirectional endpoints
- 8 host channels with periodic OUT support
- HNP/SNP/IP inside (no need for any external resistor)
- For OTG/Host modes, a power switch is needed in case bus-powered devices are connected

3.34 Universal serial bus on-the-go high-speed (OTG_HS)

The devices embed a USB OTG high-speed (up to 480 Mb/s) device/host/OTG peripheral. The USB OTG HS supports both full-speed and high-speed operations. It integrates the transceivers for full-speed operation (12 MB/s) and features a UTMI low-pin interface (ULPI) for high-speed operation (480 MB/s). When using the USB OTG HS in HS mode, an external PHY device connected to the ULPI is required.

The USB OTG HS peripheral is compliant with the USB 2.0 specification and with the OTG 1.0 specification. It has software-configurable endpoint setting and supports suspend/resume. The USB OTG full-speed controller requires a dedicated 48 MHz clock that is generated by a PLL connected to the HSE oscillator.

The major features are:

- Combined Rx and Tx FIFO size of 1 Kbit × 35 with dynamic FIFO sizing
- Supports the session request protocol (SRP) and host negotiation protocol (HNP)
- 6 bidirectional endpoints
- 12 host channels with periodic OUT support
- Internal FS OTG PHY support
- External HS or HS OTG operation supporting ULPI in SDR mode. The OTG PHY is connected to the microcontroller ULPI port through 12 signals. It can be clocked using the 60 MHz output.
- Internal USB DMA
- HNP/SNP/IP inside (no need for any external resistor)
- for OTG/Host modes, a power switch is needed in case bus-powered devices are connected

3.35 Digital camera interface (DCMI)

The devices embed a camera interface that can connect with camera modules and CMOS sensors through an 8-bit to 14-bit parallel interface, to receive video data. The camera interface can sustain a data transfer rate up to 54 Mbyte/s at 54 MHz. It features:

- Programmable polarity for the input pixel clock and synchronization signals
- Parallel data communication can be 8-, 10-, 12- or 14-bit
- Supports 8-bit progressive video monochrome or raw bayer format, YCbCr 4:2:2 progressive video, RGB 565 progressive video or compressed data (like JPEG)
- Supports continuous mode or snapshot (a single frame) mode
- Capability to automatically crop the image

3.36 Random number generator (RNG)

All devices embed an RNG that delivers 32-bit random numbers generated by an integrated analog circuit.

3.37 General-purpose input/outputs (GPIOs)

Each of the GPIO pins can be configured by software as output (push-pull or open-drain, with or without pull-up or pull-down), as input (floating, with or without pull-up or pull-down) or as peripheral alternate function. Most of the GPIO pins are shared with digital or analog alternate functions. All GPIOs are high-current-capable and have speed selection to better manage internal noise, power consumption and electromagnetic emission.

The I/O configuration can be locked if needed by following a specific sequence in order to avoid spurious writing to the I/Os registers.

Fast I/O handling allowing maximum I/O toggling up to 90 MHz.

3.38 Analog-to-digital converters (ADCs)

Three 12-bit analog-to-digital converters are embedded and each ADC shares up to 16 external channels, performing conversions in the single-shot or scan mode. In scan mode, automatic conversion is performed on a selected group of analog inputs.

Additional logic functions embedded in the ADC interface allow:

- Simultaneous sample and hold
- Interleaved sample and hold

The ADC can be served by the DMA controller. An analog watchdog feature allows very precise monitoring of the converted voltage of one, some or all selected channels. An interrupt is generated when the converted voltage is outside the programmed thresholds.

To synchronize A/D conversion and timers, the ADCs could be triggered by any of TIM1, TIM2, TIM3, TIM4, TIM5, or TIM8 timer.

3.39 Temperature sensor

The temperature sensor has to generate a voltage that varies linearly with temperature. The conversion range is between 1.7 V and 3.6 V. The temperature sensor is internally connected to the same input channel as V_{BAT}, ADC1_IN18, which is used to convert the sensor output voltage into a digital value. When the temperature sensor and V_{BAT} conversion are enabled at the same time, only V_{BAT} conversion is performed.

As the offset of the temperature sensor varies from chip to chip due to process variation, the internal temperature sensor is mainly suitable for applications that detect temperature changes instead of absolute temperatures. If an accurate temperature reading is needed, then an external temperature sensor part should be used.

3.40 Digital-to-analog converter (DAC)

The two 12-bit buffered DAC channels can be used to convert two digital signals into two analog voltage signal outputs.

This dual digital Interface supports the following features:

- two DAC converters: one for each output channel
- 8-bit or 10-bit monotonic output
- left or right data alignment in 12-bit mode
- synchronized update capability
- noise-wave generation
- triangular-wave generation
- dual DAC channel independent or simultaneous conversions
- DMA capability for each channel
- external triggers for conversion
- input voltage reference V_{REF+}

Eight DAC trigger inputs are used in the device. The DAC channels are triggered through the timer update outputs that are also connected to different DMA streams.

3.41 Serial wire JTAG debug port (SWJ-DP)

The Arm SWJ-DP interface is embedded, and is a combined JTAG and serial wire debug port that enables either a serial wire debug or a JTAG probe to be connected to the target.

Debug is performed using 2 pins only instead of 5 required by the JTAG (JTAG pins could be re-use as GPIO with alternate function): the JTAG TMS and TCK pins are shared with SWDIO and SWCLK, respectively, and a specific sequence on the TMS pin is used to switch between JTAG-DP and SW-DP.

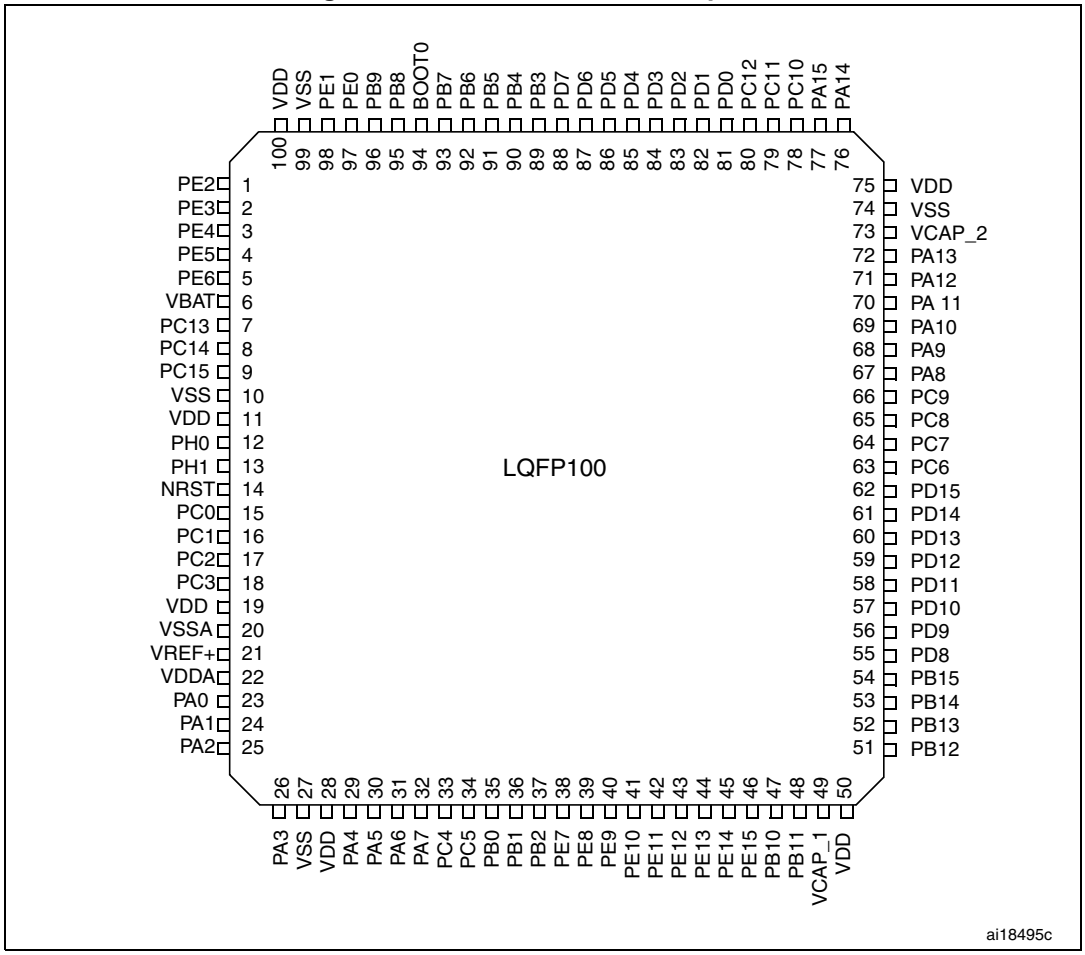
3.42 Embedded Trace Macrocell™

The Arm Embedded Trace Macrocell provides a greater visibility of the instruction and data flow inside the CPU core by streaming compressed data at a very high rate from the STM32F42x through a small number of ETM pins to an external hardware trace port analyzer (TPA) device. The TPA is connected to a host computer using USB, Ethernet, or any other high-speed channel. Real-time instruction and data flow activity can be recorded and then formatted for display on the host computer that runs the debugger software. TPA hardware is commercially available from common development tool vendors.

The Embedded Trace Macrocell operates with third party debugger software tools.

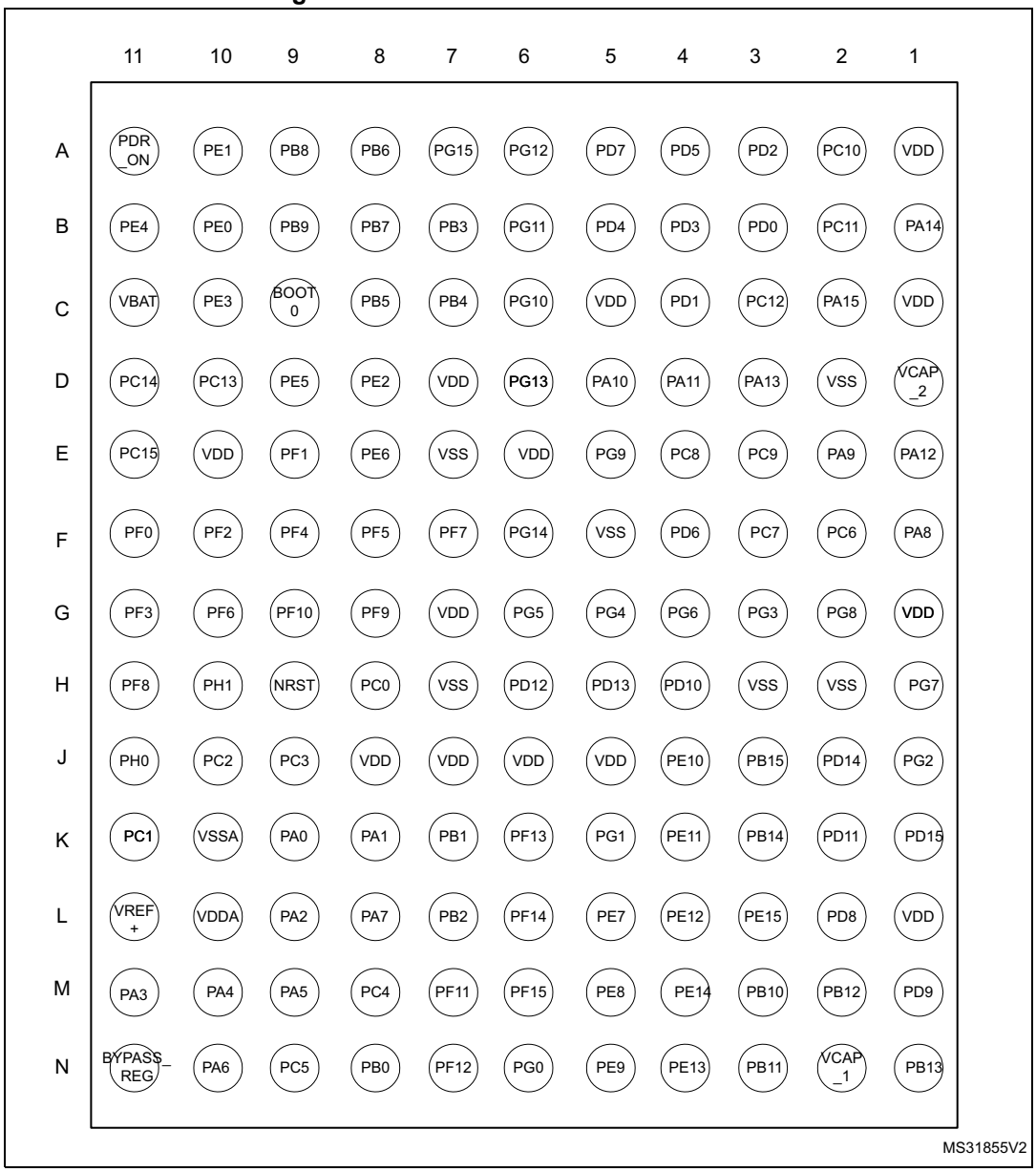
4 Pinouts and pin description

Figure 11. STM32F42x LQFP100 pinout



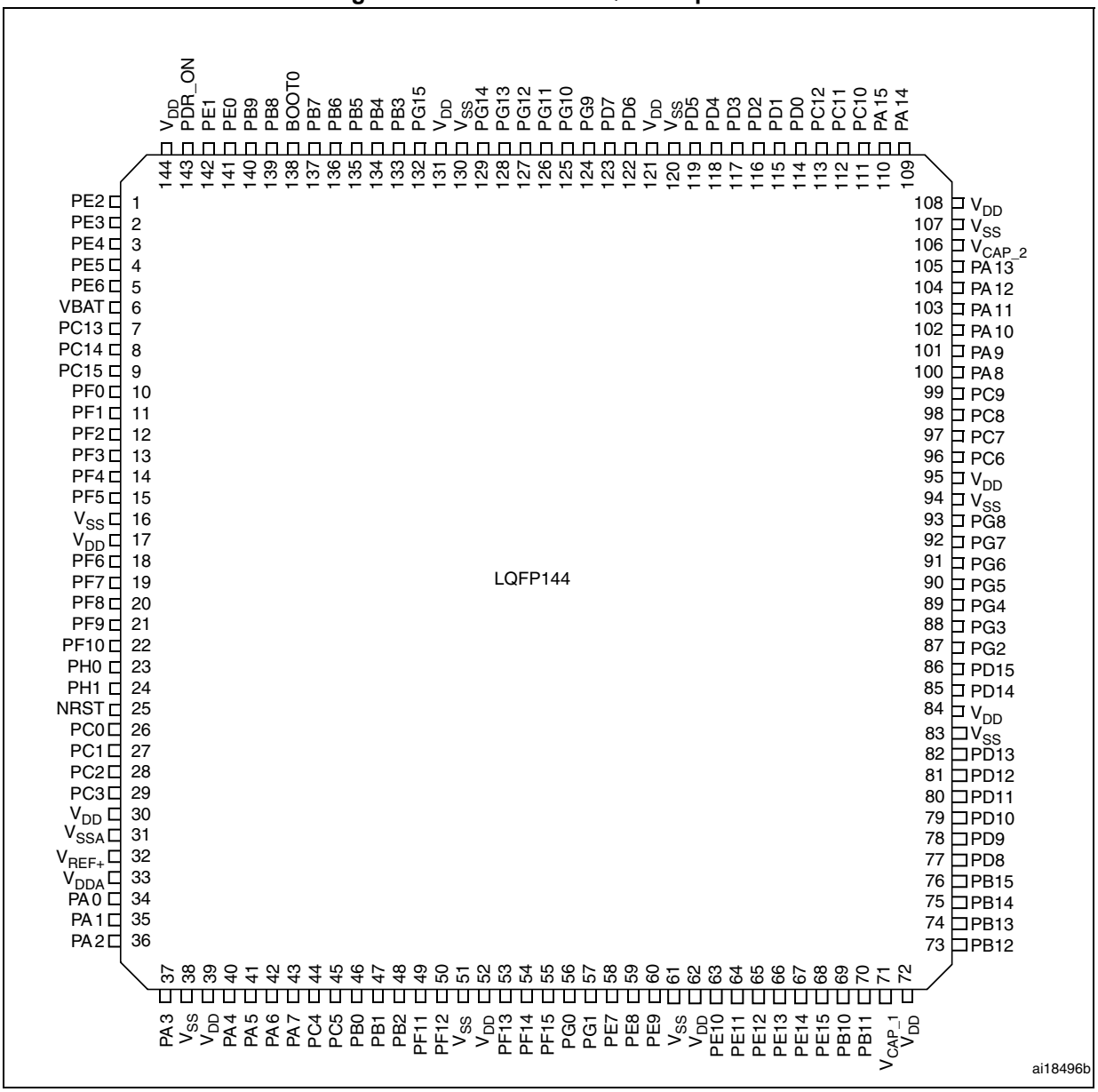
1. The above figure shows the package top view.

Figure 12. STM32F42x WLCSP143 ballout



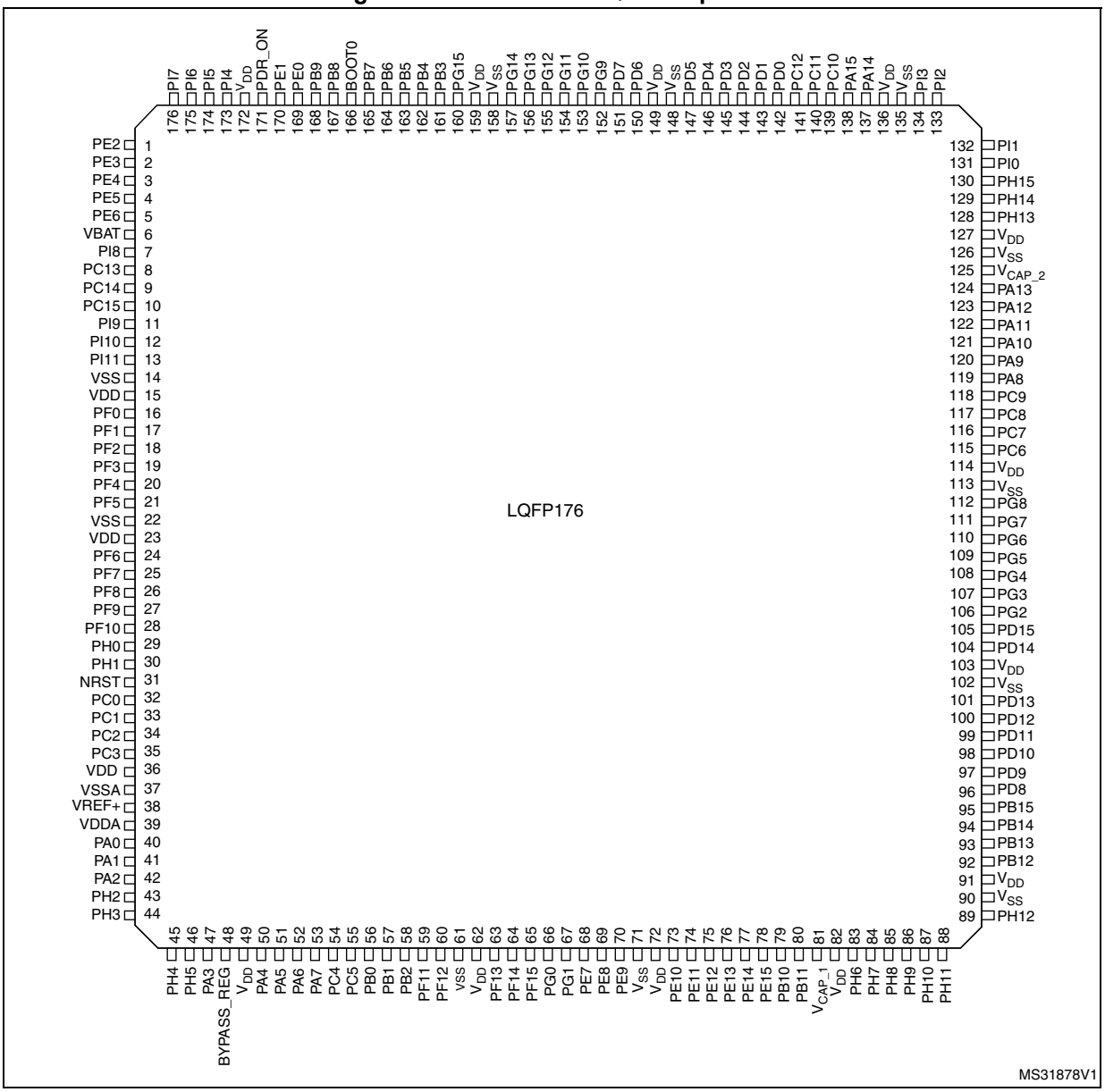
1. The above figure shows the package bump view.

Figure 13. STM32F42x LQFP144 pinout



1. The above figure shows the package top view.

Figure 14. STM32F42x LQFP176 pinout



1. The above figure shows the package top view.

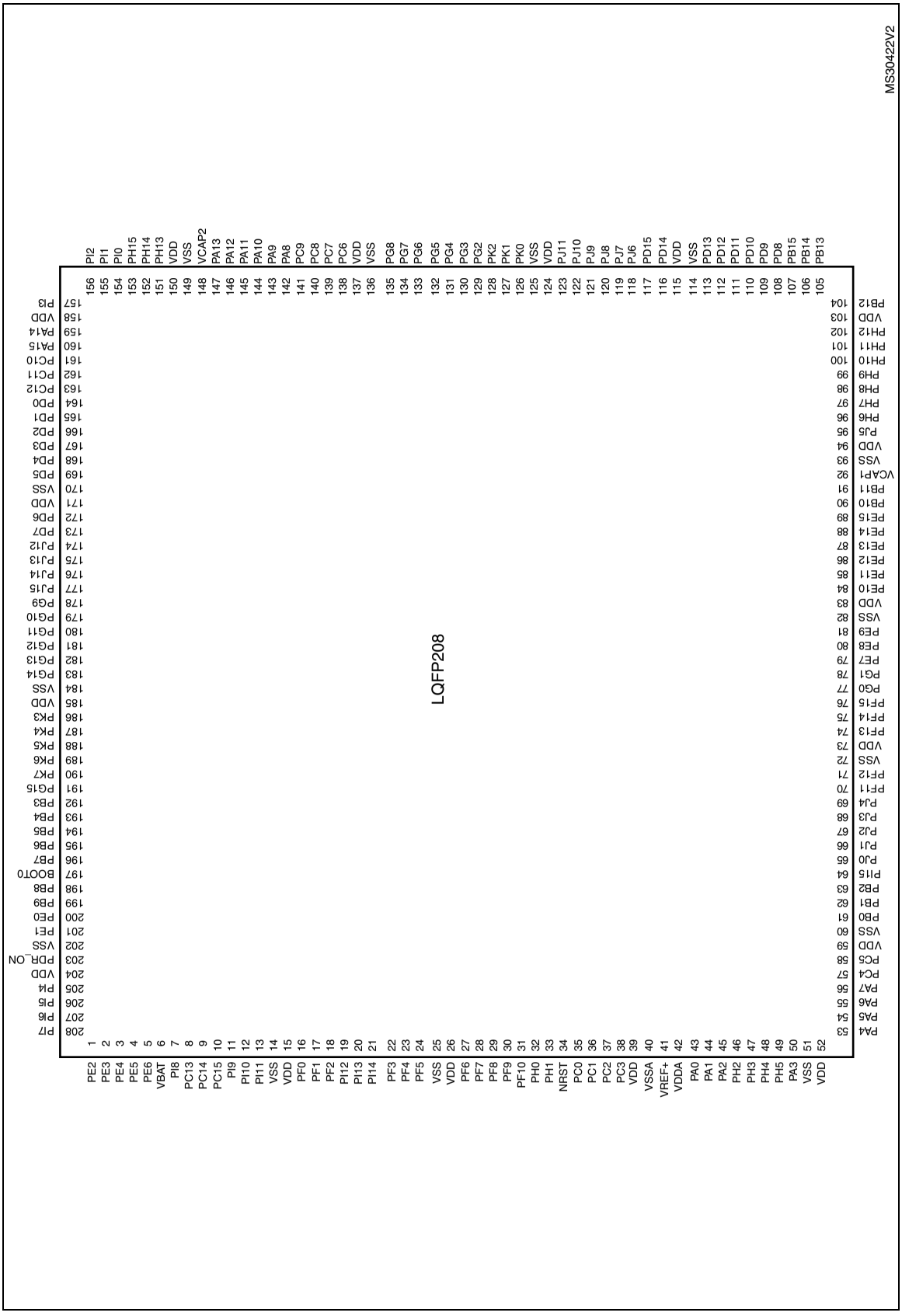
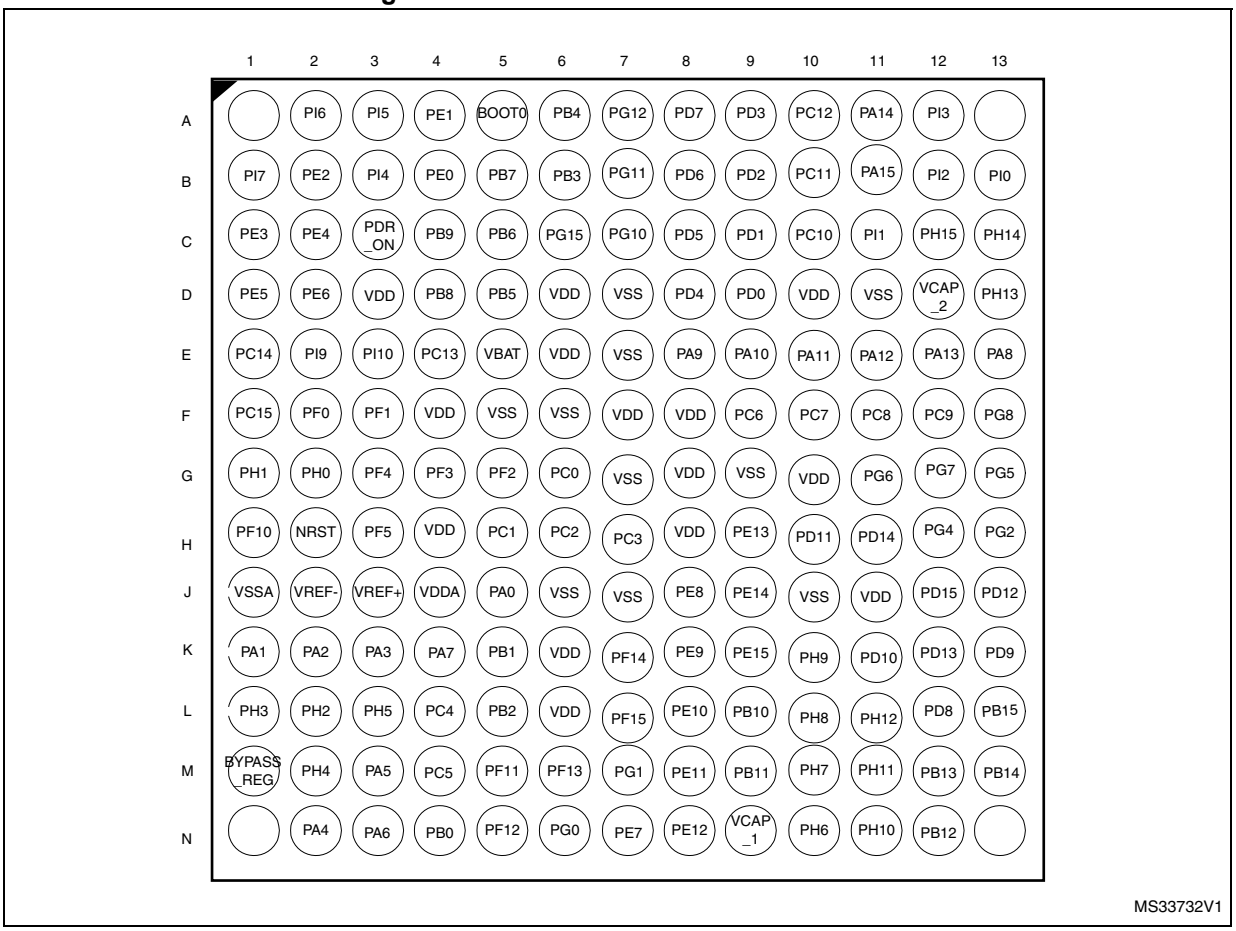


Figure 15. STM32F42x LQFP208 pinout

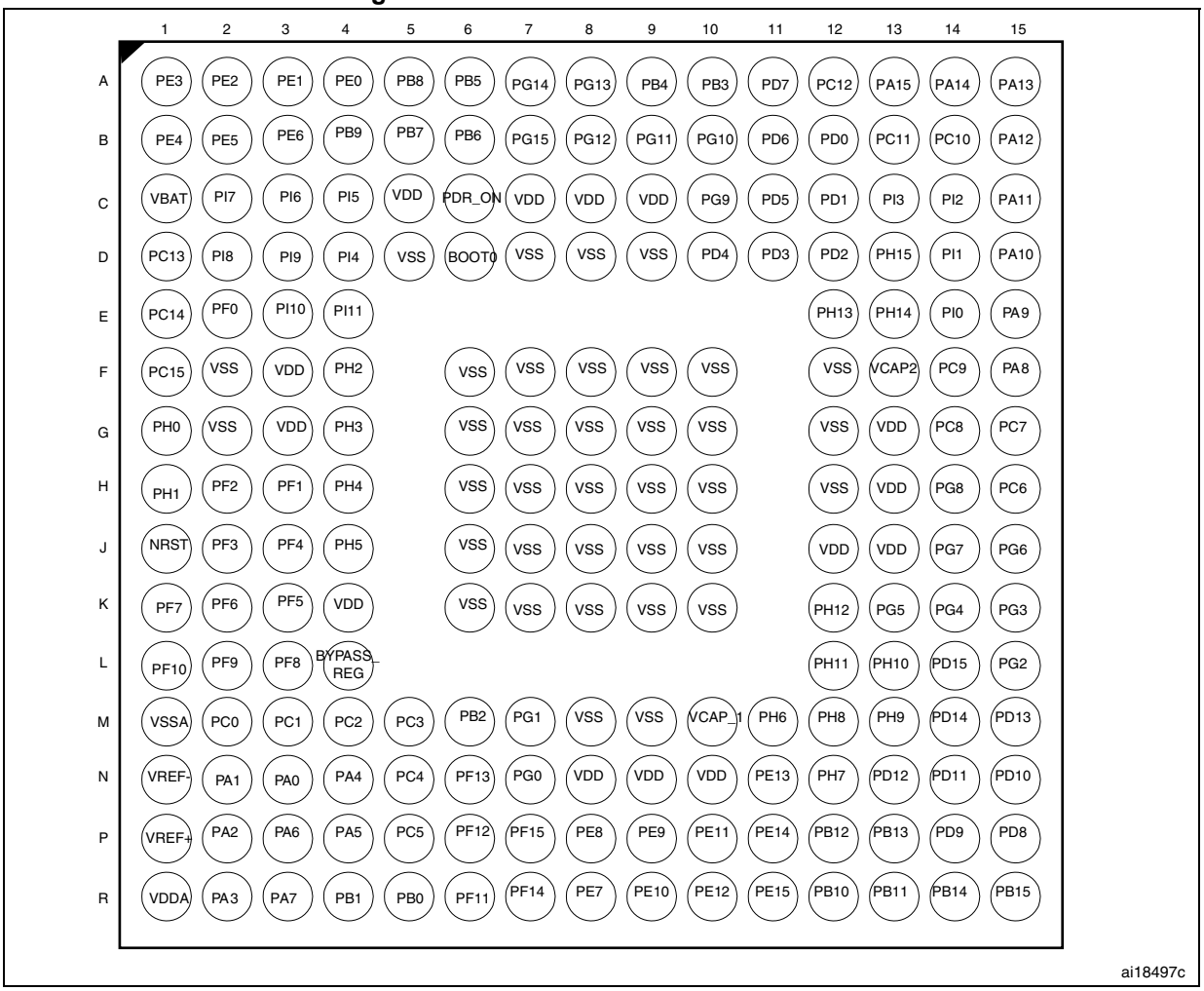
1. The above figure shows the package top view.

Figure 16. STM32F42x UFBGA169 ballout



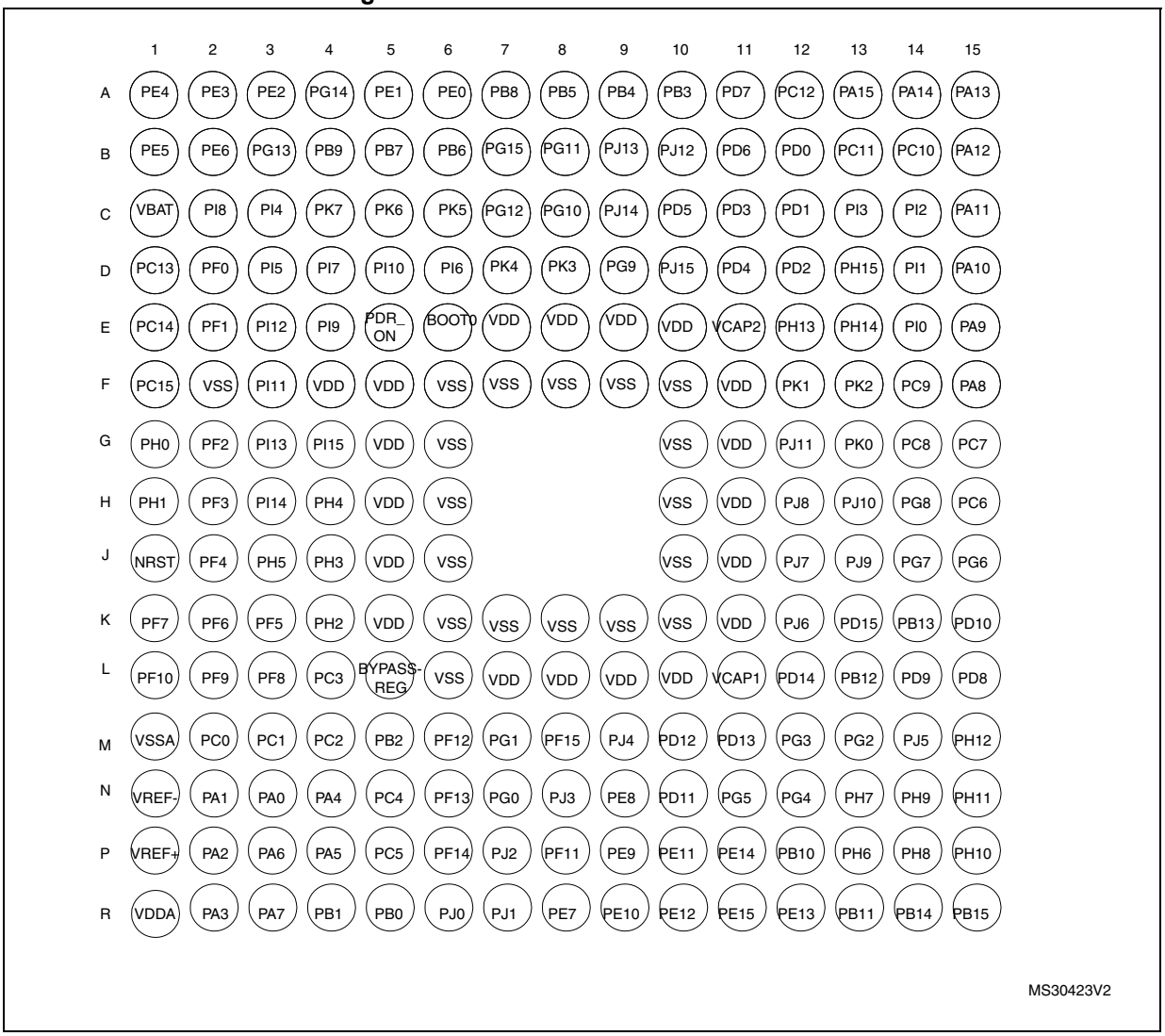
1. The above figure shows the package top view.
2. The 4 corners balls, A1,A13, N1 and N13, are not bonded internally and should be left not connected on the PCB.

Figure 17. STM32F42x UFBGA176 ballout



1. The above figure shows the package top view.

Figure 18. STM32F42x TFBGA216 ballout



1. The above figure shows the package top view.

Table 9. Legend/abbreviations used in the pinout table

Name	Abbreviation	Definition
Pin name	Unless otherwise specified in brackets below the pin name, the pin function during and after reset is the same as the actual pin name	
Pin type	S	Supply pin
	I	Input only pin
	I/O	Input / output pin
I/O structure	FT	5 V tolerant I/O
	TTa	3.3 V tolerant I/O directly connected to ADC
	B	Dedicated BOOT0 pin
	RST	Bidirectional reset pin with weak pull-up resistor
Notes	Unless otherwise specified by a note, all I/Os are set as floating inputs during and after reset	
Alternate functions	Functions selected through GPIOx_AFR registers	
Additional functions	Functions directly selected/enabled through peripheral registers	

Table 10. STM32F427xx and STM32F429xx pin and ball definitions

Pin number								Pin name (function after reset)(1)	Pin type	I / O structure	Notes	Alternate functions	Additional functions
LQFP100	LQFP144	UFBGA169	UFBGA176	LQFP176	WLCSP143	LQFP208	TFBGA216						
1	1	B2	A2	1	D8	1	A3	PE2	I/O	FT	-	TRACECLK, SPI4_SCK, SAI1_MCLK_A, ETH_MII_TXD3, FMC_A23, EVENTOUT	-
2	2	C1	A1	2	C10	2	A2	PE3	I/O	FT	-	TRACED0, SAI1_SD_B, FMC_A19, EVENTOUT	-
3	3	C2	B1	3	B11	3	A1	PE4	I/O	FT	-	TRACED1, SPI4_NSS, SAI1_FS_A, FMC_A20, DCMI_D4, LCD_B0, EVENTOUT	-

Table 10. STM32F427xx and STM32F429xx pin and ball definitions (continued)

Pin number								Pin name (function after reset)(1)	Pin type	I / O structure	Notes	Alternate functions	Additional functions
LQFP100	LQFP144	UFBGA169	UFBGA176	LQFP176	WLCSP143	LQFP208	TFBGA216						
4	4	D1	B2	4	D9	4	B1	PE5	I/O	FT	-	TRACED2, TIM9_CH1, SPI4_MISO, SAI1_SCK_A, FMC_A21, DCMI_D6, LCD_G0, EVENTOUT	-
5	5	D2	B3	5	E8	5	B2	PE6	I/O	FT	-	TRACED3, TIM9_CH2, SPI4_MOSI, SAI1_SD_A, FMC_A22, DCMI_D7, LCD_G1, EVENTOUT	-
-	-	-	-	-	-	-	G6	V_{SS}	S	-	-	-	-
-	-	-	-	-	-	-	F5	V_{DD}	S	-	-	-	-
6	6	E5	C1	6	C11	6	C1	V_{BAT}	S	-	-	-	-
-	-	NC (2)	D2	7	-	7	C2	PI8	I/O	FT	(3) (4)	EVENTOUT	TAMP_2
7	7	E4	D1	8	D10	8	D1	PC13	I/O	FT	(3) (4)	EVENTOUT	TAMP_1
8	8	E1	E1	9	D11	9	E1	PC14-OSC32_IN (PC14)	I/O	FT	(3) (4)	EVENTOUT	OSC32_IN (5)
9	9	F1	F1	10	E11	10	F1	PC15-OSC32_OUT (PC15)	I/O	FT	(3) (4)	EVENTOUT	OSC32_OUT(5)
-	-	-	-	-	-	-	G5	V_{DD}	S	-	-	-	-
-	-	E2	D3	11	-	11	E4	PI9	I/O	FT	-	CAN1_RX, FMC_D30, LCD_VSYNC, EVENTOUT	-
-	-	E3	E3	12	-	12	D5	PI10	I/O	FT	-	ETH_MII_RX_ER, FMC_D31, LCD_HSYNC, EVENTOUT	-
-	-	NC (2)	E4	13	-	13	F3	PI11	I/O	FT	-	OTG_HS_ULPI_DIR, EVENTOUT	-
-	-	F6	F2	14	E7	14	F2	V_{SS}	S	-	-	-	-
-	-	F4	F3	15	E10	15	F4	V_{DD}	S	-	-	-	-

Table 10. STM32F427xx and STM32F429xx pin and ball definitions (continued)

Pin number								Pin name (function after reset)(1)	Pin type	I / O structure	Notes	Alternate functions	Additional functions
LQFP100	LQFP144	UFBGA169	UFBGA176	LQFP176	WLCSP143	LQFP208	TFBGA216						
-	10	F2	E2	16	F11	16	D2	PF0	I/O	FT	-	I2C2_SDA, FMC_A0, EVENTOUT	-
-	11	F3	H3	17	E9	17	E2	PF1	I/O	FT	-	I2C2_SCL, FMC_A1, EVENTOUT	-
-	12	G5	H2	18	F10	18	G2	PF2	I/O	FT	-	I2C2_SMBA, FMC_A2, EVENTOUT	-
-	-	-	-	-	-	19	E3	PI12	I/O	FT	-	LCD_HSYNC, EVENTOUT	-
-	-	-	-	-	-	20	G3	PI13	I/O	FT	-	LCD_VSYNC, EVENTOUT	-
-	-	-	-	-	-	21	H3	PI14	I/O	FT		LCD_CLK, EVENTOUT	-
-	13	G4	J2	19	G11	22	H2	PF3	I/O	FT	(5)	FMC_A3, EVENTOUT	ADC3_IN9
-	14	G3	J3	20	F9	23	J2	PF4	I/O	FT	(5)	FMC_A4, EVENTOUT	ADC3_ IN14
-	15	H3	K3	21	F8	24	K3	PF5	I/O	FT	(5)	FMC_A5, EVENTOUT	ADC3_ IN15
10	16	G7	G2	22	H7	25	H6	V_{SS}	S	-	-	-	-
11	17	G8	G3	23	-	26	H5	V_{DD}	S	-	-	-	-
-	18	NC (2)	K2	24	G10	27	K2	PF6	I/O	FT	(5)	TIM10_CH1, SPI5_NSS, SAI1_SD_B, UART7_Rx, FMC_NIORD, EVENTOUT	ADC3_IN4
-	19	NC (2)	K1	25	F7	28	K1	PF7	I/O	FT	(5)	TIM11_CH1, SPI5_SCK, SAI1_MCLK_B, UART7_Tx, FMC_NREG, EVENTOUT	ADC3_IN5
-	20	NC (2)	L3	26	H11	29	L3	PF8	I/O	FT	(5)	SPI5_MISO, SAI1_SCK_B, TIM13_CH1, FMC_NIOWR, EVENTOUT	ADC3_IN6

Table 10. STM32F427xx and STM32F429xx pin and ball definitions (continued)

Pin number								Pin name (function after reset)(1)	Pin type	I / O structure	Notes	Alternate functions	Additional functions
LQFP100	LQFP144	UFBGA169	UFBGA176	LQFP176	WLCSP143	LQFP208	TFBGA216						
-	21	NC (2)	L2	27	G8	30	L2	PF9	I/O	FT	(5)	SPI5_MOSI, SAI1_FS_B, TIM14_CH1, FMC_CD, EVENTOUT	ADC3_IN7
-	22	H1	L1	28	G9	31	L1	PF10	I/O	FT	(5)	FMC_INTR, DCMI_D11, LCD_DE, EVENTOUT	ADC3_IN8
12	23	G2	G1	29	J11	32	G1	PH0-OSC_IN (PH0)	I/O	FT	-	EVENTOUT	OSC_IN(5)
13	24	G1	H1	30	H10	33	H1	PH1-OSC_OUT (PH1)	I/O	FT	-	EVENTOUT	OSC_OUT (5)
14	25	H2	J1	31	H9	34	J1	NRST	I/O	RST	-	-	-
15	26	G6	M2	32	H8	35	M2	PC0	I/O	FT	(5)	OTG_HS_ULPI_STP, FMC_SDNWE, EVENTOUT	ADC123_IN10
16	27	H5	M3	33	K11	36	M3	PC1	I/O	FT	(5)	ETH_MDC, EVENTOUT	ADC123_IN11
17	28	H6	M4	34	J10	37	M4	PC2	I/O	FT	(5)	SPI2_MISO, I2S2ext_SD, OTG_HS_ULPI_DIR, ETH_MII_TXD2, FMC_SDNE0, EVENTOUT	ADC123_IN12
18	29	H7	M5	35	J9	38	L4	PC3	I/O	FT	(5)	SPI2_MOSI/I2S2_SD, OTG_HS_ULPI_NXT, ETH_MII_TX_CLK, FMC_SDCKE0, EVENTOUT	ADC123_IN13
19	30	-	-	36	G7	39	J5	V_{DD}	S	-	-	-	-
-	-	-	-	-	-	-	J6	V_{SS}	S	-	-	-	-
20	31	J1	M1	37	K10	40	M1	V_{SSA}	S	-	-	-	-
-	-	J2	N1	-	-	-	N1	V_{REF-}	S	-	-	-	-
21	32	J3	P1	38	L11	41	P1	V_{REF+}	S	-	-	-	-

Table 10. STM32F427xx and STM32F429xx pin and ball definitions (continued)

Pin number								Pin name (function after reset)(1)	Pin type	I / O structure	Notes	Alternate functions	Additional functions
LQFP100	LQFP144	UFBGA169	UFBGA176	LQFP176	WLCSP143	LQFP208	TFBGA216						
22	33	J4	R1	39	L10	42	R1	V_{DDA}	S	-	-	-	-
23	34	J5	N3	40	K9	43	N3	PA0-WKUP (PA0)	I/O	FT	(6)	TIM2_CH1/TIM2_ETR, TIM5_CH1, TIM8_ETR, USART2_CTS, UART4_TX, ETH_MII_CRS, EVENTOUT	ADC123_IN0/WKUP (5)
24	35	K1	N2	41	K8	44	N2	PA1	I/O	FT	(5)	TIM2_CH2, TIM5_CH2, USART2_RTS, UART4_RX, ETH_MII_RX_CLK/ETH_RMII_REF_CLK, EVENTOUT	ADC123_IN1
25	36	K2	P2	42	L9	45	P2	PA2	I/O	FT	(5)	TIM2_CH3, TIM5_CH3, TIM9_CH1, USART2_TX, ETH_MDIO, EVENTOUT	ADC123_IN2
-	-	L2	F4	43	-	46	K4	PH2	I/O	FT	-	ETH_MII_CRS, FMC_SDCKE0, LCD_R0, EVENTOUT	-
-	-	L1	G4	44	-	47	J4	PH3	I/O	FT	-	ETH_MII_COL, FMC_SDNE0, LCD_R1, EVENTOUT	-
-	-	M2	H4	45	-	48	H4	PH4	I/O	FT	-	I2C2_SCL, OTG_HS_ULPI_NXT, EVENTOUT	-
-	-	L3	J4	46	-	49	J3	PH5	I/O	FT	-	I2C2_SDA, SPI5_NSS, FMC_SDNWE, EVENTOUT	-
26	37	K3	R2	47	M11	50	R2	PA3	I/O	FT	(5)	TIM2_CH4, TIM5_CH4, TIM9_CH2, USART2_RX, OTG_HS_ULPI_D0, ETH_MII_COL, LCD_B5, EVENTOUT	ADC123_IN3
27	38	-	-		-	51	K6	V_{SS}	S	-	-	-	-

Table 10. STM32F427xx and STM32F429xx pin and ball definitions (continued)

Pin number								Pin name (function after reset)[1]	Pin type	I / O structure	Notes	Alternate functions	Additional functions
LQFP100	LQFP144	UFBGA169	UFBGA176	LQFP176	WLCSP143	LQFP208	TFBGA216						
-	-	M1	L4	48	N11	-	L5	BYPASS_ REG	I	FT	-	-	-
28	39	J11	K4	49	J8	52	K5	V_{DD}	S	-	-	-	-
29	40	N2	N4	50	M10	53	N4	PA4	I/O	TTa	(5)	SPI1_NSS, SPI3_NSS/I2S3_WS, USART2_CK, OTG_HS_SOF, DCMI_HSYNC, LCD_VSYNC, EVENTOUT	ADC12_ IN4 /DAC_ OUT1
30	41	M3	P4	51	M9	54	P4	PA5	I/O	TTa	(5)	TIM2_CH1/TIM2_ETR, TIM8_CH1N, SPI1_SCK, OTG_HS_ULPI_CK, EVENTOUT	ADC12_ IN5/DAC_ OUT2
31	42	N3	P3	52	N10	55	P3	PA6	I/O	FT	(5)	TIM1_BKIN, TIM3_CH1, TIM8_BKIN, SPI1_MISO, TIM13_CH1, DCMI_PIXCLK, LCD_G2, EVENTOUT	ADC12_ IN6
32	43	K4	R3	53	L8	56	R3	PA7	I/O	FT	(5)	TIM1_CH1N, TIM3_CH2, TIM8_CH1N, SPI1_MOSI, TIM14_CH1, ETH_MII_RX_DV/ETH_ RMII_CRS_DV, EVENTOUT	ADC12_ IN7
33	44	L4	N5	54	M8	57	N5	PC4	I/O	FT	(5)	ETH_MII_RXD0/ETH_ RMII_RXD0, EVENTOUT	ADC12_ IN14
34	45	M4	P5	55	N9	58	P5	PC5	I/O	FT	(5)	ETH_MII_RXD1/ETH_ RMII_RXD1, EVENTOUT	ADC12_ IN15
-	-	-	-	-	J7	59	L7	V_{DD}	S	-	-	-	-
-	-	-	-	-	-	60	L6	VSS	S	-	-	-	-

Table 10. STM32F427xx and STM32F429xx pin and ball definitions (continued)

Pin number								Pin name (function after reset)(1)	Pin type	I / O structure	Notes	Alternate functions	Additional functions
LQFP100	LQFP144	UFBGA169	UFBGA176	LQFP176	WLCSP143	LQFP208	TFBGA216						
35	46	N4	R5	56	N8	61	R5	PB0	I/O	FT	(5)	TIM1_CH2N, TIM3_CH3, TIM8_CH2N, LCD_R3, OTG_HS_ULPI_D1, ETH_MII_RXD2, EVENTOUT	ADC12_IN8
36	47	K5	R4	57	K7	62	R4	PB1	I/O	FT	(5)	TIM1_CH3N, TIM3_CH4, TIM8_CH3N, LCD_R6, OTG_HS_ULPI_D2, ETH_MII_RXD3, EVENTOUT	ADC12_IN9
37	48	L5	M6	58	L7	63	M5	PB2-BOOT1 (PB2)	I/O	FT	-	EVENTOUT	-
-	-	-	-	-	-	64	G4	PI15	I/O	FT	-	LCD_R0, EVENTOUT	-
-	-	-	-	-	-	65	R6	PJ0	I/O	FT	-	LCD_R1, EVENTOUT	-
-	-	-	-	-	-	66	R7	PJ1	I/O	FT	-	LCD_R2, EVENTOUT	-
-	-	-	-	-	-	67	P7	PJ2	I/O	FT	-	LCD_R3, EVENTOUT	-
-	-	-	-	-	-	68	N8	PJ3	I/O	FT	-	LCD_R4, EVENTOUT	-
-	-	-	-	-	-	69	M9	PJ4	I/O	FT	-	LCD_R5, EVENTOUT	-
-	49	M5	R6	59	M7	70	P8	PF11	I/O	FT	-	SPI5_MOSI, FMC_SDNRAS, DCMI_D12, EVENTOUT	-
-	50	N5	P6	60	N7	71	M6	PF12	I/O	FT	-	FMC_A6, EVENTOUT	-
-	51	G9	M8	61	-	72	K7	V_{SS}	S		-	-	-
-	52	D10	N8	62	-	73	L8	V_{DD}	S		-	-	-
-	53	M6	N6	63	K6	74	N6	PF13	I/O	FT	-	FMC_A7, EVENTOUT	-
-	54	K7	R7	64	L6	75	P6	PF14	I/O	FT	-	FMC_A8, EVENTOUT	-
-	55	L7	P7	65	M6	76	M8	PF15	I/O	FT	-	FMC_A9, EVENTOUT	-
-	56	N6	N7	66	N6	77	N7	PG0	I/O	FT	-	FMC_A10, EVENTOUT	-
-	57	M7	M7	67	K5	78	M7	PG1	I/O	FT	-	FMC_A11, EVENTOUT	-

Table 10. STM32F427xx and STM32F429xx pin and ball definitions (continued)

Pin number								Pin name (function after reset)(1)	Pin type	I / O structure	Notes	Alternate functions	Additional functions
LQFP100	LQFP144	UFBGA169	UFBGA176	LQFP176	WLCSP143	LQFP208	TFBGA216						
38	58	N7	R8	68	L5	79	R8	PE7	I/O	FT	-	TIM1_ETR, UART7_Rx, FMC_D4, EVENTOUT	-
39	59	J8	P8	69	M5	80	N9	PE8	I/O	FT	-	TIM1_CH1N, UART7_Tx, FMC_D5, EVENTOUT	-
40	60	K8	P9	70	N5	81	P9	PE9	I/O	FT	-	TIM1_CH1, FMC_D6, EVENTOUT	-
-	61	J6	M9	71	H3	82	K8	V_{SS}	S		-		-
-	62	G10	N9	72	J5	83	L9	V_{DD}	S		-		-
41	63	L8	R9	73	J4	84	R9	PE10	I/O	FT	-	TIM1_CH2N, FMC_D7, EVENTOUT	-
42	64	M8	P10	74	K4	85	P10	PE11	I/O	FT	-	TIM1_CH2, SPI4_NSS, FMC_D8, LCD_G3, EVENTOUT	-
43	65	N8	R10	75	L4	86	R10	PE12	I/O	FT	-	TIM1_CH3N, SPI4_SCK, FMC_D9, LCD_B4, EVENTOUT	-
44	66	H9	N11	76	N4	87	R12	PE13	I/O	FT	-	TIM1_CH3, SPI4_MISO, FMC_D10, LCD_DE, EVENTOUT	-
45	67	J9	P11	77	M4	88	P11	PE14	I/O	FT	-	TIM1_CH4, SPI4_MOSI, FMC_D11, LCD_CLK, EVENTOUT	-
46	68	K9	R11	78	L3	89	R11	PE15	I/O	FT	-	TIM1_BKIN, FMC_D12, LCD_R7, EVENTOUT	-
47	69	L9	R12	79	M3	90	P12	PB10	I/O	FT	-	TIM2_CH3, I2C2_SCL, SPI2_SCK/I2S2_CK, USART3_TX, OTG_HS_ULPI_D3, ETH_MII_RX_ER, LCD_G4, EVENTOUT	-
48	70	M9	R13	80	N3	91	R13	PB11	I/O	FT	-	TIM2_CH4, I2C2_SDA, USART3_RX, OTG_HS_ULPI_D4, ETH_MII_TX_EN/ETH_RMII_TX_EN, LCD_G5, EVENTOUT	-

Table 10. STM32F427xx and STM32F429xx pin and ball definitions (continued)

Pin number								Pin name (function after reset)(1)	Pin type	I / O structure	Notes	Alternate functions	Additional functions
LQFP100	LQFP144	UFBGA169	UFBGA176	LQFP176	WLCSP143	LQFP208	TFBGA216						
49	71	N9	M10	81	N2	92	L11	V_{CAP_1}	S	-	-	-	-
-	-	-	-	-	H2	93	K9	V_{SS}	S	-	-	-	-
50	72	F8	N10	82	J6	94	L10	V_{DD}	S	-	-	-	-
-	-	-	-	-	-	95	M14	PJ5	I/O	-	-	LCD_R6, EVENTOUT	-
-	-	N10	M11	83	-	96	P13	PH6	I/O	FT	-	I2C2_SMBA, SPI5_SCK, TIM12_CH1, ETH_MII_RXD2, FMC_SDNE1, DCMI_D8, EVENTOUT	-
-	-	M10	N12	84	-	97	N13	PH7	I/O	FT	-	I2C3_SCL, SPI5_MISO, ETH_MII_RXD3, FMC_SDCKE1, DCMI_D9, EVENTOUT	-
-	-	L10	M12	85	-	98	P14	PH8	I/O	FT	-	I2C3_SDA, FMC_D16, DCMI_HSYNC, LCD_R2, EVENTOUT	-
-	-	K10	M13	86	-	99	N14	PH9	I/O	FT	-	I2C3_SMBA, TIM12_CH2, FMC_D17, DCMI_D0, LCD_R3, EVENTOUT	-
-	-	N11	L13	87	-	100	P15	PH10	I/O	FT	-	TIM5_CH1, FMC_D18, DCMI_D1, LCD_R4, EVENTOUT	-
-	-	M11	L12	88	-	101	N15	PH11	I/O	FT	-	TIM5_CH2, FMC_D19, DCMI_D2, LCD_R5, EVENTOUT	-
-	-	L11	K12	89	-	102	M15	PH12	I/O	FT	-	TIM5_CH3, FMC_D20, DCMI_D3, LCD_R6, EVENTOUT	-
-	-	E7	H12	90	-	-	K10	V_{SS}	S	-	-	-	-
-	-	H8	J12	91	-	103	K11	V_{DD}	S	-	-	-	-

Table 10. STM32F427xx and STM32F429xx pin and ball definitions (continued)

Pin number								Pin name (function after reset)(1)	Pin type	I / O structure	Notes	Alternate functions	Additional functions
LQFP100	LQFP144	UFBGA169	UFBGA176	LQFP176	WLCSP143	LQFP208	TFBGA216						
51	73	N12	P12	92	M2	104	L13	PB12	I/O	FT	-	TIM1_BKIN, I2C2_SMBA, SPI2_NSS/I2S2_WS, USART3_CK, CAN2_RX, OTG_HS_ULPI_D5, ETH_MII_TXD0/ETH_RMII_TXD0, OTG_HS_ID, EVENTOUT	-
52	74	M12	P13	93	N1	105	K14	PB13	I/O	FT	-	TIM1_CH1N, SPI2_SCK/I2S2_CK, USART3_CTS, CAN2_TX, OTG_HS_ULPI_D6, ETH_MII_TXD1/ETH_RMII_TXD1, EVENTOUT	OTG_HS_VBUS
53	75	M13	R14	94	K3	106	R14	PB14	I/O	FT	-	TIM1_CH2N, TIM8_CH2N, SPI2_MISO, I2S2ext_SD, USART3_RTS, TIM12_CH1, OTG_HS_DM, EVENTOUT	-
54	76	L13	R15	95	J3	107	R15	PB15	I/O	FT	-	RTC_REFIN, TIM1_CH3N, TIM8_CH3N, SPI2_MOSI/I2S2_SD, TIM12_CH2, OTG_HS_DP, EVENTOUT	-
55	77	L12	P15	96	L2	108	L15	PD8	I/O	FT	-	USART3_TX, FMC_D13, EVENTOUT	-
56	78	K13	P14	97	M1	109	L14	PD9	I/O	FT	-	USART3_RX, FMC_D14, EVENTOUT	-
57	79	K11	N15	98	H4	110	K15	PD10	I/O	FT	-	USART3_CK, FMC_D15, LCD_B3, EVENTOUT	-

Table 10. STM32F427xx and STM32F429xx pin and ball definitions (continued)

Pin number								Pin name (function after reset)(1)	Pin type	I / O structure	Notes	Alternate functions	Additional functions
LQFP100	LQFP144	UFBGA169	UFBGA176	LQFP176	WLCSP143	LQFP208	TFBGA216						
58	80	H10	N14	99	K2	111	N10	PD11	I/O	FT	-	USART3_CTS, FMC_A16, EVENTOUT	-
59	81	J13	N13	100	H6	112	M10	PD12	I/O	FT	-	TIM4_CH1, USART3_RTS, FMC_A17, EVENTOUT	-
60	82	K12	M15	101	H5	113	M11	PD13	I/O	FT	-	TIM4_CH2, FMC_A18, EVENTOUT	-
-	83	-	-	102	-	114	J10	V_{SS}	S		-	-	-
-	84	F7	J13	103	L1	115	J11	V_{DD}	S		-	-	-
61	85	H11	M14	104	J2	116	L12	PD14	I/O	FT	-	TIM4_CH3, FMC_D0, EVENTOUT	-
62	86	J12	L14	105	K1	117	K13	PD15	I/O	FT	-	TIM4_CH4, FMC_D1, EVENTOUT	-
-	-	-	-	-	-	118	K12	PJ6	I/O	FT	-	LCD_R7, EVENTOUT	-
-	-	-	-	-	-	119	J12	PJ7	I/O	FT	-	LCD_G0, EVENTOUT	-
-	-	-	-	-	-	120	H12	PJ8	I/O	FT	-	LCD_G1, EVENTOUT	-
-	-	-	-	-	-	121	J13	PJ9	I/O	FT	-	LCD_G2, EVENTOUT	-
-	-	-	-	-	-	122	H13	PJ10	I/O	FT	-	LCD_G3, EVENTOUT	-
-	-	-	-	-	-	123	G12	PJ11	I/O	FT	-	LCD_G4, EVENTOUT	-
-	-	-	-	-	-	124	H11	VDD	I/O	FT	-	-	-
-	-	-	-	-	-	125	H10	VSS	I/O	FT	-	-	-
-	-	-	-	-	-	126	G13	PK0	I/O	FT	-	LCD_G5, EVENTOUT	-
-	-	-	-	-	-	127	F12	PK1	I/O	FT	-	LCD_G6, EVENTOUT	-
-	-	-	-	-	-	128	F13	PK2	I/O	FT	-	LCD_G7, EVENTOUT	-
-	87	H13	L15	106	J1	129	M13	PG2	I/O	FT	-	FMC_A12, EVENTOUT	-
-	88	NC (2)	K15	107	G3	130	M12	PG3	I/O	FT	-	FMC_A13, EVENTOUT	-
-	89	H12	K14	108	G5	131	N12	PG4	I/O	FT	-	FMC_A14/FMC_BA0, EVENTOUT	-
-	90	G13	K13	109	G6	132	N11	PG5	I/O	FT	-	FMC_A15/FMC_BA1, EVENTOUT	-

Table 10. STM32F427xx and STM32F429xx pin and ball definitions (continued)

Pin number								Pin name (function after reset)(1)	Pin type	I / O structure	Notes	Alternate functions	Additional functions
LQFP100	LQFP144	UFBGA169	UFBGA176	LQFP176	WLCSP143	LQFP208	TFBGA216						
-	91	G11	J15	110	G4	133	J15	PG6	I/O	FT	-	FMC_INT2, DCMI_D12, LCD_R7, EVENTOUT	-
-	92	G12	J14	111	H1	134	J14	PG7	I/O	FT	-	USART6_CK, FMC_INT3, DCMI_D13, LCD_CLK, EVENTOUT	-
-	93	F13	H14	112	G2	135	H14	PG8	I/O	FT	-	SPI6_NSS, USART6_RTS, ETH_PPS_OUT, FMC_SDCLK, EVENTOUT	-
-	94	J7	G12	113	D2	136	G10	V_{SS}	S		-	-	-
-	95	E6	H13	114	G1	137	G11	V_{DD}	S		-	-	-
63	96	F9	H15	115	F2	138	H15	PC6	I/O	FT	-	TIM3_CH1, TIM8_CH1, I2S2_MCK, USART6_TX, SDIO_D6, DCMI_D0, LCD_HSYNC, EVENTOUT	-
64	97	F10	G15	116	F3	139	G15	PC7	I/O	FT	-	TIM3_CH2, TIM8_CH2, I2S3_MCK, USART6_RX, SDIO_D7, DCMI_D1, LCD_G6, EVENTOUT	-
65	98	F11	G14	117	E4	140	G14	PC8	I/O	FT	-	TIM3_CH3, TIM8_CH3, USART6_CK, SDIO_D0, DCMI_D2, EVENTOUT	-
66	99	F12	F14	118	E3	141	F14	PC9	I/O	FT	-	MCO2, TIM3_CH4, TIM8_CH4, I2C3_SDA, I2S_CKIN, SDIO_D1, DCMI_D3, EVENTOUT	-
67	100	E13	F15	119	F1	142	F15	PA8	I/O	FT	-	MCO1, TIM1_CH1, I2C3_SCL, USART1_CK, OTG_FS_SOF, LCD_R6, EVENTOUT	-

Table 10. STM32F427xx and STM32F429xx pin and ball definitions (continued)

Pin number								Pin name (function after reset)[1]	Pin type	I / O structure	Notes	Alternate functions	Additional functions
LQFP100	LQFP144	UFBGA169	UFBGA176	LQFP176	WLCSP143	LQFP208	TFBGA216						
68	101	E8	E15	120	E2	143	E15	PA9	I/O	FT	-	TIM1_CH2, I2C3_SMBA, USART1_TX, DCMI_D0, EVENTOUT	OTG_FS_VBUS
69	102	E9	D15	121	D5	144	D15	PA10	I/O	FT	-	TIM1_CH3, USART1_RX, OTG_FS_ID, DCMI_D1, EVENTOUT	-
70	103	E10	C15	122	D4	145	C15	PA11	I/O	FT	-	TIM1_CH4, USART1_CTS, CAN1_RX, LCD_R4, OTG_FS_DM, EVENTOUT	-
71	104	E11	B15	123	E1	146	B15	PA12	I/O	FT	-	TIM1_ETR, USART1_RTS, CAN1_TX, LCD_R5, OTG_FS_DP, EVENTOUT	-
72	105	E12	A15	124	D3	147	A15	PA13 (JTMS-SWDIO)	I/O	FT	-	JTMS-SWDIO, EVENTOUT	-
73	106	D12	F13	125	D1	148	E11	V_{CAP_2}	S		-	-	-
74	107	J10	F12	126	D2	149	F10	V_{SS}	S		-	-	-
75	108	H4	G13	127	C1	150	F11	V_{DD}	S		-	-	-
-	-	D13	E12	128	-	151	E12	PH13	I/O	FT	-	TIM8_CH1N, CAN1_TX, FMC_D21, LCD_G2, EVENTOUT	-
-	-	C13	E13	129	-	152	E13	PH14	I/O	FT	-	TIM8_CH2N, FMC_D22, DCMI_D4, LCD_G3, EVENTOUT	-
-	-	C12	D13	130	-	153	D13	PH15	I/O	FT	-	TIM8_CH3N, FMC_D23, DCMI_D11, LCD_G4, EVENTOUT	-
-	-	B13	E14	131	-	154	E14	PI0	I/O	FT	-	TIM5_CH4, SPI2_NSS/I2S2_WS[7], FMC_D24, DCMI_D13, LCD_G5, EVENTOUT	-

Table 10. STM32F427xx and STM32F429xx pin and ball definitions (continued)

Pin number								Pin name (function after reset)(1)	Pin type	I / O structure	Notes	Alternate functions	Additional functions
LQFP100	LQFP144	UFBGA169	UFBGA176	LQFP176	WLCSP143	LQFP208	TFBGA216						
-	-	C11	D14	132	-	155	D14	PI1	I/O	FT	-	SPI2_SCK/I2S2_CK(7), FMC_D25, DCMI_D8, LCD_G6, EVENTOUT	-
-	-	B12	C14	133	-	156	C14	PI2	I/O	FT	-	TIM8_CH4, SPI2_MISO, I2S2ext_SD, FMC_D26, DCMI_D9, LCD_G7, EVENTOUT	-
-	-	A12	C13	134	-	157	C13	PI3	I/O	FT	-	TIM8_ETR, SPI2_MOSI/I2S2_SD, FMC_D27, DCMI_D10, EVENTOUT	-
-	-	D11	D9	135	F5	-	F9	V_{SS}	S		-	-	-
-	-	D3	C9	136	A1	158	E10	V_{DD}	S		-	-	-
76	109	A11	A14	137	B1	159	A14	PA14 (JTCK-SWCLK)	I/O	FT	-	JTCK-SWCLK/ EVENTOUT	-
77	110	B11	A13	138	C2	160	A13	PA15 (JTDI)	I/O	FT	-	JTDI, TIM2_CH1/TIM2_ETR, SPI1_NSS, SPI3_NSS/I2S3_WS, EVENTOUT	-
78	111	C10	B14	139	A2	161	B14	PC10	I/O	FT	-	SPI3_SCK/I2S3_CK, USART3_TX, UART4_TX, SDIO_D2, DCMI_D8, LCD_R2, EVENTOUT	-
79	112	B10	B13	140	B2	162	B13	PC11	I/O	FT	-	I2S3ext_SD, SPI3_MISO, USART3_RX, UART4_RX, SDIO_D3, DCMI_D4, EVENTOUT	-
80	113	A10	A12	141	C3	163	A12	PC12	I/O	FT	-	SPI3_MOSI/I2S3_SD, USART3_CK, UART5_TX, SDIO_CK, DCMI_D9, EVENTOUT	-
81	114	D9	B12	142	B3	164	B12	PD0	I/O	FT	-	CAN1_RX, FMC_D2, EVENTOUT	-

Table 10. STM32F427xx and STM32F429xx pin and ball definitions (continued)

Pin number								Pin name (function after reset)[1]	Pin type	I / O structure	Notes	Alternate functions	Additional functions
LQFP100	LQFP144	UFBGA169	UFBGA176	LQFP176	WLCSP143	LQFP208	TFBGA216						
82	115	C9	C12	143	C4	165	C12	PD1	I/O	FT	-	CAN1_TX, FMC_D3, EVENTOUT	-
83	116	B9	D12	144	A3	166	D12	PD2	I/O	FT	-	TIM3_ETR, UART5_RX, SDIO_CMD, DCMI_D11, EVENTOUT	-
84	117	A9	D11	145	B4	167	C11	PD3	I/O	FT	-	SPI2_SCK/I2S2_CK, USART2_CTS, FMC_CLK, DCMI_D5, LCD_G7, EVENTOUT	-
85	118	D8	D10	146	B5	168	D11	PD4	I/O	FT	-	USART2_RTS, FMC_NOE, EVENTOUT	-
86	119	C8	C11	147	A4	169	C10	PD5	I/O	FT	-	USART2_TX, FMC_NWE, EVENTOUT	-
-	120	-	D8	148	-	170	F8	V_{SS}	S		-	-	-
-	121	D6	C8	149	C5	171	E9	V_{DD}	S		-	-	-
87	122	B8	B11	150	F4	172	B11	PD6	I/O	FT	-	SPI3_MOSI/I2S3_SD, SAI1_SD_A, USART2_RX, FMC_NWAIT, DCMI_D10, LCD_B2, EVENTOUT	-
88	123	A8	A11	151	A5	173	A11	PD7	I/O	FT	-	USART2_CK, FMC_NE1/FMC_NCE2, EVENTOUT	-
-	-	-	-	-	-	174	B10	PJ12	I/O	FT	-	LCD_B0, EVENTOUT	-
-	-	-	-	-	-	175	B9	PJ13	I/O	FT	-	LCD_B1, EVENTOUT	-
-	-	-	-	-	-	176	C9	PJ14	I/O	FT	-	LCD_B2, EVENTOUT	-
-	-	-	-	-	-	177	D10	PJ15	I/O	FT	-	LCD_B3, EVENTOUT	-
-	124	NC (2)	C10	152	E5	178	D9	PG9	I/O	FT	-	USART6_RX, FMC_NE2/FMC_NCE3, DCMI_VSYNC[8], EVENTOUT	-

Table 10. STM32F427xx and STM32F429xx pin and ball definitions (continued)

Pin number								Pin name (function after reset)(1)	Pin type	I / O structure	Notes	Alternate functions	Additional functions
LQFP100	LQFP144	UFBGA169	UFBGA176	LQFP176	WLCSP143	LQFP208	TFBGA216						
-	125	C7	B10	153	C6	179	C8	PG10	I/O	FT	-	LCD_G3, FMC_NCE4_1/FMC_NE3, DCMI_D2, LCD_B2, EVENTOUT	-
-	126	B7	B9	154	B6	180	B8	PG11	I/O	FT	-	ETH_MII_TX_EN/ETH_RMII_TX_EN, FMC_NCE4_2, DCMI_D3, LCD_B3, EVENTOUT	-
-	127	A7	B8	155	A6	181	C7	PG12	I/O	FT	-	SPI6_MISO, USART6_RTS, LCD_B4, FMC_NE4, LCD_B1, EVENTOUT	-
-	128	NC (2)	A8	156	D6	182	B3	PG13	I/O	FT	-	SPI6_SCK, USART6_CTS, ETH_MII_TXD0/ETH_RMII_TXD0, FMC_A24, EVENTOUT	-
-	129	NC (2)	A7	157	F6	183	A4	PG14	I/O	FT	-	SPI6_MOSI, USART6_TX, ETH_MII_TXD1/ETH_RMII_TXD1, FMC_A25, EVENTOUT	-
-	130	D7	D7	158	-	184	F7	V_{SS}	S		-	-	-
-	131	L6	C7	159	E6	185	E8	V_{DD}	S		-	-	-
-	-	-	-	-	-	186	D8	PK3	I/O	FT	-	LCD_B4, EVENTOUT	-
-	-	-	-	-	-	187	D7	PK4	I/O	FT	-	LCD_B5, EVENTOUT	-
-	-	-	-	-	-	188	C6	PK5	I/O	FT	-	LCD_B6, EVENTOUT	-
-	-	-	-	-	-	189	C5	PK6	I/O	FT	-	LCD_B7, EVENTOUT	-
-	-	-	-	-	-	190	C4	PK7	I/O	FT	-	LCD_DE, EVENTOUT	-
-	132	C6	B7	160	A7	191	B7	PG15	I/O	FT	-	USART6_CTS, FMC_SDNCAS, DCMI_D13, EVENTOUT	-

Table 10. STM32F427xx and STM32F429xx pin and ball definitions (continued)

Pin number								Pin name (function after reset)(1)	Pin type	I / O structure	Notes	Alternate functions	Additional functions
LQFP100	LQFP144	UFBGA169	UFBGA176	LQFP176	WLCSP143	LQFP208	TFBGA216						
89	133	B6	A10	161	B7	192	A10	PB3 (JTDO/TRACESWO)	I/O	FT	-	JTDO/TRACESWO, TIM2_CH2, SPI1_SCK, SPI3_SCK/I2S3_CK, EVENTOUT	-
90	134	A6	A9	162	C7	193	A9	PB4 (NJTRST)	I/O	FT	-	NJTRST, TIM3_CH1, SPI1_MISO, SPI3_MISO, I2S3ext_SD, EVENTOUT	-
91	135	D5	A6	163	C8	194	A8	PB5	I/O	FT	-	TIM3_CH2, I2C1_SMBA, SPI1_MOSI, SPI3_MOSI/I2S3_SD, CAN2_RX, OTG_HS_ULPI_D7, ETH_PPS_OUT, FMC_SDCKE1, DCMI_D10, EVENTOUT	-
92	136	C5	B6	164	A8	195	B6	PB6	I/O	FT	-	TIM4_CH1, I2C1_SCL, USART1_TX, CAN2_TX, FMC_SDNE1, DCMI_D5, EVENTOUT	-
93	137	B5	B5	165	B8	196	B5	PB7	I/O	FT	-	TIM4_CH2, I2C1_SDA, USART1_RX, FMC_NL, DCMI_VSYNC, EVENTOUT	-
94	138	A5	D6	166	C9	197	E6	BOOT0	I	B	-		V_{PP}
95	139	D4	A5	167	A9	198	A7	PB8	I/O	FT	-	TIM4_CH3, TIM10_CH1, I2C1_SCL, CAN1_RX, ETH_MII_TXD3, SDIO_D4, DCMI_D6, LCD_B6, EVENTOUT	-

Table 10. STM32F427xx and STM32F429xx pin and ball definitions (continued)

Pin number								Pin name (function after reset)(1)	Pin type	I / O structure	Notes	Alternate functions	Additional functions
LQFP100	LQFP144	UFBGA169	UFBGA176	LQFP176	WLCSP143	LQFP208	TFBGA216						
96	140	C4	B4	168	B9	199	B4	PB9	I/O	FT	-	TIM4_CH4, TIM11_CH1, I2C1_SDA, SPI2_NSS/I2S2_WS, CAN1_TX, SDIO_D5, DCMI_D7, LCD_B7, EVENTOUT	-
97	141	B4	A4	169	B10	200	A6	PE0	I/O	FT	-	TIM4_ETR, UART8_RX, FMC_NBL0, DCMI_D2, EVENTOUT	-
98	142	A4	A3	170	A10	201	A5	PE1	I/O	FT	-	UART8_Tx, FMC_NBL1, DCMI_D3, EVENTOUT	-
99	-	F5	D5	-	-	202	F6	V_{SS}	S		-		-
-	143	C3	C6	171	A11	203	E5	PDR_ON	S		-		-
100	144	K6	C5	172	D7	204	E7	V_{DD}	S		-		-
-	-	B3	D4	173	-	205	C3	PI4	I/O	FT	-	TIM8_BKIN, FMC_NBL2, DCMI_D5, LCD_B4, EVENTOUT	-
-	-	A3	C4	174	-	206	D3	PI5	I/O	FT	-	TIM8_CH1, FMC_NBL3, DCMI_VSYNC, LCD_B5, EVENTOUT	-
-	-	A2	C3	175	-	207	D6	PI6	I/O	FT	-	TIM8_CH2, FMC_D28, DCMI_D6, LCD_B6, EVENTOUT	-
-	-	B1	C2	176	-	208	D4	PI7	I/O	FT	-	TIM8_CH3, FMC_D29, DCMI_D7, LCD_B7, EVENTOUT	-

1. Function availability depends on the chosen device.
2. NC (not-connected) pins are not bonded. They must be configured by software to output push-pull and forced to 0 in the output data register to avoid extra current consumption in low power modes.
3. PC13, PC14, PC15 and PI8 are supplied through the power switch. Since the switch only sinks a limited amount of current (3 mA), the use of GPIOs PC13 to PC15 and PI8 in output mode is limited:
   - The speed should not exceed 2 MHz with a maximum load of 30 pF.
   - These I/Os must not be used as a current source (e.g. to drive an LED).

4. Main function after the first backup domain power-up. Later on, it depends on the contents of the RTC registers even after reset (because these registers are not reset by the main reset). For details on how to manage these I/Os, refer to the RTC register description sections in the STM32F4xx reference manual, available from the STMicroelectronics website: *www.st.com*.
5. FT = 5 V tolerant except when in analog mode or oscillator mode (for PC14, PC15, PH0 and PH1).
6. If the device is delivered in an WLCSP143, UFBGA169, UFBGA176, LQFP176 or TFBGA216 package, and the BYPASS_REG pin is set to V_{DD} (Regulator OFF/internal reset ON mode), then PA0 is used as an internal Reset (active low).
7. PI0 and PI1 cannot be used for I2S2 full-duplex mode.
8. The DCMI_VSYNC alternate function on PG9 is only available on silicon revision 3.

Table 11. FMC pin definition

Pin name	CF	NOR/PSRAM/ SRAM	NOR/PSRAM Mux	NAND16	SDRAM
PF0	A0	A0			A0
PF1	A1	A1			A1
PF2	A2	A2			A2
PF3	A3	A3			A3
PF4	A4	A4			A4
PF5	A5	A5			A5
PF12	A6	A6			A6
PF13	A7	A7			A7
PF14	A8	A8			A8
PF15	A9	A9			A9
PG0	A10	A10			A10
PG1		A11			A11
PG2		A12			A12
PG3		A13			
PG4		A14			BA0
PG5		A15			BA1
PD11		A16	A16	CLE	
PD12		A17	A17	ALE	
PD13		A18	A18		
PE3		A19	A19		
PE4		A20	A20		
PE5		A21	A21		
PE6		A22	A22		
PE2		A23	A23		
PG13		A24	A24		
PG14		A25	A25		
PD14	D0	D0	DA0	D0	D0
PD15	D1	D1	DA1	D1	D1
PD0	D2	D2	DA2	D2	D2
PD1	D3	D3	DA3	D3	D3
PE7	D4	D4	DA4	D4	D4
PE8	D5	D5	DA5	D5	D5
PE9	D6	D6	DA6	D6	D6
PE10	D7	D7	DA7	D7	D7

Table 11. FMC pin definition (continued)

Pin name	CF	NOR/PSRAM/ SRAM	NOR/PSRAM Mux	NAND16	SDRAM
PE11	D8	D8	DA8	D8	D8
PE12	D9	D9	DA9	D9	D9
PE13	D10	D10	DA10	D10	D10
PE14	D11	D11	DA11	D11	D11
PE15	D12	D12	DA12	D12	D12
PD8	D13	D13	DA13	D13	D13
PD9	D14	D14	DA14	D14	D14
PD10	D15	D15	DA15	D15	D15
PH8		D16			D16
PH9		D17			D17
PH10		D18			D18
PH11		D19			D19
PH12		D20			D20
PH13		D21			D21
PH14		D22			D22
PH15		D23			D23
PI0		D24			D24
PI1		D25			D25
PI2		D26			D26
PI3		D27			D27
PI6		D28			D28
PI7		D29			D29
PI9		D30			D30
PI10		D31			D31
PD7		NE1	NE1	NCE2	
PG9		NE2	NE2	NCE3	
PG10	NCE4_1	NE3	NE3		
PG11	NCE4_2				
PG12		NE4	NE4		
PD3		CLK	CLK		
PD4	NOE	NOE	NOE	NOE	
PD5	NWE	NWE	NWE	NWE	
PD6	NWAIT	NWAIT	NWAIT	NWAIT	
PB7		NL(NADV)	NL(NADV)		

Table 11. FMC pin definition (continued)

Pin name	CF	NOR/PSRAM/ SRAM	NOR/PSRAM Mux	NAND16	SDRAM
PF6	NIORD				
PF7	NREG				
PF8	NIOWR				
PF9	CD				
PF10	INTR				
PG6				INT2	
PG7				INT3	
PE0		NBL0	NBL0		NBL0
PE1		NBL1	NBL1		NBL1
PI4		NBL2			NBL2
PI5		NBL3			NBL3
PG8					SDCLK
PC0					SDNWE
PF11					SDNRAS
PG15					SDNCAS
PH2					SDCKE0
PH3					SDNE0
PH6					SDNE1
PH7					SDCKE1
PH5					SDNWE
PC2					SDNE0
PC3					SDCKE0
PB5					SDCKE1
PB6					SDNE1

Table 12. STM32F427xx and STM32F429xx alternate function mapping

Port		AF0	AF1	AF2	AF3	AF4	AF5	AF6	AF7	AF8	AF9	AF10	AF11	AF12	AF13	AF14	AF15
		SYS	TIM1/2	TIM3/4/5	TIM8/9/10/11	I2C1/2/3	SPI1/2/3/4/5/6	SPI2/3/SAI1	SPI3/USART1/2/3	USART6/UART4/5/7/8	CAN1/2/TIM12/13/14/LCD	OTG2_HS/OTG1_FS	ETH	FMC/SDIO/OTG2_FS	DCMI	LCD	SYS
Port A	PA0	-	TIM2_CH1/TIM2_ETR	TIM5_CH1	TIM8_ETR	-	-	-	USART2_CTS	UART4_TX	-	-	ETH_MII_CRS	-	-	-	EVENTOUT
	PA1	-	TIM2_CH2	TIM5_CH2	-	-	-	-	USART2_RTS	UART4_RX	-	-	ETH_MII_RX_CLK/ETH_RMII_REF_CLK	-	-	-	EVENTOUT
	PA2	-	TIM2_CH3	TIM5_CH3	TIM9_CH1	-	-	-	USART2_TX	-	-	-	ETH_MDIO	-	-	-	EVENTOUT
	PA3	-	TIM2_CH4	TIM5_CH4	TIM9_CH2	-	-	-	USART2_RX	-	-	OTG_HS_ULPI_D0	ETH_MII_COL	-	-	LCD_B5	EVENTOUT
	PA4	-	-	-	-	-	SPI1_NSS	SPI3_NSS/I2S3_WS	USART2_CK	-	-	-	-	OTG_HS_SOF	DCMI_HSYNC	LCD_VSYNC	EVENTOUT
	PA5	-	TIM2_CH1/TIM2_ETR	-	TIM8_CH1N	-	SPI1_SCK	-	-	-	-	OTG_HS_ULPI_CK	-	-	-	-	EVENTOUT
	PA6	-	TIM1_BKIN	TIM3_CH1	TIM8_BKIN	-	SPI1_MISO	-	-	-	TIM13_CH1	-	-	-	DCMI_PIXCLK	LCD_G2	EVENTOUT
	PA7	-	TIM1_CH1N	TIM3_CH2	TIM8_CH1N	-	SPI1_MOSI	-	-	-	TIM14_CH1	-	ETH_MII_RX_DV/ETH_RMII_CRS_DV	-	-	-	EVENTOUT
	PA8	MCO1	TIM1_CH1	-	-	I2C3_SCL	-	-	USART1_CK	-	-	OTG_FS_SOF	-	-	-	LCD_R6	EVENTOUT
	PA9	-	TIM1_CH2	-	-	I2C3_SMBA	-	-	USART1_TX	-	-	-	-	-	DCMI_D0	-	EVENTOUT
	PA10	-	TIM1_CH3	-	-	-	-	-	USART1_RX	-	-	OTG_FS_ID	-	-	DCMI_D1	-	EVENTOUT
	PA11	-	TIM1_CH4	-	-	-	-	-	USART1_CTS	-	CAN1_RX	OTG_FS_DM	-	-	-	LCD_R4	EVENTOUT
	PA12	-	TIM1_ETR	-	-	-	-	-	USART1_RTS	-	CAN1_TX	OTG_FS_DP	-	-	-	LCD_R5	EVENTOUT

Table 12. STM32F427xx and STM32F429xx alternate function mapping (continued)

Port		AF0	AF1	AF2	AF3	AF4	AF5	AF6	AF7	AF8	AF9	AF10	AF11	AF12	AF13	AF14	AF15
		SYS	TIM1/2	TIM3/4/5	TIM8/9/10/11	I2C1/2/3	SPI1/2/3/4/5/6	SPI2/3/SAI1	SPI3/USART1/2/3	USART6/UART4/5/7/8	CAN1/2/TIM12/13/14/LCD	OTG2_HS/OTG1_FS	ETH	FMC/SDIO/OTG2_FS	DCMI	LCD	SYS
Port A	PA13	JTMS-SWDIO	-	-	-	-	-	-	-	-	-	-	-	-	-	-	EVENTOUT
	PA14	JTCK-SWCLK	-	-	-	-	-	-	-	-	-	-	-	-	-	-	EVENTOUT
	PA15	JTDI	TIM2_CH1/TIM2_ETR	-	-	-	SPI1_NSS	SPI3_NSS/I2S3_WS	-	-	-	-	-	-	-	-	EVENTOUT
Port B	PB0	-	TIM1_CH2N	TIM3_CH3	TIM8_CH2N	-	-	-	-	-	LCD_R3	OTG_HS_ULPI_D1	ETH_MII_RXD2	-	-	-	EVENTOUT
	PB1	-	TIM1_CH3N	TIM3_CH4	TIM8_CH3N	-	-	-	-	-	LCD_R6	OTG_HS_ULPI_D2	ETH_MII_RXD3	-	-	-	EVENTOUT
	PB2	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	EVENTOUT
	PB3	JTDO/TRACESWO	TIM2_CH2	-	-	-	SPI1_SCK	SPI3_SCK/I2S3_CK	-	-	-	-	-	-	-	-	EVENTOUT
	PB4	NJTRST	-	TIM3_CH1	-	-	SPI1_MISO	SPI3_MISO	I2S3ext_SD	-	-	-	-	-	-	-	EVENTOUT
	PB5	-	-	TIM3_CH2	-	I2C1_SMBA	SPI1_MOSI	SPI3_MOSI/I2S3_SD	-	-	CAN2_RX	OTG_HS_ULPI_D7	ETH_PPS_OUT	FMC_SDCKE1	DCMI_D10	-	EVENTOUT
	PB6	-	-	TIM4_CH1	-	I2C1_SCL	-	-	USART1_TX	-	CAN2_TX	-	-	FMC_SDNE1	DCMI_D5	-	EVENTOUT
	PB7	-	-	TIM4_CH2	-	I2C1_SDA	-	-	USART1_RX	-	-	-	-	FMC_NL	DCMI_VSYNC	-	EVENTOUT
	PB8	-	-	TIM4_CH3	TIM10_CH1	I2C1_SCL	-	-	-	-	CAN1_RX	-	ETH_MII_TXD3	SDIO_D4	DCMI_D6	LCD_B6	EVENTOUT
	PB9	-	-	TIM4_CH4	TIM11_CH1	I2C1_SDA	SPI2_NSS/I2S2_WS	-	-	-	CAN1_TX	-	-	SDIO_D5	DCMI_D7	LCD_B7	EVENTOUT
	PB10	-	TIM2_CH3	-	-	I2C2_SCL	SPI2_SCK/I2S2_CK	-	USART3_TX	-	-	OTG_HS_ULPI_D3	ETH_MII_RX_ER	-	-	LCD_G4	EVENTOUT

Table 12. STM32F427xx and STM32F429xx alternate function mapping (continued)

Port		AF0	AF1	AF2	AF3	AF4	AF5	AF6	AF7	AF8	AF9	AF10	AF11	AF12	AF13	AF14	AF15
		SYS	TIM1/2	TIM3/4/5	TIM8/9/10/11	I2C1/2/3	SPI1/2/3/4/5/6	SPI2/3/SAI1	SPI3/USART1/2/3	USART6/UART4/5/7/8	CAN1/2/TIM12/13/14/LCD	OTG2_HS/OTG1_FS	ETH	FMC/SDIO/OTG2_FS	DCMI	LCD	SYS
Port B	PB11	-	TIM2_CH4	-	-	I2C2_SDA	-	-	USART3_RX	-	-	OTG_HS_ULPI_D4	ETH_MII_TX_EN/ETH_RMII_TX_EN	-	-	LCD_G5	EVENTOUT
	PB12	-	TIM1_BKIN	-	-	I2C2_SMBA	SPI2_NSS/I2S2_WS	-	USART3_CK	-	CAN2_RX	OTG_HS_ULPI_D5	ETH_MII_TXD0/ETH_RMII_TXD0	OTG_HS_ID	-	-	EVENTOUT
	PB13	-	TIM1_CH1N	-	-	-	SPI2_SCK/I2S2_CK	-	USART3_CTS	-	CAN2_TX	OTG_HS_ULPI_D6	ETH_MII_TXD1/ETH_RMII_TXD1	-	-	-	EVENTOUT
	PB14	-	TIM1_CH2N	-	TIM8_CH2N	-	SPI2_MISO	I2S2ext_SD	USART3_RTS	-	TIM12_CH1	-	-	OTG_HS_DM	-	-	EVENTOUT
	PB15	RTC_REFIN	TIM1_CH3N	-	TIM8_CH3N	-	SPI2_MOSI/I2S2_SD	-	-	-	TIM12_CH2	-	-	OTG_HS_DP	-	-	EVENTOUT
Port C	PC0	-	-	-	-	-	-	-	-	-	-	OTG_HS_ULPI_STP	-	FMC_SDNWE	-	-	EVENTOUT
	PC1	-	-	-	-	-	-	-	-	-	-	-	ETH_MDC	-	-	-	EVENTOUT
	PC2	-	-	-	-	-	SPI2_MISO	I2S2ext_SD	-	-	-	OTG_HS_ULPI_DIR	ETH_MII_TXD2	FMC_SDNE0	-	-	EVENTOUT
	PC3	-	-	-	-	-	SPI2_MOSI/I2S2_SD	-	-	-	-	OTG_HS_ULPI_NXT	ETH_MII_TX_CLK	FMC_SDCKE0	-	-	EVENTOUT
	PC4	-	-	-	-	-	-	-	-	-	-	-	ETH_MII_RXD0/ETH_RMII_RXD0	-	-	-	EVENTOUT
	PC5	-	-	-	-	-	-	-	-	-	-	-	ETH_MII_RXD1/ETH_RMII_RXD1	-	-	-	EVENTOUT
	PC6	-	-	TIM3_CH1	TIM8_CH1	-	I2S2_MCK	-	-	USART6_TX	-	-	-	SDIO_D6	DCMI_D0	LCD_HSYNC	EVENTOUT
	PC7	-	-	TIM3_CH2	TIM8_CH2	-	-	I2S3_MCK	-	USART6_RX	-	-	-	SDIO_D7	DCMI_D1	LCD_G6	EVENTOUT

Table 12. STM32F427xx and STM32F429xx alternate function mapping (continued)

Port		AF0	AF1	AF2	AF3	AF4	AF5	AF6	AF7	AF8	AF9	AF10	AF11	AF12	AF13	AF14	AF15
		SYS	TIM1/2	TIM3/4/5	TIM8/9/10/11	I2C1/2/3	SPI1/2/3/4/5/6	SPI2/3/SAI1	SPI3/USART1/2/3	USART6/UART4/5/7/8	CAN1/2/TIM12/13/14/LCD	OTG2_HS/OTG1_FS	ETH	FMC/SDIO/OTG2_FS	DCMI	LCD	SYS
Port C	PC8	-	-	TIM3_CH3	TIM8_CH3	-	-	-	-	USART6_CK	-	-	-	SDIO_D0	DCMI_D2	-	EVENTOUT
	PC9	MCO2	-	TIM3_CH4	TIM8_CH4	I2C3_SDA	I2S_CKIN	-	-	-	-	-	-	SDIO_D1	DCMI_D3	-	EVENTOUT
	PC10	-	-	-	-	-	-	SPI3_SCK/I2S3_CK	USART3_TX	UART4_TX	-	-	-	SDIO_D2	DCMI_D8	LCD_R2	EVENTOUT
	PC11	-	-	-	-	-	I2S3ext_SD	SPI3_MISO	USART3_RX	UART4_RX	-	-	-	SDIO_D3	DCMI_D4	-	EVENTOUT
	PC12	-	-	-	-	-	-	SPI3_MOSI/I2S3_SD	USART3_CK	UART5_TX	-	-	-	SDIO_CK	DCMI_D9	-	EVENTOUT
	PC13	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	EVENTOUT
	PC14	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	EVENTOUT
	PC15	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	EVENTOUT
Port D	PD0	-	-	-	-	-	-	-	-	-	CAN1_RX	-	-	FMC_D2	-	-	EVENTOUT
	PD1	-	-	-	-	-	-	-	-	-	CAN1_TX	-	-	FMC_D3	-	-	EVENTOUT
	PD2	-	-	TIM3_ETR	-	-	-	-	-	UART5_RX	-	-	-	SDIO_CMD	DCMI_D11	-	EVENTOUT
	PD3	-	-	-	-	-	SPI2_SCK/I2S2_CK	-	USART2_CTS	-	-	-	-	FMC_CLK	DCMI_D5	LCD_G7	EVENTOUT
	PD4	-	-	-	-	-	-	-	USART2_RTS	-	-	-	-	FMC_NOE	-	-	EVENTOUT
	PD5	-	-	-	-	-	-	-	USART2_TX	-	-	-	-	FMC_NWE	-	-	EVENTOUT
	PD6	-	-	-	-	-	SPI3_MOSI/I2S3_SD	SAI1_SD_A	USART2_RX	-	-	-	-	FMC_NWAIT	DCMI_D10	LCD_B2	EVENTOUT

Table 12. STM32F427xx and STM32F429xx alternate function mapping (continued)

Port		AF0	AF1	AF2	AF3	AF4	AF5	AF6	AF7	AF8	AF9	AF10	AF11	AF12	AF13	AF14	AF15
		SYS	TIM1/2	TIM3/4/5	TIM8/9/10/11	I2C1/2/3	SPI1/2/3/4/5/6	SPI2/3/SAI1	SPI3/USART1/2/3	USART6/UART4/5/7/8	CAN1/2/TIM12/13/14/LCD	OTG2_HS/OTG1_FS	ETH	FMC/SDIO/OTG2_FS	DCMI	LCD	SYS
Port D	PD7	-	-	-	-	-	-	-	USART2_CK	-	-	-	-	FMC_NE1/FMC_NCE2	-	-	EVENTOUT
	PD8	-	-	-	-	-	-	-	USART3_TX	-	-	-	-	FMC_D13	-	-	EVENTOUT
	PD9	-	-	-	-	-	-	-	USART3_RX	-	-	-	-	FMC_D14	-	-	EVENTOUT
	PD10	-	-	-	-	-	-	-	USART3_CK	-	-	-	-	FMC_D15	-	LCD_B3	EVENTOUT
	PD11	-	-	-	-	-	-	-	USART3_CTS	-	-	-	-	FMC_A16	-	-	EVENTOUT
	PD12	-	-	TIM4_CH1	-	-	-	-	USART3_RTS	-	-	-	-	FMC_A17	-	-	EVENTOUT
	PD13	-	-	TIM4_CH2	-	-	-	-	-	-	-	-	-	FMC_A18	-	-	EVENTOUT
	PD14	-	-	TIM4_CH3	-	-	-	-	-	-	-	-	-	FMC_D0	-	-	EVENTOUT
	PD15	-	-	TIM4_CH4	-	-	-	-	-	-	-	-	-	FMC_D1	-	-	EVENTOUT
Port E	PE0	-	-	TIM4_ETR	-	-	-	-	-	UART8_Rx	-	-	-	FMC_NBL0	DCMI_D2	-	EVENTOUT
	PE1	-	-	-	-	-	-	-	-	UART8_Tx	-	-	-	FMC_NBL1	DCMI_D3	-	EVENTOUT
	PE2	TRACECLK	-	-	-	-	SPI4_SCK	SAI1_MCLK_A	-	-	-	-	ETH_MII_TXD3	FMC_A23	-	-	EVENTOUT
	PE3	TRACED0	-	-	-	-	-	SAI1_SD_B	-	-	-	-	-	FMC_A19	-	-	EVENTOUT
	PE4	TRACED1	-	-	-	-	SPI4_NSS	SAI1_FS_A	-	-	-	-	-	FMC_A20	DCMI_D4	LCD_B0	EVENTOUT
	PE5	TRACED2	-	-	TIM9_CH1	-	SPI4_MISO	SAI1_SCK_A	-	-	-	-	-	FMC_A21	DCMI_D6	LCD_G0	EVENTOUT
	PE6	TRACED3	-	-	TIM9_CH2	-	SPI4_MOSI	SAI1_SD_A	-	-	-	-	-	FMC_A22	DCMI_D7	LCD_G1	EVENTOUT

Table 12. STM32F427xx and STM32F429xx alternate function mapping (continued)

Port		AF0	AF1	AF2	AF3	AF4	AF5	AF6	AF7	AF8	AF9	AF10	AF11	AF12	AF13	AF14	AF15
		SYS	TIM1/2	TIM3/4/5	TIM8/9/10/11	I2C1/2/3	SPI1/2/3/4/5/6	SPI2/3/SAI1	SPI3/USART1/2/3	USART6/UART4/5/7/8	CAN1/2/TIM12/13/14/LCD	OTG2_HS/OTG1_FS	ETH	FMC/SDIO/OTG2_FS	DCMI	LCD	SYS
Port E	PE7	-	TIM1_ETR	-	-	-	-	-	-	UART7_Rx	-	-	-	FMC_D4	-	-	EVENTOUT
	PE8	-	TIM1_CH1N	-	-	-	-	-	-	UART7_Tx	-	-	-	FMC_D5	-	-	EVENTOUT
	PE9	-	TIM1_CH1	-	-	-	-	-	-	-	-	-	-	FMC_D6	-	-	EVENTOUT
	PE10	-	TIM1_CH2N	-	-	-	-	-	-	-	-	-	-	FMC_D7	-	-	EVENTOUT
	PE11	-	TIM1_CH2	-	-	-	SPI4_NSS	-	-	-	-	-	-	FMC_D8	-	LCD_G3	EVENTOUT
	PE12	-	TIM1_CH3N	-	-	-	SPI4_SCK	-	-	-	-	-	-	FMC_D9	-	LCD_B4	EVENTOUT
	PE13	-	TIM1_CH3	-	-	-	SPI4_MISO	-	-	-	-	-	-	FMC_D10	-	LCD_DE	EVENTOUT
	PE14	-	TIM1_CH4	-	-	-	SPI4_MOSI	-	-	-	-	-	-	FMC_D11	-	LCD_CLK	EVENTOUT
	PE15	-	TIM1_BKIN	-	-	-		-	-	-	-	-	-	FMC_D12	-	LCD_R7	EVENTOUT
Port F	PF0	-	-	-	-	I2C2_SDA	-	-	-	-	-	-	-	FMC_A0	-	-	EVENTOUT
	PF1	-				I2C2_SCL	-	-	-	-	-	-	-	FMC_A1	-	-	EVENTOUT
	PF2	-	-	-	-	I2C2_SMBA	-	-	-	-	-	-	-	FMC_A2	-	-	EVENTOUT
	PF3	-	-	-	-		-	-	-	-	-	-	-	FMC_A3	-	-	EVENTOUT
	PF4	-	-	-	-		-	-	-	-	-	-	-	FMC_A4	-	-	EVENTOUT
	PF5	-	-	-	-		-	-	-	-	-	-	-	FMC_A5	-	-	EVENTOUT
	PF6	-	-	-	TIM10_CH1	-	SPI5_NSS	SAI1_SD_B	-	UART7_Rx	-	-	-	FMC_NIORD	-	-	EVENTOUT
	PF7	-	-	-	TIM11_CH1	-	SPI5_SCK	SAI1_MCLK_B	-	UART7_Tx	-	-	-	FMC_NREG	-	-	EVENTOUT

Table 12. STM32F427xx and STM32F429xx alternate function mapping (continued)

Port		AF0	AF1	AF2	AF3	AF4	AF5	AF6	AF7	AF8	AF9	AF10	AF11	AF12	AF13	AF14	AF15
		SYS	TIM1/2	TIM3/4/5	TIM8/9/10/11	I2C1/2/3	SPI1/2/3/4/5/6	SPI2/3/SAI1	SPI3/USART1/2/3	USART6/UART4/5/7/8	CAN1/2/TIM12/13/14/LCD	OTG2_HS/OTG1_FS	ETH	FMC/SDIO/OTG2_FS	DCMI	LCD	SYS
Port F	PF8	-	-	-	-	-	SPI5_MISO	SAI1_SCK_B	-	-	TIM13_CH1	-	-	FMC_NIOWR	-	-	EVENTOUT
	PF9	-	-	-	-	-	SPI5_MOSI	SAI1_FS_B	-	-	TIM14_CH1	-	-	FMC_CD	-	-	EVENTOUT
	PF10	-	-	-	-	-	-	-	-	-	-	-	-	FMC_INTR	DCMI_D11	LCD_DE	EVENTOUT
	PF11	-	-	-	-	-	SPI5_MOSI	-	-	-	-	-	-	FMC_SDNRAS	DCMI_D12	-	EVENTOUT
	PF12	-	-	-	-	-	-	-	-	-	-	-	-	FMC_A6	-	-	EVENTOUT
	PF13	-	-	-	-	-	-	-	-	-	-	-	-	FMC_A7	-	-	EVENTOUT
	PF14	-	-	-	-	-	-	-	-	-	-	-	-	FMC_A8	-	-	EVENTOUT
	PF15	-	-	-	-	-	-	-	-	-	-	-	-	FMC_A9	-	-	EVENTOUT
Port G	PG0	-	-	-	-	-	-	-	-	-	-	-	-	FMC_A10	-	-	EVENTOUT
	PG1	-	-	-	-	-	-	-	-	-	-	-	-	FMC_A11	-	-	EVENTOUT
	PG2	-	-	-	-	-	-	-	-	-	-	-	-	FMC_A12	-	-	EVENTOUT
	PG3	-	-	-	-	-	-	-	-	-	-	-	-	FMC_A13	-	-	EVENTOUT
	PG4	-	-	-	-	-	-	-	-	-	-	-	-	FMC_A14/FMC_BA0	-	-	EVENTOUT
	PG5	-	-	-	-	-	-	-	-	-	-	-	-	FMC_A15/FMC_BA1	-	-	EVENTOUT
	PG6	-	-	-	-	-	-	-	-	-	-	-	-	FMC_INT2	DCMI_D12	LCD_R7	EVENTOUT
	PG7	-	-	-	-	-	-	-	-	USART6_CK	-	-	-	FMC_INT3	DCMI_D13	LCD_CLK	EVENTOUT
	PG8	-	-	-	-	-	SPI6_NSS	-	-	USART6_RTS	-	-	ETH_PPS_OUT	FMC_SDCLK	-	-	EVENTOUT

Table 12. STM32F427xx and STM32F429xx alternate function mapping (continued)

Port		AF0	AF1	AF2	AF3	AF4	AF5	AF6	AF7	AF8	AF9	AF10	AF11	AF12	AF13	AF14	AF15
		SYS	TIM1/2	TIM3/4/5	TIM8/9/10/11	I2C1/2/3	SPI1/2/3/4/5/6	SPI2/3/SAI1	SPI3/USART1/2/3	USART6/UART4/5/7/8	CAN1/2/TIM12/13/14/LCD	OTG2_HS/OTG1_FS	ETH	FMC/SDIO/OTG2_FS	DCMI	LCD	SYS
Port G	PG9	-	-	-	-	-	-	-	-	USART6_RX	-	-	-	FMC_NE2/FMC_NCE3	DCMI_VSYNC (1)	-	EVENTOUT
	PG10	-	-	-	-	-	-	-	-	-	LCD_G3	-	-	FMC_NCE4_1/FMC_NE3	DCMI_D2	LCD_B2	EVENTOUT
	PG11	-	-	-	-	-	-	-	-	-	-	-	ETH_MII_TX_EN/ETH_RMII_TX_EN	FMC_NCE4_2	DCMI_D3	LCD_B3	EVENTOUT
	PG12	-	-	-	-	-	SPI6_MISO	-	-	USART6_RTS	LCD_B4	-	-	FMC_NE4	-	LCD_B1	EVENTOUT
	PG13	-	-	-	-	-	SPI6_SCK	-	-	USART6_CTS	-	-	ETH_MII_TXD0/ETH_RMII_TXD0	FMC_A24	-	-	EVENTOUT
	PG14	-	-	-	-	-	SPI6_MOSI	-	-	USART6_TX	-	-	ETH_MII_TXD1/ETH_RMII_TXD1	FMC_A25	-	-	EVENTOUT
	PG15	-	-	-	-	-	-	-	-	USART6_CTS	-	-	-	FMC_SDNCAS	DCMI_D13	-	EVENTOUT
Port H	PH0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	EVENTOUT
	PH1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	EVENTOUT
	PH2	-	-	-	-	-	-	-	-	-	-	-	ETH_MII_CRS	FMC_SDCKE0	-	LCD_R0	EVENTOUT
	PH3	-	-	-	-	-	-	-	-	-	-	-	ETH_MII_COL	FMC_SDNE0	-	LCD_R1	EVENTOUT
	PH4	-	-	-	-	I2C2_SCL	-	-	-	-	-	OTG_HS_ULPI_NXT	-	-	-	-	EVENTOUT
	PH5	-	-	-	-	I2C2_SDA	SPI5_NSS	-	-	-	-	-	-	FMC_SDNWE	-	-	EVENTOUT
	PH6	-	-	-	-	I2C2_SMBA	SPI5_SCK	-	-	-	TIM12_CH1	-	-	FMC_SDNE1	DCMI_D8	-	-

Table 12. STM32F427xx and STM32F429xx alternate function mapping (continued)

Port		AF0	AF1	AF2	AF3	AF4	AF5	AF6	AF7	AF8	AF9	AF10	AF11	AF12	AF13	AF14	AF15
		SYS	TIM1/2	TIM3/4/5	TIM8/9/10/11	I2C1/2/3	SPI1/2/3/4/5/6	SPI2/3/SAI1	SPI3/USART1/2/3	USART6/UART4/5/7/8	CAN1/2/TIM12/13/14/LCD	OTG2_HS/OTG1_FS	ETH	FMC/SDIO/OTG2_FS	DCMI	LCD	SYS
Port H	PH7	-	-	-	-	I2C3_SCL	SPI5_MISO	-	-	-	-	-	ETH_MII_RXD3	FMC_SDCKE1	DCMI_D9	-	-
	PH8	-	-	-	-	I2C3_SDA	-	-	-	-	-	-	-	FMC_D16	DCMI_HSYNC	LCD_R2	EVENTOUT
	PH9	-	-	-	-	I2C3_SMBA	-	-	-	-	TIM12_CH2	-	-	FMC_D17	DCMI_D0	LCD_R3	EVENTOUT
	PH10	-	-	TIM5_CH1	-	-	-	-	-	-	-	-	-	FMC_D18	DCMI_D1	LCD_R4	EVENTOUT
	PH11	-	-	TIM5_CH2	-	-	-	-	-	-	-	-	-	FMC_D19	DCMI_D2	LCD_R5	EVENTOUT
	PH12	-	-	TIM5_CH3	-	-	-	-	-	-	-	-	-	FMC_D20	DCMI_D3	LCD_R6	EVENTOUT
	PH13	-	-	-	TIM8_CH1N	-	-	-	-	-	CAN1_TX	-	-	FMC_D21	-	LCD_G2	EVENTOUT
	PH14	-	-	-	TIM8_CH2N	-	-	-	-	-	-	-	-	FMC_D22	DCMI_D4	LCD_G3	EVENTOUT
	PH15	-	-	-	TIM8_CH3N	-	-	-	-	-	-	-	-	FMC_D23	DCMI_D11	LCD_G4	EVENTOUT
Port I	PI0	-	-	TIM5_CH4	-	-	SPI2_NSS/I2S2_WS	-	-	-	-	-	-	FMC_D24	DCMI_D13	LCD_G5	EVENTOUT
	PI1	-	-	-	-	-	SPI2_SCK/I2S2_CK	-	-	-	-	-	-	FMC_D25	DCMI_D8	LCD_G6	EVENTOUT
	PI2	-	-	-	TIM8_CH4	-	SPI2_MISO	I2S2ext_SD	-	-	-	-	-	FMC_D26	DCMI_D9	LCD_G7	EVENTOUT
	PI3	-	-	-	TIM8_ETR	-	SPI2_MOSI/I2S2_SD							FMC_D27	DCMI_D10		EVENTOUT
	PI4	-	-	-	TIM8_BKIN	-	-	-	-	-	-	-	-	FMC_NBL2	DCMI_D5	LCD_B4	EVENTOUT
	PI5	-	-	-	TIM8_CH1	-	-	-	-	-	-	-	-	FMC_NBL3	DCMI_VSYNC	LCD_B5	EVENTOUT
	PI6	-	-	-	TIM8_CH2	-	-	-	-	-	-	-	-	FMC_D28	DCMI_D6	LCD_B6	EVENTOUT

Table 12. STM32F427xx and STM32F429xx alternate function mapping (continued)

Port		AF0	AF1	AF2	AF3	AF4	AF5	AF6	AF7	AF8	AF9	AF10	AF11	AF12	AF13	AF14	AF15
		SYS	TIM1/2	TIM3/4/5	TIM8/9/10/11	I2C1/2/3	SPI1/2/3/4/5/6	SPI2/3/SAI1	SPI3/USART1/2/3	USART6/UART4/5/7/8	CAN1/2/TIM12/13/14/LCD	OTG2_HS/OTG1_FS	ETH	FMC/SDIO/OTG2_FS	DCMI	LCD	SYS
Port I	PI7	-	-	-	TIM8_CH3	-	-	-	-	-	-	-	-	FMC_D29	DCMI_D7	LCD_B7	EVENTOUT
	PI8	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	EVENTOUT
	PI9	-	-	-	-	-	-	-	-	-	CAN1_RX	-	-	FMC_D30	-	LCD_VSYNC	EVENTOUT
	PI10	-	-	-	-	-	-	-	-	-	-	-	ETH_MII_RX_ER	FMC_D31	-	LCD_HSYNC	EVENTOUT
	PI11	-	-	-	-	-	-	-	-	-	-	OTG_HS_ULPI_DIR	-	-	-	-	EVENTOUT
	PI12	-	-	-	-	-	-	-	-	-	-	-	-	-	-	LCD_HSYNC	EVENTOUT
	PI13	-	-	-	-	-	-	-	-	-	-	-	-	-	-	LCD_VSYNC	EVENTOUT
	PI14	-	-	-	-	-	-	-	-	-	-	-	-	-	-	LCD_CLK	EVENTOUT
	PI15	-	-	-	-	-	-	-	-	-	-	-	-	-	-	LCD_R0	EVENTOUT
Port J	PJ0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	LCD_R1	EVENTOUT
	PJ1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	LCD_R2	EVENTOUT
	PJ2	-	-	-	-	-	-	-	-	-	-	-	-	-	-	LCD_R3	EVENTOUT
	PJ3	-	-	-	-	-	-	-	-	-	-	-	-	-	-	LCD_R4	EVENTOUT
	PJ4	-	-	-	-	-	-	-	-	-	-	-	-	-	-	LCD_R5	EVENTOUT
	PJ5	-	-	-	-	-	-	-	-	-	-	-	-	-	-	LCD_R6	EVENTOUT
	PJ6	-	-	-	-	-	-	-	-	-	-	-	-	-	-	LCD_R7	EVENTOUT
	PJ7	-	-	-	-	-	-	-	-	-	-	-	-	-	-	LCD_G0	EVENTOUT

Table 12. STM32F427xx and STM32F429xx alternate function mapping (continued)

Port		AF0	AF1	AF2	AF3	AF4	AF5	AF6	AF7	AF8	AF9	AF10	AF11	AF12	AF13	AF14	AF15
		SYS	TIM1/2	TIM3/4/5	TIM8/9/ 10/11	I2C1/ 2/3	SPI1/2/ 3/4/5/6	SPI2/3/ SAI1	SPI3/ USART1/ 2/3	USART6/ UART4/5/7 /8	CAN1/2/ TIM12/13/14 /LCD	OTG2_HS /OTG1_ FS	ETH	FMC/SDIO /OTG2_FS	DCMI	LCD	SYS
Port J	PJ8	-	-	-	-	-	-	-	-	-	-	-	-	-	-	LCD_G1	EVEN TOUT
	PJ9	-	-	-	-	-	-	-	-	-	-	-	-	-	-	LCD_G2	EVEN TOUT
	PJ10	-	-	-	-	-	-	-	-	-	-	-	-	-	-	LCD_G3	EVEN TOUT
	PJ11	-	-	-	-	-	-	-	-	-	-	-	-	-	-	LCD_G4	EVEN TOUT
	PJ12	-	-	-	-	-	-	-	-	-	-	-	-	-	-	LCD_B0	EVEN TOUT
	PJ13	-	-	-	-	-	-	-	-	-	-	-	-	-	-	LCD_B1	EVEN TOUT
	PJ14	-	-	-	-	-	-	-	-	-	-	-	-	-	-	LCD_B2	EVEN TOUT
	PJ15	-	-	-	-	-	-	-	-	-	-	-	-	-	-	LCD_B3	EVEN TOUT
Port K	PK0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	LCD_G5	EVEN TOUT
	PK1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	LCD_G6	EVEN TOUT
	PK2	-	-	-	-	-	-	-	-	-	-	-	-	-	-	LCD_G7	EVEN TOUT
	PK3	-	-	-	-	-	-	-	-	-	-	-	-	-	-	LCD_B4	EVEN TOUT
	PK4	-	-	-	-	-	-	-	-	-	-	-	-	-	-	LCD_B5	EVEN TOUT
	PK5	-	-	-	-	-	-	-	-	-	-	-	-	-	-	LCD_B6	EVEN TOUT
	PK6	-	-	-	-	-	-	-	-	-	-	-	-	-	-	LCD_B7	EVEN TOUT
	PK7	-	-	-	-	-	-	-	-	-	-	-	-	-	-	LCD_DE	EVEN TOUT

1. The DCMI_VSYNC alternate function on PG9 is only available on silicon revision 3.

5 Memory mapping

The memory map is shown in *Figure 19*.

Figure 19. Memory map

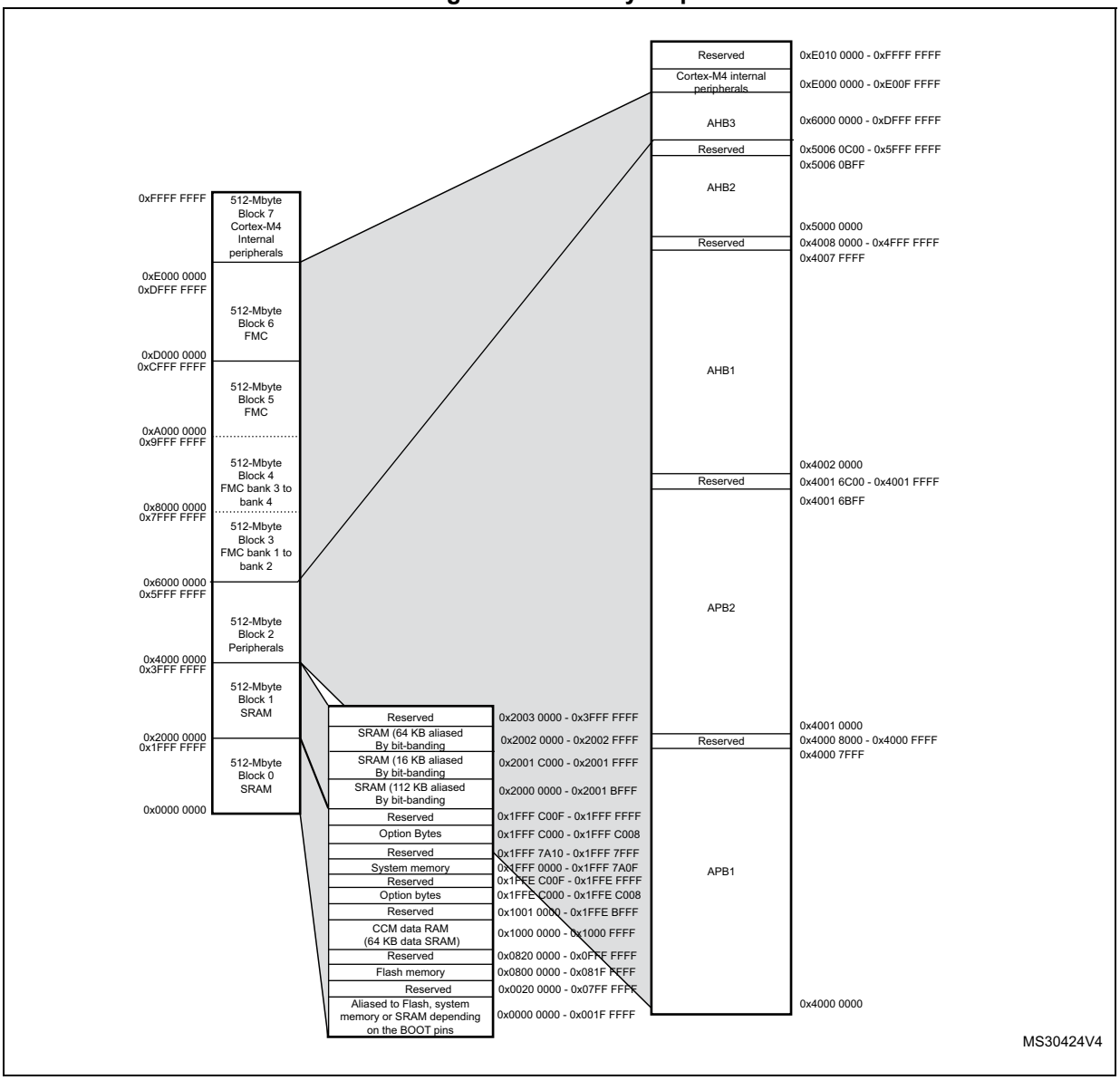


Table 13. STM32F427xx and STM32F429xx register boundary addresses

Bus	Boundary address	Peripheral
	0xE00F FFFF - 0xFFFF FFFF	Reserved
Cortex-M4	0xE000 0000 - 0xE00F FFFF	Cortex-M4 internal peripherals
AHB3	0xD000 0000 - 0xDFFF FFFF	FMC bank 6
	0xC000 0000 - 0xCFFF FFFF	FMC bank 5
	0xA000 1000 - 0xBFFF FFFF	Reserved
	0xA000 0000- 0xA000 0FFF	FMC control register
	0x9000 0000 - 0x9FFF FFFF	FMC bank 4
	0x8000 0000 - 0x8FFF FFFF	FMC bank 3
	0x7000 0000 - 0x7FFF FFFF	FMC bank 2
	0x6000 0000 - 0x6FFF FFFF	FMC bank 1
	0x5006 0C00- 0x5FFF FFFF	Reserved
AHB2	0x5006 0800 - 0X5006 0BFF	RNG
	0x5005 0400 - X5006 07FF	Reserved
	0x5005 0000 - 0X5005 03FF	DCMI
	0x5004 0000- 0x5004 FFFF	Reserved
	0x5000 0000 - 0X5003 FFFF	USB OTG FS

Table 13. STM32F427xx and STM32F429xx register boundary addresses (continued)

Bus	Boundary address	Peripheral
	0x4008 0000- 0x4FFF FFFF	Reserved
AHB1	0x4004 0000 - 0x4007 FFFF	USB OTG HS
	0x4002 BC00- 0x4003 FFFF	Reserved
	0x4002 B000 - 0x4002 BBFF	DMA2D
	0x4002 9400 - 0x4002 AFFF	Reserved
	0x4002 9000 - 0x4002 93FF	ETHERNET MAC
	0x4002 8C00 - 0x4002 8FFF	
	0x4002 8800 - 0x4002 8BFF	
	0x4002 8400 - 0x4002 87FF	
	0x4002 8000 - 0x4002 83FF	
	0x4002 6800 - 0x4002 7FFF	Reserved
	0x4002 6400 - 0x4002 67FF	DMA2
	0x4002 6000 - 0x4002 63FF	DMA1
	0X4002 5000 - 0X4002 5FFF	Reserved
	0x4002 4000 - 0x4002 4FFF	BKPSRAM
	0x4002 3C00 - 0x4002 3FFF	Flash interface register
	0x4002 3800 - 0x4002 3BFF	RCC
	0X4002 3400 - 0X4002 37FF	Reserved
	0x4002 3000 - 0x4002 33FF	CRC
	0x4002 2C00 - 0x4002 2FFF	Reserved
	0x4002 2800 - 0x4002 2BFF	GPIOK
	0x4002 2400 - 0x4002 27FF	GPIOJ
	0x4002 2000 - 0x4002 23FF	GPIOI
	0x4002 1C00 - 0x4002 1FFF	GPIOH
	0x4002 1800 - 0x4002 1BFF	GPIOG
	0x4002 1400 - 0x4002 17FF	GPIOF
	0x4002 1000 - 0x4002 13FF	GPIOE
	0X4002 0C00 - 0x4002 0FFF	GPIOD
	0x4002 0800 - 0x4002 0BFF	GPIOC
	0x4002 0400 - 0x4002 07FF	GPIOB
	0x4002 0000 - 0x4002 03FF	GPIOA

Table 13. STM32F427xx and STM32F429xx register boundary addresses (continued)

Bus	Boundary address	Peripheral
	0x4001 6C00- 0x4001 FFFF	Reserved
APB2	0x4001 6800 - 0x4001 6BFF	LCD-TFT
	0x4001 5C00 - 0x4001 67FF	Reserved
	0x4001 5800 - 0x4001 5BFF	SAI1
	0x4001 5400 - 0x4001 57FF	SPI6
	0x4001 5000 - 0x4001 53FF	SPI5
	0x4001 5400 - 0x4001 57FF	SPI6
	0x4001 5000 - 0x4001 53FF	SPI5
	0x4001 4C00 - 0x4001 4FFF	Reserved
	0x4001 4800 - 0x4001 4BFF	TIM11
	0x4001 4400 - 0x4001 47FF	TIM10
	0x4001 4000 - 0x4001 43FF	TIM9
	0x4001 3C00 - 0x4001 3FFF	EXTI
	0x4001 3800 - 0x4001 3BFF	SYSCFG
	0x4001 3400 - 0x4001 37FF	SPI4
	0x4001 3000 - 0x4001 33FF	SPI1
	0x4001 2C00 - 0x4001 2FFF	SDIO
	0x4001 2400 - 0x4001 2BFF	Reserved
	0x4001 2000 - 0x4001 23FF	ADC1 - ADC2 - ADC3
	0x4001 1800 - 0x4001 1FFF	Reserved
	0x4001 1400 - 0x4001 17FF	USART6
	0x4001 1000 - 0x4001 13FF	USART1
	0x4001 0800 - 0x4001 0FFF	Reserved
	0x4001 0400 - 0x4001 07FF	TIM8
	0x4001 0000 - 0x4001 03FF	TIM1

Table 13. STM32F427xx and STM32F429xx register boundary addresses (continued)

Bus	Boundary address	Peripheral
	0x4000 8000- 0x4000 FFFF	Reserved
APB1	0x4000 7C00 - 0x4000 7FFF	UART8
	0x4000 7800 - 0x4000 7BFF	UART7
	0x4000 7400 - 0x4000 77FF	DAC
	0x4000 7000 - 0x4000 73FF	PWR
	0x4000 6C00 - 0x4000 6FFF	Reserved
	0x4000 6800 - 0x4000 6BFF	CAN2
	0x4000 6400 - 0x4000 67FF	CAN1
	0x4000 6000 - 0x4000 63FF	Reserved
	0x4000 5C00 - 0x4000 5FFF	I2C3
	0x4000 5800 - 0x4000 5BFF	I2C2
	0x4000 5400 - 0x4000 57FF	I2C1
	0x4000 5000 - 0x4000 53FF	UART5
	0x4000 4C00 - 0x4000 4FFF	UART4
	0x4000 4800 - 0x4000 4BFF	USART3
	0x4000 4400 - 0x4000 47FF	USART2
	0x4000 4000 - 0x4000 43FF	I2S3ext
	0x4000 3C00 - 0x4000 3FFF	SPI3 / I2S3
	0x4000 3800 - 0x4000 3BFF	SPI2 / I2S2
	0x4000 3400 - 0x4000 37FF	I2S2ext
	0x4000 3000 - 0x4000 33FF	IWDG
	0x4000 2C00 - 0x4000 2FFF	WWDG
	0x4000 2800 - 0x4000 2BFF	RTC & BKP Registers
	0x4000 2400 - 0x4000 27FF	Reserved
	0x4000 2000 - 0x4000 23FF	TIM14
	0x4000 1C00 - 0x4000 1FFF	TIM13
	0x4000 1800 - 0x4000 1BFF	TIM12
	0x4000 1400 - 0x4000 17FF	TIM7
	0x4000 1000 - 0x4000 13FF	TIM6
	0x4000 0C00 - 0x4000 0FFF	TIM5
	0x4000 0800 - 0x4000 0BFF	TIM4
	0x4000 0400 - 0x4000 07FF	TIM3
	0x4000 0000 - 0x4000 03FF	TIM2

6 Electrical characteristics

6.1 Parameter conditions

Unless otherwise specified, all voltages are referenced to V_{SS}.

6.1.1 Minimum and maximum values

Unless otherwise specified the minimum and maximum values are guaranteed in the worst conditions of ambient temperature, supply voltage and frequencies by tests in production on 100% of the devices with an ambient temperature at T_A = 25 °C and T_A = $T_A$max (given by the selected temperature range).

Data based on characterization results, design simulation and/or technology characteristics are indicated in the table footnotes and are not tested in production. Based on characterization, the minimum and maximum values refer to sample tests and represent the mean value plus or minus three times the standard deviation (mean±3σ).

6.1.2 Typical values

Unless otherwise specified, typical data are based on T_A = 25 °C, V_{DD} = 3.3 V (for the 1.7 V ≤V_{DD} ≤3.6 V voltage range). They are given only as design guidelines and are not tested.

Typical ADC accuracy values are determined by characterization of a batch of samples from a standard diffusion lot over the full temperature range, where 95% of the devices have an error less than or equal to the value indicated (mean±2σ).

6.1.3 Typical curves

Unless otherwise specified, all typical curves are given only as design guidelines and are not tested.

6.1.4 Loading capacitor

The loading conditions used for pin parameter measurement are shown in *Figure 20*.

6.1.5 Pin input voltage

The input voltage measurement on a pin of the device is described in *Figure 21*.

Figure 20. Pin loading conditions

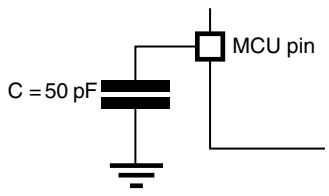
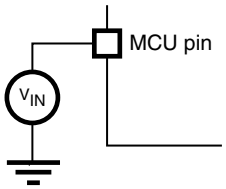
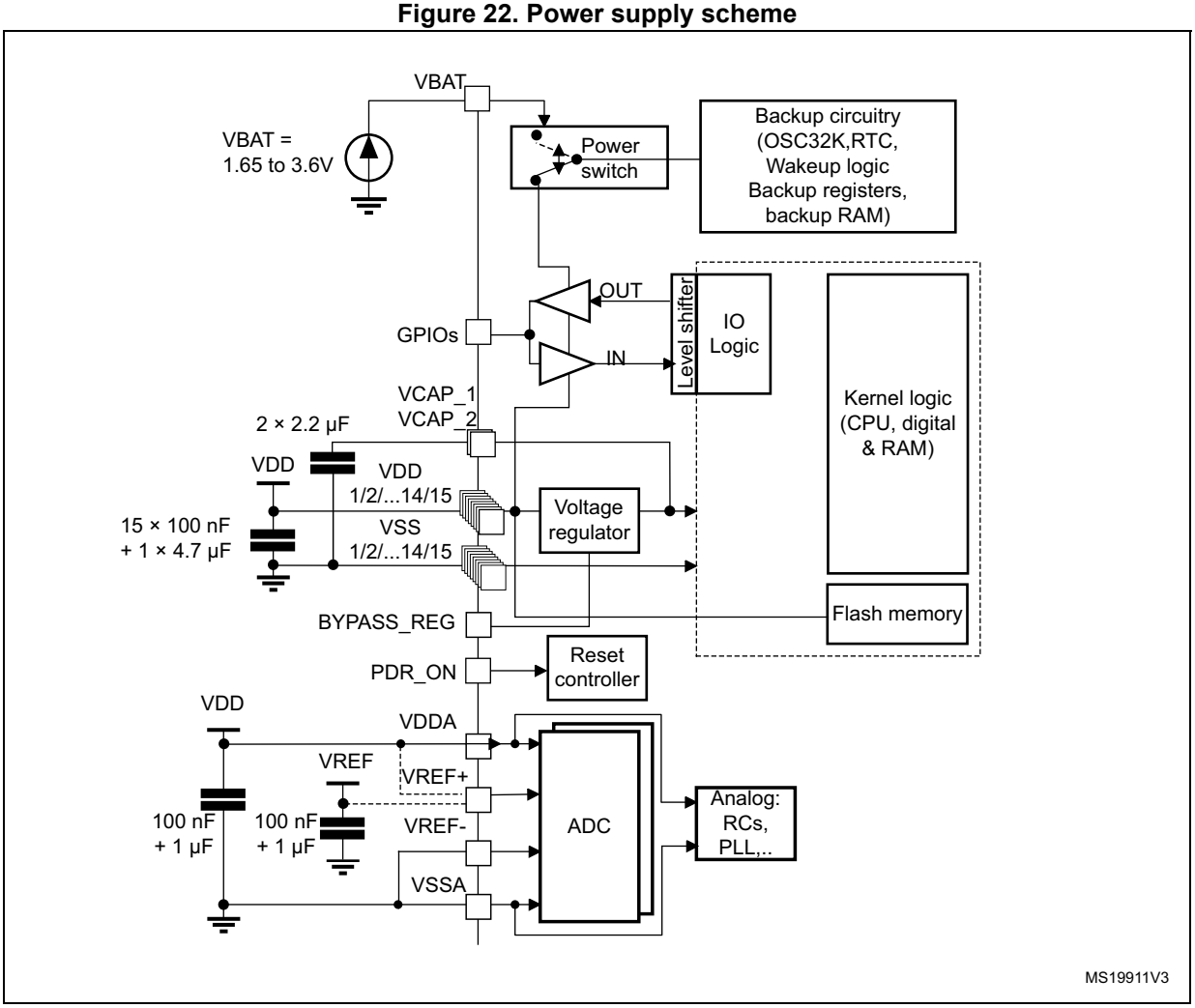


Figure 21. Pin input voltage



6.1.6 Power supply scheme

Figure 22. Power supply scheme

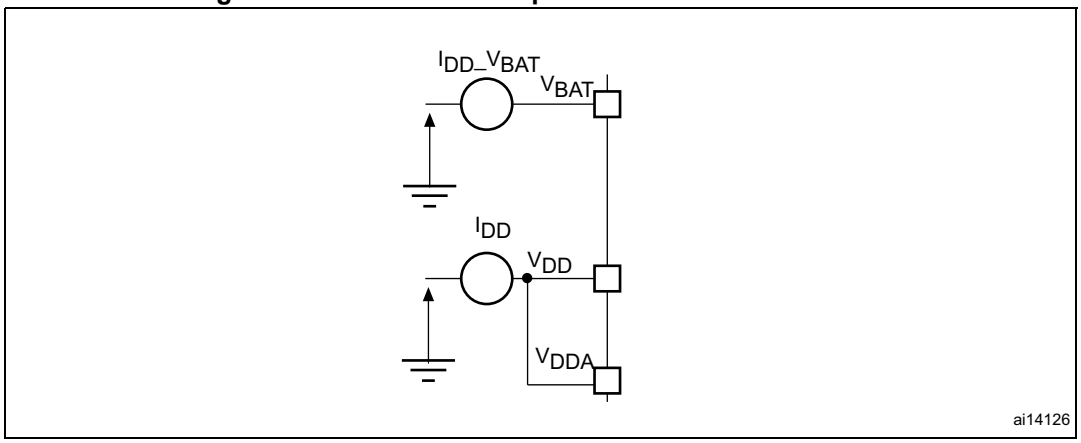


1. To connect BYPASS_REG and PDR_ON pins, refer to *Section 3.17: Power supply supervisor* and *Section 3.18: Voltage regulator*
2. The two 2.2 µF ceramic capacitors should be replaced by two 100 nF decoupling capacitors when the voltage regulator is OFF.
3. The 4.7 µF ceramic capacitor must be connected to one of the V_{DD} pin.
4. V_{DDA}=V_{DD} and V_{SSA}=V_{SS}.

Caution: Each power supply pair (V_{DD}/V_{SS}, V_{DDA}/V_{SSA} ...) must be decoupled with filtering ceramic capacitors as shown above. These capacitors must be placed as close as possible to, or below, the appropriate pins on the underside of the PCB to ensure good operation of the device. It is not recommended to remove filtering capacitors to reduce PCB size or cost. This might cause incorrect operation of the device.

6.1.7 Current consumption measurement

Figure 23. Current consumption measurement scheme



6.2 Absolute maximum ratings

Stresses above the absolute maximum ratings listed in *Table 14: Voltage characteristics*, *Table 15: Current characteristics*, and *Table 16: Thermal characteristics* may cause permanent damage to the device. These are stress ratings only and functional operation of the device at these conditions is not implied. Exposure to maximum rating conditions for extended periods may affect device reliability.

Device mission profile (application conditions) is compliant with JEDEC JESD47 Qualification Standard, extended mission profiles are available on demand.

Table 14. Voltage characteristics

Symbol	Ratings	Min	Max	Unit
V_{DD}–V_{SS}	External main supply voltage (including V_{DDA}, V_{DD} and VBAT)(1)	− 0.3	4.0	V
V_{IN}	Input voltage on FT pins(2)	V_{SS} − 0.3	V_{DD}+4.0	
	Input voltage on TTa pins	V_{SS} − 0.3	4.0	
	Input voltage on any other pin	V_{SS} − 0.3	4.0	
	Input voltage on BOOT0 pin	V_{SS}	9.0	
$\lvert\Delta V_{DDx}\rvert$	Variations between different V_{DD} power pins	-	50	mV
$\lvert V_{SSX} - V_{SS}\rvert$	Variations between all the different ground pins including V_{REF-}	-	50	
$V_{ESD(HBM)}$	Electrostatic discharge voltage (human body model)	see *Section 6.3.15: Absolute maximum ratings (electrical sensitivity)*		

1. All main power (V_{DD}, V_{DDA}) and ground (V_{SS}, V_{SSA}) pins must always be connected to the external power supply, in the permitted range.
2. V_{IN} maximum value must always be respected. Refer to *Table 15* for the values of the maximum allowed injected current.

Table 15. Current characteristics

Symbol	Ratings	Max.	Unit
ΣI_{VDD}	Total current into sum of all V_{DD_x} power lines (source)(1)	270	mA
ΣI_{VSS}	Total current out of sum of all V_{SS_x} ground lines (sink)(1)	− 270	
I_{VDD}	Maximum current into each V_{DD_x} power line (source)(1)	100	
I_{VSS}	Maximum current out of each V_{SS_x} ground line (sink)(1)	− 100	
I_{IO}	Output current sunk by any I/O and control pin	25	
	Output current sourced by any I/Os and control pin	− 25	
ΣI_{IO}	Total output current sunk by sum of all I/O and control pins (2)	120	
	Total output current sourced by sum of all I/Os and control pins(2)	− 120	
$I_{INJ(PIN)}$ (3)	Injected current on FT pins (4)	− 5/+0	
	Injected current on NRST and BOOT0 pins (4)		
	Injected current on TTa pins(5)	±5	
$\Sigma I_{INJ(PIN)}$(5)	Total injected current (sum of all I/O and control pins)(6)	±25	

1. All main power (V_{DD}, V_{DDA}) and ground (V_{SS}, V_{SSA}) pins must always be connected to the external power supply, in the permitted range.
2. This current consumption must be correctly distributed over all I/Os and control pins. The total output current must not be sunk/sourced between two consecutive power supply pins referring to high pin count LQFP packages.
3. Negative injection disturbs the analog performance of the device. See note in *Section 6.3.21: 12-bit ADC characteristics*.
4. Positive injection is not possible on these I/Os and does not occur for input voltages lower than the specified maximum value.
5. A positive injection is induced by $V_{IN}>V_{DDA}$ while a negative injection is induced by $V_{IN}<V_{SS}$. $I_{INJ(PIN)}$ must never be exceeded. Refer to *Table 14* for the values of the maximum allowed input voltage.
6. When several inputs are submitted to a current injection, the maximum $\Sigma I_{INJ(PIN)}$ is the absolute sum of the positive and negative injected currents (instantaneous values).

Table 16. Thermal characteristics

Symbol	Ratings	Value	Unit
T_{STG}	Storage temperature range	− 65 to +150	°C
T_J	Maximum junction temperature	125	°C

6.3 Operating conditions

6.3.1 General operating conditions

Table 17. General operating conditions

Symbol	Parameter	Conditions[1]		Min	Typ	Max	Unit
f_{HCLK}	Internal AHB clock frequency	Power Scale 3 (VOS[1:0] bits in PWR_CR register = 0x01), Regulator ON, over-drive OFF		0	-	120	MHz
		Power Scale 2 (VOS[1:0] bits in PWR_CR register = 0x10), Regulator ON	Over-drive OFF	0	-	144	
			Over-drive ON		-	168	
		Power Scale 1 (VOS[1:0] bits in PWR_CR register= 0x11), Regulator ON	Over-drive OFF	0	-	168	
			Over-drive ON		-	180	
f_{PCLK1}	Internal APB1 clock frequency	Over-drive OFF		0	-	42	
		Over-drive ON		0	-	45	
f_{PCLK2}	Internal APB2 clock frequency	Over-drive OFF		0	-	84	
		Over-drive ON		0	-	90	
V_{DD}	Standard operating voltage			1.7[2]	-	3.6	V
V_{DDA} [3][4]	Analog operating voltage (ADC limited to 1.2 M samples)	Must be the same potential as V_{DD}[5]		1.7[2]	-	2.4	
	Analog operating voltage (ADC limited to 2.4 M samples)			2.4	-	3.6	
V_{BAT}	Backup operating voltage			1.65	-	3.6	
V_{12}	Regulator ON: 1.2 V internal voltage on V_{CAP_1}/V_{CAP_2} pins	Power Scale 3 ((VOS[1:0] bits in PWR_CR register = 0x01), 120 MHz HCLK max frequency		1.08	1.14	1.20	V
		Power Scale 2 ((VOS[1:0] bits in PWR_CR register = 0x10), 144 MHz HCLK max frequency with over-drive OFF or 168 MHz with over-drive ON		1.20	1.26	1.32	
		Power Scale 1 ((VOS[1:0] bits in PWR_CR register = 0x11), 168 MHz HCLK max frequency with over-drive OFF or 180 MHz with over-drive ON		1.26	1.32	1.40	
	Regulator OFF: 1.2 V external voltage must be supplied from external regulator on V_{CAP_1}/V_{CAP_2} pins[6]	Max frequency 120 MHz		1.10	1.14	1.20	
		Max frequency 144 MHz		1.20	1.26	1.32	
		Max frequency 168 MHz		1.26	1.32	1.38	

Table 17. General operating conditions (continued)

Symbol	Parameter	Conditions[1]	Min	Typ	Max	Unit
V_{IN}	Input voltage on RST and FT pins[7]	2 V ≤ V_{DD} ≤ 3.6 V	− 0.3	-	5.5	V
		V_{DD} ≤ 2 V	− 0.3	-	5.2	
	Input voltage on TTa pins		− 0.3	-	V_{DDA}+ 0.3	
	Input voltage on BOOT0 pin		0	-	9	
P_D	Power dissipation at T_A = 85 °C for suffix 6 or T_A = 105 °C for suffix 7[8]	LQFP100	-	-	465	mW
		WLCSP143	-	-	641	
		LQFP144	-	-	500	
		UFBGA169	-	-	385	
		LQFP176	-	-	526	
		UFBGA176	-	-	513	
		LQFP208	-	-	1053	
		TFBGA216	-	-	690	
T_A	Ambient temperature for 6 suffix version	Maximum power dissipation	− 40		85	°C
		Low power dissipation[9]	− 40		105	
	Ambient temperature for 7 suffix version	Maximum power dissipation	− 40		105	°C
		Low power dissipation[9]	− 40		125	
T_J	Junction temperature range	6 suffix version	− 40		105	°C
		7 suffix version	− 40		125	

1. The over-drive mode is not supported at the voltage ranges from 1.7 to 2.1 V.
2. V_{DD}/V_{DDA} minimum value of 1.7 V is obtained with the use of an external power supply supervisor (refer to *Section 3.17.2: Internal reset OFF*).
3. When the ADC is used, refer to *Table 74: ADC characteristics*.
4. If V_{REF+} pin is present, it must respect the following condition: V_{DDA}-V_{REF+} < 1.2 V.
5. It is recommended to power V_{DD} and V_{DDA} from the same source. A maximum difference of 300 mV between V_{DD} and V_{DDA} can be tolerated during power-up and power-down operation.
6. The over-drive mode is not supported when the internal regulator is OFF.
7. To sustain a voltage higher than VDD+0.3, the internal Pull-up and Pull-Down resistors must be disabled
8. If T_A is lower, higher P_D values are allowed as long as T_J does not exceed T_{Jmax}.
9. In low power dissipation state, T_A can be extended to this range as long as T_J does not exceed T_{Jmax}.

Table 18. Limitations depending on the operating power supply range

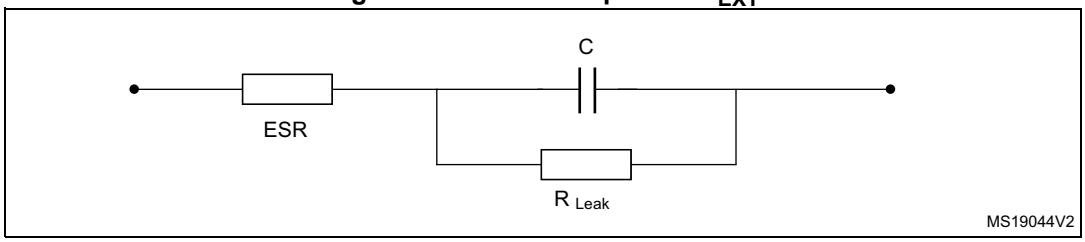
Operating power supply range	ADC operation	Maximum Flash memory access frequency with no wait states ($f_{Flashmax}$)	Maximum HCLK frequency vs Flash memory wait states (1)(2)	I/O operation	Possible Flash memory operations
V_{DD} =1.7 to 2.1 V(3)	Conversion time up to 1.2 Msps	20 MHz(4)	168 MHz with 8 wait states and over-drive OFF	No I/O compensation	8-bit erase and program operations only
V_{DD} = 2.1 to 2.4 V	Conversion time up to 1.2 Msps	22 MHz	180 MHz with 8 wait states and over-drive ON	No I/O compensation	16-bit erase and program operations
V_{DD} = 2.4 to 2.7 V	Conversion time up to 2.4 Msps	24 MHz	180 MHz with 7 wait states and over-drive ON	I/O compensation works	16-bit erase and program operations
V_{DD} = 2.7 to 3.6 V(5)	Conversion time up to 2.4 Msps	30 MHz	180 MHz with 5 wait states and over-drive ON	I/O compensation works	32-bit erase and program operations

1. Applicable only when the code is executed from Flash memory. When the code is executed from RAM, no wait state is required.
2. Thanks to the ART accelerator and the 128-bit Flash memory, the number of wait states given here does not impact the execution speed from Flash memory since the ART accelerator allows to achieve a performance equivalent to 0 wait state program execution.
3. V_{DD}/V_{DDA} minimum value of 1.7 V is obtained with the use of an external power supply supervisor (refer to *Section 3.17.2: Internal reset OFF*).
4. Prefetch is not available.
5. The voltage range for USB full speed PHYs can drop down to 2.7 V. However the electrical characteristics of D- and D+ pins will be degraded between 2.7 and 3 V.

6.3.2 VCAP1/VCAP2 external capacitor

Stabilization for the main regulator is achieved by connecting an external capacitor C_{EXT} to the VCAP1/VCAP2 pins. C_{EXT} is specified in *Table 19*.

Figure 24. External capacitor C_{EXT}



1. Legend: ESR is the equivalent series resistance.

Table 19. VCAP1/VCAP2 operating conditions(1)

Symbol	Parameter	Conditions
CEXT	Capacitance of external capacitor	2.2 µF
ESR	ESR of external capacitor	< 2 Ω

1. When bypassing the voltage regulator, the two 2.2 µF V_{CAP} capacitors are not required and should be replaced by two 100 nF decoupling capacitors.

6.3.3 Operating conditions at power-up / power-down (regulator ON)

Subject to general operating conditions for T_A.

Table 20. Operating conditions at power-up / power-down (regulator ON)

Symbol	Parameter	Min	Max	Unit
t_{VDD}	V_{DD} rise time rate	20	∞	µs/V
	V_{DD} fall time rate	20	∞	

6.3.4 Operating conditions at power-up / power-down (regulator OFF)

Subject to general operating conditions for T_A.

Table 21. Operating conditions at power-up / power-down (regulator OFF)(1)

Symbol	Parameter	Conditions	Min	Max	Unit
t_{VDD}	V_{DD} rise time rate	Power-up	20	∞	µs/V
	V_{DD} fall time rate	Power-down	20	∞	
t_{VCAP}	V_{CAP_1} and V_{CAP_2} rise time rate	Power-up	20	∞	
	V_{CAP_1} and V_{CAP_2} fall time rate	Power-down	20	∞	

1. To reset the internal logic at power-down, a reset must be applied on pin PA0 when V_{DD} reach below 1.08 V.

6.3.5 Reset and power control block characteristics

The parameters given in *Table 22* are derived from tests performed under ambient temperature and V_{DD} supply voltage conditions summarized in *Table 17*.

Table 22. reset and power control block characteristics

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
V_{PVD}	Programmable voltage detector level selection	PLS[2:0]=000 (rising edge)	2.09	2.14	2.19	V
		PLS[2:0]=000 (falling edge)	1.98	2.04	2.08	V
		PLS[2:0]=001 (rising edge)	2.23	2.30	2.37	V
		PLS[2:0]=001 (falling edge)	2.13	2.19	2.25	V
		PLS[2:0]=010 (rising edge)	2.39	2.45	2.51	V
		PLS[2:0]=010 (falling edge)	2.29	2.35	2.39	V
		PLS[2:0]=011 (rising edge)	2.54	2.60	2.65	V
		PLS[2:0]=011 (falling edge)	2.44	2.51	2.56	V
		PLS[2:0]=100 (rising edge)	2.70	2.76	2.82	V
		PLS[2:0]=100 (falling edge)	2.59	2.66	2.71	V
		PLS[2:0]=101 (rising edge)	2.86	2.93	2.99	V
		PLS[2:0]=101 (falling edge)	2.65	2.84	2.92	V
		PLS[2:0]=110 (rising edge)	2.96	3.03	3.10	V
		PLS[2:0]=110 (falling edge)	2.85	2.93	2.99	V
		PLS[2:0]=111 (rising edge)	3.07	3.14	3.21	V
		PLS[2:0]=111 (falling edge)	2.95	3.03	3.09	V
$V_{PVDhyst}$(1)	PVD hysteresis		-	100	-	mV
$V_{POR/PDR}$	Power-on/power-down reset threshold	Falling edge	1.60	1.68	1.76	V
		Rising edge	1.64	1.72	1.80	V
$V_{PDRhyst}$(1)	PDR hysteresis		-	40	-	mV
V_{BOR1}	Brownout level 1 threshold	Falling edge	2.13	2.19	2.24	V
		Rising edge	2.23	2.29	2.33	V
V_{BOR2}	Brownout level 2 threshold	Falling edge	2.44	2.50	2.56	V
		Rising edge	2.53	2.59	2.63	V
V_{BOR3}	Brownout level 3 threshold	Falling edge	2.75	2.83	2.88	V
		Rising edge	2.85	2.92	2.97	V
$V_{BORhyst}$(1)	BOR hysteresis		-	100	-	mV
$T_{RSTTEMPO}$(1)(2)	POR reset temporization		0.5	1.5	3.0	ms

Table 22. reset and power control block characteristics (continued)

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
I_{RUSH}(1)	InRush current on voltage regulator power-on (POR or wakeup from Standby)		-	160	200	mA
E_{RUSH}(1)	InRush energy on voltage regulator power-on (POR or wakeup from Standby)	V_{DD} = 1.7 V, T_A = 105 °C, I_{RUSH} = 171 mA for 31 µs	-	-	5.4	µC

1. Guaranteed by design.
2. The reset temporization is measured from the power-on (POR reset or wakeup from V_{BAT}) to the instant when first instruction is read by the user application code.

6.3.6 Over-drive switching characteristics

When the over-drive mode switches from enabled to disabled or disabled to enabled, the system clock is stalled during the internal voltage set-up.

The over-drive switching characteristics are given in *Table 23*. They are sbject to general operating conditions for T_A.

Table 23. Over-drive switching characteristics(1)

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
Tod_swen	Over_drive switch enable time	HSI	-	45	-	µs
		HSE max for 4 MHz and min for 26 MHz	45	-	100	
		External HSE 50 MHz	-	40	-	
Tod_swdis	Over_drive switch disable time	HSI	-	20	-	
		HSE max for 4 MHz and min for 26 MHz.	20	-	80	
		External HSE 50 MHz	-	15	-	

1. Guaranteed by design.

6.3.7 Supply current characteristics

The current consumption is a function of several parameters and factors such as the operating voltage, ambient temperature, I/O pin loading, device software configuration, operating frequencies, I/O pin switching rate, program location in memory and executed binary code.

The current consumption is measured as described in *Figure 23: Current consumption measurement scheme*.

All the run-mode current consumption measurements given in this section are performed with a reduced code that gives a consumption equivalent to CoreMark code.

Typical and maximum current consumption

The MCU is placed under the following conditions:

- All I/O pins are in input mode with a static value at V_{DD} or V_{SS} (no load).
- All peripherals are disabled except if it is explicitly mentioned.
- The Flash memory access time is adjusted both to f_{HCLK} frequency and V_{DD} range (see *Table 18: Limitations depending on the operating power supply range*).
- Regulator ON
- The voltage scaling and over-drive mode are adjusted to f_{HCLK} frequency as follows:
  - Scale 3 for $f_{HCLK} \leq 120$ MHz
  - Scale 2 for 120 MHz < $f_{HCLK} \leq 144$ MHz
  - Scale 1 for 144 MHz < $f_{HCLK} \leq 180$ MHz. The over-drive is only ON at 180 MHz.
- The system clock is HCLK, $f_{PCLK1} = f_{HCLK}/4$, and $f_{PCLK2} = f_{HCLK}/2$.
- External clock frequency is 4 MHz and PLL is ON when f_{HCLK} is higher than 25 MHz.
- The maximum values are obtained for V_{DD} = 3.6 V and a maximum ambient temperature (T_A), and the typical values for T_A= 25 °C and V_{DD} = 3.3 V unless otherwise specified.

Table 24. Typical and maximum current consumption in Run mode, code with data processing running from Flash memory (ART accelerator enabled except prefetch) or RAM(1)

Symbol	Parameter	Conditions	f_{HCLK} (MHz)	Typ	Max(2)			Unit
					T_A = 25 °C	T_A = 85 °C	T_A = 105 °C	
I_{DD}	Supply current in RUN mode	All Peripherals enabled(3)(4)	180	98	104(5)	123	141(5)	mA
			168	89	98(5)	116	133(5)	
			150	75	84	100	115	
			144	72	81	96	112	
			120	54	58	72	85	
			90	43	45	56	66	
			60	29	30	52	62	
			30	16	20	34	46	
			25	13	16	30	43	
			16	11	13	27	39	
			8	5	9	23	36	
			4	4	8	21	34	
			2	2	7	20	33	
		All Peripherals disabled(3)	180	44	47(5)	69	87(5)	
			168	41	45(5)	66	83(5)	
			150	36	39	57	73	
			144	33	37	56	72	
			120	25	29	43	56	
			90	20	23	41	53	
			60	14	16	34	45	
			30	8	12	26	39	
			25	7	10	24	37	
			16	7	9	22	35	
			8	3	7	21	34	
			4	3	6	20	33	
			2	2	6	20	33	

1. Code and data processing running from SRAM1 using boot pins.
2. Guaranteed by characterization.
3. When analog peripheral blocks such as ADCs, DACs, HSE, LSE, HSI, or LSI are ON, an additional power consumption should be considered.
4. When the ADC is ON (ADON bit set in the ADC_CR2 register), add an additional power consumption of 1.6 mA per ADC for the analog part.
5. Guaranteed by test in production.

Table 25. Typical and maximum current consumption in Run mode, code with data processing running from Flash memory (ART accelerator disabled)

Symbol	Parameter	Conditions	f_{HCLK} (MHz)	Typ	Max(1)			Unit
					TA= 25 °C	TA=85 °C	TA=105 °C	
I_{DD}	Supply current in RUN mode	All Peripherals enabled(2)(3)	180	103	112	140	151	mA
			168	98	107	126	144	
			150	87	95	112	128	
			144	85	92	108	124	
			120	66	71	85	99	
			90	54	58	69	80	
			60	37	39	47	55	
			30	20	24	39	51	
			25	17	21	35	48	
			16	12	16	30	42	
			8	7	11	24	37	
			4	5	8	22	35	
			2	3	7	21	34	
		All Peripherals disabled(3)	180	57	62	87	106	
			168	50	54	76	93	
			150	46	50	70	86	
			144	45	49	68	84	
			120	36	41	56	69	
			90	29	34	46	57	
			60	21	24	33	41	
			30	13	17	31	44	
			25	11	15	28	41	
			16	8	12	25	38	
			8	5	9	23	35	
			4	4	7	21	34	
			2	3	6.5	20	33	

1. Guaranteed by characterization unless otherwise specified.
2. When analog peripheral blocks such as ADCs, DACs, HSE, LSE, HSI, or LSI are ON, an additional power consumption should be considered.
3. When the ADC is ON (ADON bit set in the ADC_CR2 register), add an additional power consumption of 1.6 mA per ADC for the analog part.

Table 26. Typical and maximum current consumption in Sleep mode

Symbol	Parameter	Conditions	f_{HCLK} (MHz)	Typ	Max(1)			Unit
					T_A = 25 °C	T_A = 85 °C	T_A = 105 °C	
I_{DD}	Supply current in Sleep mode	All Peripherals enabled(2)	180	78	89(3)	110	130(3)	mA
			168	66	75(3)	93	110(3)	
			150	56	61	80	96	
			144	54	58	78	94	
			120	40	44	59	72	
			90	32	34	46	56	
			60	22	23	31	45	
			30	10	16	30	43	
			25	9	14	28	40	
			16	5	12	25	40	
			8	3	8	22	35	
			4	3	7	21	34	
			2	2	6.5	20	33	
		All Peripherals disabled	180	21	26(3)	54	76(3)	
			168	16	20(3)	41	58(3)	
			150	14	17	36	52	
			144	13	16.5	35	51	
			120	10	14	28	41	
			90	8	13	26	37	
			60	6	9	24	37	
			30	5	8	22	35	
			25	3	7	21	34	
			16	3	7	21	34	
			8	2	6	20	33	
			4	2	6	20	33	
			2	2	6	20	33	

1. Guaranteed by characterization unless otherwise specified.
2. When analog peripheral blocks such as ADCs, DACs, HSE, LSE, HSI, or LSI are ON, an additional power consumption should be considered.
3. Based on characterization, tested in production.

Table 27. Typical and maximum current consumptions in Stop mode

Symbol	Parameter	Conditions	Typ	Max(1)			Unit
				V_{DD} = 3.6 V			
			T_A = 25 °C	T_A = 25 °C	T_A = 85 °C	T_A = 105 °C	
$I_{DD_STOP_NM}$ (normal mode)	Supply current in Stop mode with voltage regulator in main regulator mode	Flash memory in Stop mode, all oscillators OFF, no independent watchdog	0.40	1.50	14.00	25.00	mA
		Flash memory in Deep power down mode, all oscillators OFF, no independent watchdog	0.35	1.50	14.00	25.00	
	Supply current in Stop mode with voltage regulator in Low Power regulator mode	Flash memory in Stop mode, all oscillators OFF, no independent watchdog	0.29	1.10	10.00	18.00	
		Flash memory in Deep power down mode, all oscillators OFF, no independent watchdog	0.23	1.10	10.00	18.00	
$I_{DD_STOP_UDM}$ (under-drive mode)	Supply current in Stop mode with voltage regulator in main regulator and under-drive mode	Flash memory in Deep power down mode, main regulator in under-drive mode, all oscillators OFF, no independent watchdog	0.19	0.50	6.00	9.00	
	Supply current in Stop mode with voltage regulator in Low Power regulator and under-drive mode	Flash memory in Deep power down mode, Low Power regulator in under-drive mode, all oscillators OFF, no independent watchdog	0.10	0.40	4.00	7.00	

1. Data based on characterization, tested in production.

Table 28. Typical and maximum current consumptions in Standby mode

Symbol	Parameter	Conditions	Typ(1)			Max(2)			Unit
			T_A = 25 °C			T_A = 25 °C	T_A = 85 °C	T_A = 105 °C	
			V_{DD} = 1.7 V	V_{DD}= 2.4 V	V_{DD} = 3.3 V	V_{DD} = 3.6 V			
I_{DD_STBY}	Supply current in Standby mode	Backup SRAM ON, low-speed oscillator (LSE) and RTC ON	2.80	3.00	3.60	7.00	19.00	36.00	µA
		Backup SRAM OFF, low-speed oscillator (LSE) and RTC ON	2.30	2.60	3.10	6.00	16.00	31.00	
		Backup SRAM ON, RTC and LSE OFF	2.30	2.50	2.90	6.00(3)	18.00(3)	35.00(3)	
		Backup SRAM OFF, RTC and LSE OFF	1.70	1.90	2.20	5.00(3)	15.00(3)	30.00(3)	

1. The typical current consumption values are given with PDR OFF (internal reset OFF). When the PDR is OFF (internal reset OFF), the typical current consumption is reduced by additional 1.2 µA.
2. Based on characterization, not tested in production unless otherwise specified.
3. Based on characterization, tested in production.

Table 29. Typical and maximum current consumptions in V_{BAT} mode

Symbol	Parameter	Conditions(1)	Typ			Max(2)		Unit
			T_A = 25 °C			T_A = 85 °C	T_A = 105 °C	
			V_{BAT} = 1.7 V	V_{BAT}= 2.4 V	V_{BAT} = 3.3 V	V_{BAT} = 3.6 V		
I_{DD_VBAT}	Backup domain supply current	Backup SRAM ON, low-speed oscillator (LSE) and RTC ON	1.28	1.40	1.62	6	11	µA
		Backup SRAM OFF, low-speed oscillator (LSE) and RTC ON	0.66	0.76	0.97	3	5	
		Backup SRAM ON, RTC and LSE OFF	0.70	0.72	0.74	5	10	
		Backup SRAM OFF, RTC and LSE OFF	0.10	0.10	0.10	2	4	

1. Crystal used: Abracon ABS07-120-32.768 kHz-T with a C_L of 6 pF for typical values.
2. Guaranteed by characterization results.

Figure 25. Typical V_{BAT} current consumption (LSE and RTC ON/backup RAM OFF)

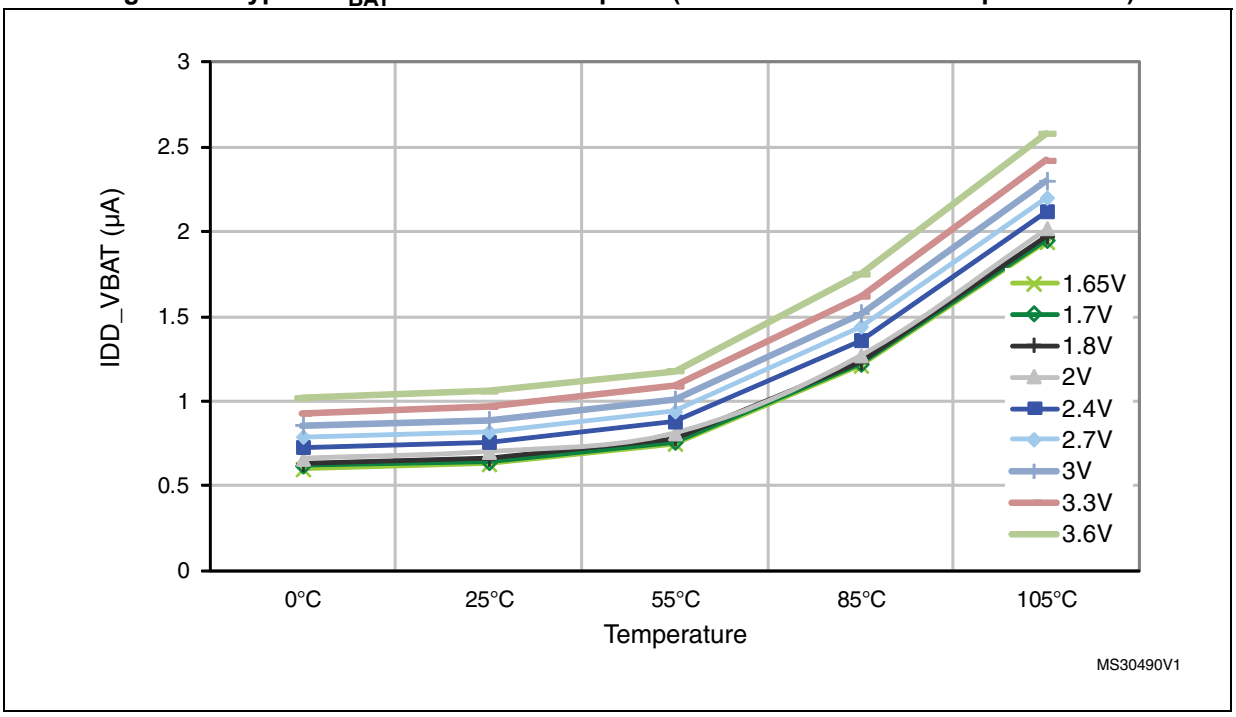
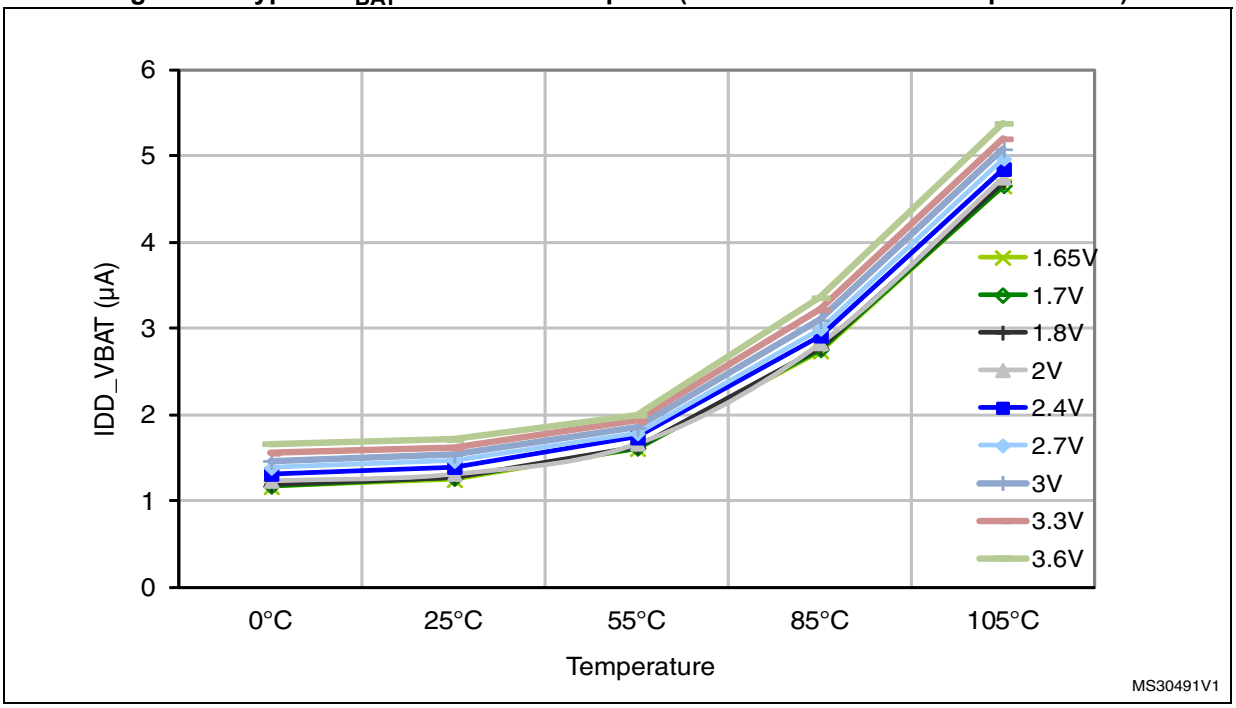


Figure 26. Typical V_{BAT} current consumption (LSE and RTC ON/backup RAM ON)



Additional current consumption

The MCU is placed under the following conditions:

- All I/O pins are configured in analog mode.
- The Flash memory access time is adjusted to fHCLK frequency.
- The voltage scaling is adjusted to fHCLK frequency as follows:
  - Scale 3 for $f_{HCLK} \leq 120$ MHz,
  - Scale 2 for 120 MHz < $f_{HCLK} \leq 144$ MHz
  - Scale 1 for 144 MHz < $f_{HCLK} \leq 180$ MHz. The over-drive is only ON at 180 MHz.
- The system clock is HCLK, $f_{PCLK1} = f_{HCLK}/4$, and $f_{PCLK2} = f_{HCLK}/2$.
- HSE crystal clock frequency is 25 MHz.
- When the regulator is OFF, V12 is provided externally as described in *Table 17: General operating conditions*
- T_A= 25 °C .

Table 30. Typical current consumption in Run mode, code with data processing running from Flash memory or RAM, regulator ON (ART accelerator enabled except prefetch), V_{DD}=1.7 V(1)

Symbol	Parameter	Conditions	f_{HCLK} (MHz)	Typ	Unit
I_{DD}	Supply current in RUN mode from V_{DD} supply	All Peripheral enabled	168	88.2	mA
			150	74.3	
			144	71.3	
			120	52.9	
			90	42.6	
			60	28.6	
			30	15.7	
			25	12.3	
		All Peripheral disabled	168	40.6	
			150	30.6	
			144	32.6	
			120	24.7	
			90	19.7	
			60	13.6	
			30	7.7	
			25	6.7	

1. When peripherals are enabled, the power consumption corresponding to the analog part of the peripherls (such as ADC, or DAC) is not included.

Table 31. Typical current consumption in Run mode, code with data processing running from Flash memory, regulator OFF (ART accelerator enabled except prefetch)(1)

Symbol	Parameter	Conditions	f_{HCLK} (MHz)	VDD=3.3 V		VDD=1.7 V		Unit
				I_{DD12}	I_{DD}	I_{DD12}	I_{DD}	
I_{DD12} / I_{DD}	Supply current in RUN mode from V_{12} and V_{DD} supply	All Peripherals enabled	168	77.8	1.3	76.8	1.0	mA
			150	70.8	1.3	69.8	1.0	
			144	64.5	1.3	63.6	1.0	
			120	49.9	1.2	49.3	0.9	
			90	39.2	1.3	38.7	1.0	
			60	27.2	1.2	26.8	0.9	
			30	15.6	1.2	15.4	0.9	
			25	13.6	1.2	13.5	0.9	
		All Peripherals disabled	168	38.2	1.3	37.0	1.0	
			150	34.6	1.3	33.4	1.0	
			144	31.3	1.3	30.3	1.0	
			120	24.0	1.2	23.2	0.9	
			90	18.1	1.4	18.0	1.0	
			60	12.9	1.2	12.5	0.9	
			30	7.2	1.2	6.9	0.9	
			25	6.3	1.2	6.1	0.9	

1. When peripherals are enabled, the power consumption corresponding to the analog part of the peripherals (such as ADC, or DAC) is not included.

Table 32. Typical current consumption in Sleep mode, regulator ON, V_{DD}=1.7 V(1)

Symbol	Parameter	Conditions	f_{HCLK} (MHz)	Typ	Unit
I_{DD}	Supply current in Sleep mode from V_{DD} supply	All Peripherals enabled	168	65.5	mA
			150	55.5	
			144	53.5	
			120	39.0	
			90	31.6	
			60	21.7	
			30	9.8	
			25	8.8	
		All Peripherals disabled	168	15.7	
			150	13.7	
			144	12.7	
			120	9.7	
			90	7.7	
			60	5.7	
			30	4.7	
			25	2.8	

1. When peripherals are enabled, the power consumption corresponding to the analog part of the peripherals (such as ADC, or DAC) is not included.

Table 33. Tyical current consumption in Sleep mode, regulator OFF(1)

Symbol	Parameter	Conditions	f_{HCLK} (MHz)	VDD=3.3 V		VDD=1.7 V		Unit
				I_{DD12}	I_{DD}	I_{DD12}	I_{DD}	
I_{DD12}/I_{DD}	Supply current in Sleep mode from V_{12} and V_{DD} supply	All Peripherals enabled	180	61.5	1.4	-	-	mA
			168	59.4	1.3	59.4	1.0	
			150	53.9	1.3	53.9	1.0	
			144	49.0	1.3	49.0	1.0	
			120	38.0	1.2	38.0	0.9	
			90	29.3	1.4	29.3	1.1	
			60	20.2	1.2	20.2	0.9	
			30	11.9	1.2	11.9	0.9	
			25	10.4	1.2	10.4	0.9	
		All Peripherals disabled	180	14.9	1.4	-	-	
			168	14.0	1.3	14.0	1.0	
			150	12.6	1.3	12.6	1.0	
			144	11.5	1.3	11.5	1.0	
			120	8.7	1.2	8.7	0.9	
			90	7.1	1.4	7.1	1.1	
			60	5.0	1.2	5.0	0.9	
			30	3.1	1.2	3.1	0.9	
			25	2.8	1.2	2.8	0.9	

1. When peripherals are enabled, the power consumption corresponding to the analog part of the peripherals (such as ADC, or DAC) is not included.

I/O system current consumption

The current consumption of the I/O system has two components: static and dynamic.

I/O static current consumption

All the I/Os used as inputs with pull-up generate current consumption when the pin is externally held low. The value of this current consumption can be simply computed by using the pull-up/pull-down resistors values given in *Table 56: I/O static characteristics*.

For the output pins, any external pull-down or external load must also be considered to estimate the current consumption.

Additional I/O current consumption is due to I/Os configured as inputs if an intermediate voltage level is externally applied. This current consumption is caused by the input Schmitt trigger circuits used to discriminate the input value. Unless this specific configuration is required by the application, this supply current consumption can be avoided by configuring these I/Os in analog mode. This is notably the case of ADC input pins which should be configured as analog inputs.

Caution: Any floating input pin can also settle to an intermediate voltage level or switch inadvertently, as a result of external electromagnetic noise. To avoid current consumption related to floating pins, they must either be configured in analog mode, or forced internally to a definite digital value. This can be done either by using pull-up/down resistors or by configuring the pins in output mode.

I/O dynamic current consumption

In addition to the internal peripheral current consumption (see *Table 35: Peripheral current consumption*), the I/Os used by an application also contribute to the current consumption. When an I/O pin switches, it uses the current from the MCU supply voltage to supply the I/O pin circuitry and to charge/discharge the capacitive load (internal or external) connected to the pin:

$$I_{SW} = V_{DD} \times f_{SW} \times C$$

where

I_{SW} is the current sunk by a switching I/O to charge/discharge the capacitive load

V_{DD} is the MCU supply voltage

f_{SW} is the I/O switching frequency

C is the total capacitance seen by the I/O pin: $C = C_{INT} + C_{EXT}$

The test pin is configured in push-pull output mode and is toggled by software at a fixed frequency.

Table 34. Switching output I/O current consumption(1)

Symbol	Parameter	Conditions	I/O toggling frequency (fsw)	Typ	Unit
I_{DDIO}	I/O switching Current	V_{DD} = 3.3 V C= C_{INT}(2)	2 MHz	0.0	mA
			8 MHz	0.2	
			25 MHz	0.6	
			50 MHz	1.1	
			60 MHz	1.3	
			84 MHz	1.8	
			90 MHz	1.9	
		V_{DD} = 3.3 V C_{EXT} = 0 pF C = C_{INT} + C_{EXT} + C_S	2 MHz	0.1	
			8 MHz	0.4	
			25 MHz	1.23	
			50 MHz	2.43	
			60 MHz	2.93	
			84 MHz	3.86	
			90 MHz	4.07	
I_{DDIO}	I/O switching Current	V_{DD} = 3.3 V C_{EXT} = 10 pF C = C_{INT} + C_{EXT} + C_S	2 MHz	0.18	mA
			8 MHz	0.67	
			25 MHz	2.09	
			50 MHz	3.6	
			60 MHz	4.5	
			84 MHz	7.8	
			90 MHz	9.8	
		V_{DD} = 3.3 V C_{EXT} = 22 pF C = C_{INT} + C_{EXT} + C_S	2 MHz	0.26	
			8 MHz	1.01	
			25 MHz	3.14	
			50 MHz	6.39	
			60 MHz	10.68	
		V_{DD} = 3.3 V C_{EXT} = 33 pF C = C_{INT} + Cext + C_S	2 MHz	0.33	
			8 MHz	1.29	
			25 MHz	4.23	
			50 MHz	11.02	

1. C_S is the PCB board capacitance including the pad pin. C_S = 7 pF (estimated value).
2. This test is performed by cutting the LQFP176 package pin (pad removal).

On-chip peripheral current consumption

The MCU is placed under the following conditions:

- At startup, all I/O pins are in analog input configuration.
- All peripherals are disabled unless otherwise mentioned.
- I/O compensation cell enabled.
- The ART accelerator is ON.
- Scale 1 mode selected, internal digital voltage V12 = 1.32 V.
- HCLK is the system clock. $f_{PCLK1} = f_{HCLK}/4$, and $f_{PCLK2} = f_{HCLK}/2$.
  The given value is calculated by measuring the difference of current consumption
  - with all peripherals clocked off
  - with only one peripheral clocked on
  - f_{HCLK} = 180 MHz (Scale1 + over-drive ON), f_{HCLK} = 144 MHz (Scale 2), f_{HCLK} = 120 MHz (Scale 3)"
- Ambient operating temperature is 25 °C and V_{DD}=3.3 V.

Table 35. Peripheral current consumption

Peripheral		I_{DD}(Typ)(1)			Unit
		Scale 1	Scale 2	Scale 3	
AHB1 (up to 180 MHz)	GPIOA	2.50	2.36	2.08	µA/MHz
	GPIOB	2.56	2.36	2.08	
	GPIOC	2.44	2.29	2.00	
	GPIOD	2.50	2.36	2.08	
	GPIOE	2.44	2.29	2.00	
	GPIOF	2.44	2.29	2.00	
	GPIOG	2.39	2.22	2.00	
	GPIOH	2.33	2.15	1.92	
	GPIOI	2.39	2.22	2.00	
	GPIOJ	2.33	2.15	1.92	
	GPIOK	2.33	2.15	1.92	
	OTG_HS+ULPI	27.00	24.86	21.92	
	CRC	0.44	0.42	0.33	
	BKPSRAM	0.78	0.69	0.58	
	DMA1	25.33	23.26	20.50	
	DMA2	24.72	22.71	20.00	
	DMA2D	28.50	26.32	23.33	
	ETH_MAC ETH_MAC_TX ETH_MAC_RX ETH_MAC_PTP	21.56	20.07	17.75	

Table 35. Peripheral current consumption (continued)

Peripheral		I_{DD}(Typ)(1)			Unit
		Scale 1	Scale 2	Scale 3	
AHB2 (up to 180 MHz)	OTG_FS	25.67	26.67	23.58	µA/MHz
	DCMI	3.72	3.40	3.00	
	RNG	2.28	2.36	2.17	
AHB3 (up to 180 MHz)	FMC	21.39	19.79	17.50	µA/MHz
Bus matrix(2)		14.06	13.19	11.75	µA/MHz
APB1 (up to 45 MHz)	TIM2	17.56	16.42	14.47	µA/MHz
	TIM3	14.22	13.36	11.80	
	TIM4	14.89	13.64	12.13	
	TIM5	17.33	16.42	14.47	
	TIM6	2.89	2.53	2.47	
	TIM7	3.11	2.81	2.47	
	TIM12	7.33	6.97	6.13	
	TIM13	4.89	4.47	4.13	
	TIM14	5.56	5.31	4.80	
	PWR	11.11	10.31	9.13	
	USART2	4.22	3.92	3.47	
	USART3	4.44	4.19	3.80	
	UART4	4.00	3.92	3.47	
	UART5	4.00	3.92	3.47	
	UART7	4.00	3.92	3.47	
	UART8	3.78	3.92	3.47	
	I2C1	4.00	3.92	3.47	
	I2C2	4.00	3.92	3.47	
	I2C3	4.00	3.92	3.47	
	SPI2(3)	3.11	3.08	2.80	
	SPI3(3)	3.56	3.36	3.13	
	I2S2	2.89	2.81	2.47	
	I2S3	3.33	3.08	2.80	
	CAN1	6.89	6.42	5.80	
	CAN2	6.67	6.14	5.47	
	DAC(4)	2.89	2.25	2.13	
	WWDG	0.89	0.86	0.80	

Table 35. Peripheral current consumption (continued)

Peripheral		I_{DD}(Typ)(1)			Unit
		Scale 1	Scale 2	Scale 3	
APB2 (up to 90 MHz)	SDIO	8.11	8.75	7.83	µA/MHz
	TIM1	17.11	15.97	14.17	
	TIM8	17.33	16.11	14.33	
	TIM9	7.22	6.67	6.00	
	TIM10	4.56	4.31	3.83	
	TIM11	4.78	4.44	4.00	
	ADC1(5)	4.67	4.31	3.83	
	ADC2(5)	4.78	4.44	4.00	
	ADC3(5)	4.56	4.17	3.67	
	SPI1	1.44	1.39	1.17	
	USART1	4.00	3.75	3.33	
	USART6	4.00	3.75	3.33	
	SPI4	1.44	1.39	1.17	
	SPI5	1.44	1.39	1.17	
	SPI6	1.44	1.39	1.17	
	SYSCFG	0.78	0.69	0.67	
	LCD_TFT	39.89	37.22	33.17	
	SAI1	3.78	3.47	3.17	

1. When the I/O compensation cell is ON, I_{DD} typical value increases by 0.22 mA.
2. The BusMatrix is automatically active when at least one master is ON.
3. To enable an I2S peripheral, first set the I2SMOD bit and then the I2SE bit in the SPI_I2SCFGR register.
4. When the DAC is ON and EN1/2 bits are set in DAC_CR register, add an additional power consumption of 0.8 mA per DAC channel for the analog part.
5. When the ADC is ON (ADON bit set in the ADC_CR2 register), add an additional power consumption of 1.6 mA per ADC for the analog part.

6.3.8 Wakeup time from low-power modes

The wakeup times given in *Table 36* are measured starting from the wakeup event trigger up to the first instruction executed by the CPU:

- For Stop or Sleep modes: the wakeup event is WFE.
- WKUP (PA0) pin is used to wakeup from Standby, Stop and Sleep modes.

All timings are derived from tests performed under ambient temperature and V_{DD}=3.3 V.

Table 36. Low-power mode wakeup timings

Symbol	Parameter	Conditions	Typ(1)	Max(1)	Unit
$t_{WUSLEEP}$(2)	Wakeup from Sleep	-	6	-	CPU clock cycle
t_{WUSTOP}(2)	Wakeup from Stop mode with MR/LP regulator in normal mode	Main regulator is ON	13.6	-	µs
		Main regulator is ON and Flash memory in Deep power down mode	93	111	
		Low power regulator is ON	22	32	
		Low power regulator is ON and Flash memory in Deep power down mode	103	126	
t_{WUSTOP}(2)	Wakeup from Stop mode with MR/LP regulator in Under-drive mode	Main regulator in under-drive mode (Flash memory in Deep power-down mode)	105	128	
		Low power regulator in under-drive mode (Flash memory in Deep power-down mode)	125	155	
tWUSTDBY (2)(3)	Wakeup from Standby mode		318	412	

1. Guaranteed by characterization results.
2. The wakeup times are measured from the wakeup event to the point in which the application code reads the first
3. $t_{WUSTDBY}$ maximum value is given at –40 °C.

6.3.9 External clock source characteristics

High-speed external user clock generated from an external source

In bypass mode the HSE oscillator is switched off and the input pin is a standard I/O. The external clock signal has to respect the *Table 56: I/O static characteristics*. However, the recommended clock input waveform is shown in *Figure 27*.

The characteristics given in *Table 37* result from tests performed using an high-speed external clock source, and under ambient temperature and supply voltage conditions summarized in *Table 17*.

Table 37. High-speed external user clock characteristics

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
f_{HSE_ext}	External user clock source frequency(1)		1	-	50	MHz
V_{HSEH}	OSC_IN input pin high level voltage		$0.7V_{DD}$	-	V_{DD}	V
V_{HSEL}	OSC_IN input pin low level voltage		V_{SS}	-	$0.3V_{DD}$	
$t_{w(HSE)}$ $t_{w(HSE)}$	OSC_IN high or low time(1)		5	-	-	ns
$t_{r(HSE)}$ $t_{f(HSE)}$	OSC_IN rise or fall time(1)		-	-	10	
$C_{in(HSE)}$	OSC_IN input capacitance(1)		-	5	-	pF
$DuCy_{(HSE)}$	Duty cycle		45	-	55	%
I_L	OSC_IN Input leakage current	$V_{SS} \leq V_{IN} \leq V_{DD}$	-	-	±1	µA

1. Guaranteed by design.

Low-speed external user clock generated from an external source

In bypass mode the LSE oscillator is switched off and the input pin is a standard I/O. The external clock signal has to respect the *Table 56: I/O static characteristics*. However, the recommended clock input waveform is shown in *Figure 28*.

The characteristics given in *Table 38* result from tests performed using an low-speed external clock source, and under ambient temperature and supply voltage conditions summarized in *Table 17*.

Table 38. Low-speed external user clock characteristics

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
f_{LSE_ext}	User External clock source frequency(1)		-	32.768	1000	kHz
V_{LSEH}	OSC32_IN input pin high level voltage		$0.7V_{DD}$	-	V_{DD}	V
V_{LSEL}	OSC32_IN input pin low level voltage		V_{SS}	-	$0.3V_{DD}$	
$t_{w(LSE)}$ $t_{f(LSE)}$	OSC32_IN high or low time(1)		450	-	-	ns
$t_{r(LSE)}$ $t_{f(LSE)}$	OSC32_IN rise or fall time(1)		-	-	50	
$C_{in(LSE)}$	OSC32_IN input capacitance(1)		-	5	-	pF
$DuCy_{(LSE)}$	Duty cycle		30	-	70	%
I_L	OSC32_IN Input leakage current	$V_{SS} \leq V_{IN} \leq V_{DD}$	-	-	±1	µA

1. Guaranteed by design.

Figure 27. High-speed external clock source AC timing diagram

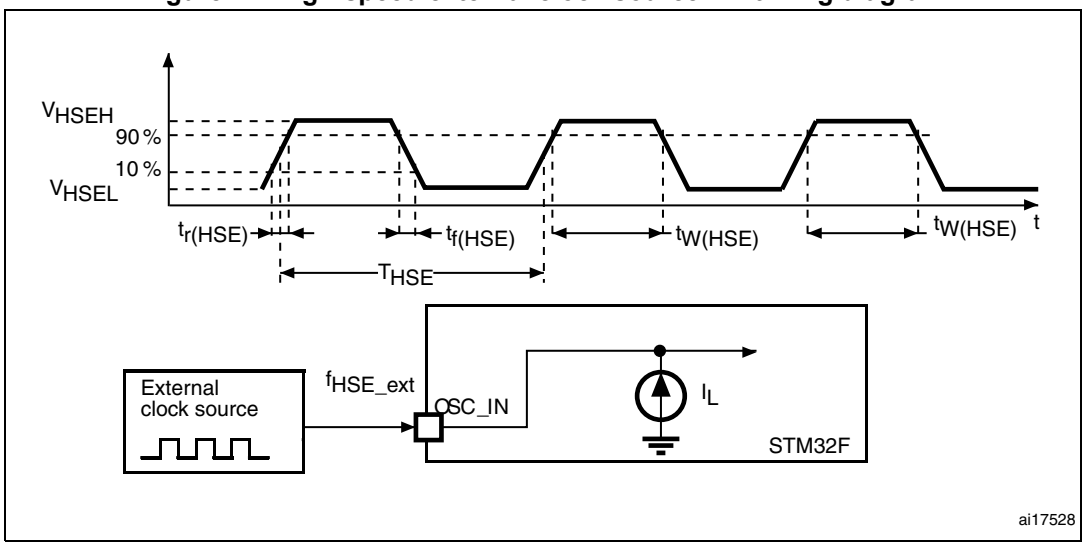
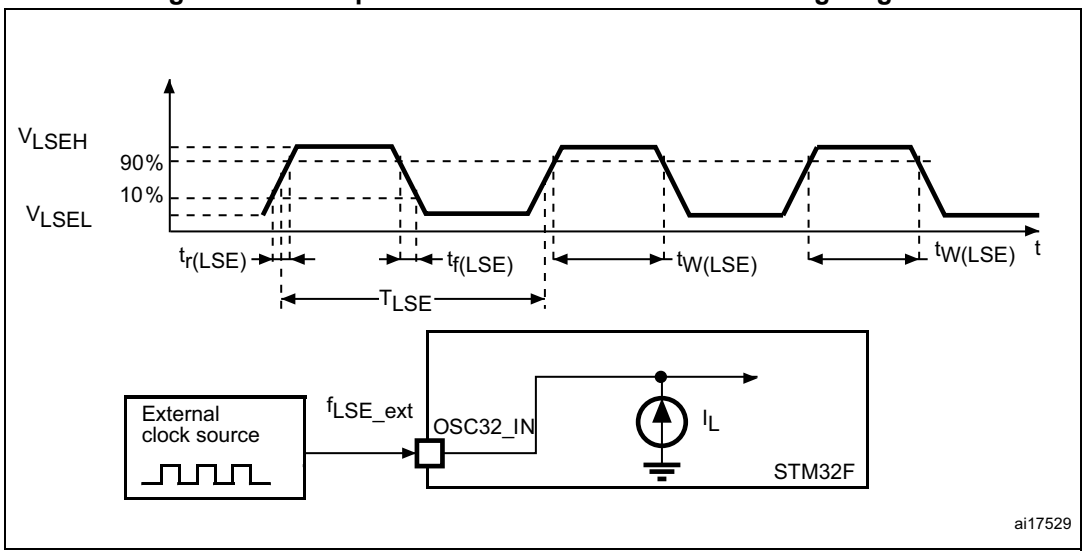


Figure 28. Low-speed external clock source AC timing diagram



High-speed external clock generated from a crystal/ceramic resonator

The high-speed external (HSE) clock can be supplied with a 4 to 26 MHz crystal/ceramic resonator oscillator. All the information given in this paragraph are based on characterization results obtained with typical external components specified in *Table 39*. In the application, the resonator and the load capacitors have to be placed as close as possible to the oscillator pins in order to minimize output distortion and startup stabilization time. Refer to the crystal resonator manufacturer for more details on the resonator characteristics (frequency, package, accuracy).

Table 39. HSE 4-26 MHz oscillator characteristics (1)

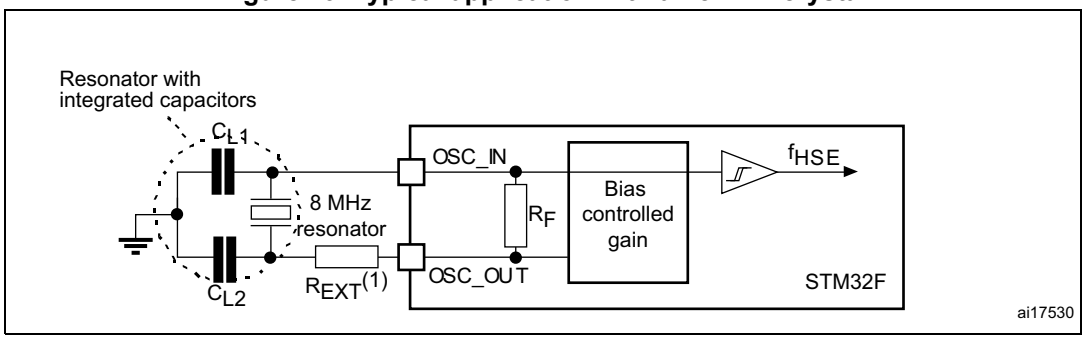
Symbol	Parameter	Conditions	Min	Typ	Max	Unit
f_{OSC_IN}	Oscillator frequency		4	-	26	MHz
R_F	Feedback resistor		-	200	-	kΩ
I_{DD}	HSE current consumption	V_{DD}=3.3 V, ESR= 30 Ω, C_L=5 pF@25 MHz	-	450	-	µA
		V_{DD}=3.3 V, ESR= 30 Ω, C_L=10 pF@25 MHz	-	530	-	
ACC_{HSE} (2)	HSE accuracy		− 500	-	500	ppm
G_m_crit_max	Maximum critical crystal g_m	Startup	-	-	1	mA/V
$t_{SU(HSE}$ (3)	Startup time	V_{DD} is stabilized	-	2	-	ms

1. Guaranteed by design.
2. This parameter depends on the crystal used in the application. The minimum and maximum values must be respected to comply with USB standard specifications.
3. $t_{SU(HSE)}$ is the startup time measured from the moment it is enabled (by software) to a stabilized 8 MHz oscillation is reached. This value is based on characterization and not tested in production. It is measured for a standard crystal resonator and it can vary significantly with the crystal manufacturer.

For C_{L1} and C_{L2}, it is recommended to use high-quality external ceramic capacitors in the 5 pF to 25 pF range (typ.), designed for high-frequency applications, and selected to match the requirements of the crystal or resonator (see *Figure 29*). C_{L1} and C_{L2} are usually the same size. The crystal manufacturer typically specifies a load capacitance which is the series combination of C_{L1} and C_{L2}. PCB and MCU pin capacitance must be included (10 pF can be used as a rough estimate of the combined pin and board capacitance) when sizing C_{L1} and C_{L2}.

Note: For information on selecting the crystal, refer to the application note AN2867 "Oscillator design guide for ST microcontrollers" available from the ST website www.st.com.

Figure 29. Typical application with an 8 MHz crystal



1. R_{EXT} value depends on the crystal characteristics.

Low-speed external clock generated from a crystal/ceramic resonator

The low-speed external (LSE) clock can be supplied with a 32.768 kHz crystal/ceramic resonator oscillator. All the information given in this paragraph are based on characterization results obtained with typical external components specified in *Table 40*. In the application, the resonator and the load capacitors have to be placed as close as possible to the oscillator pins in order to minimize output distortion and startup stabilization time. Refer to the crystal resonator manufacturer for more details on the resonator characteristics (frequency, package, accuracy).

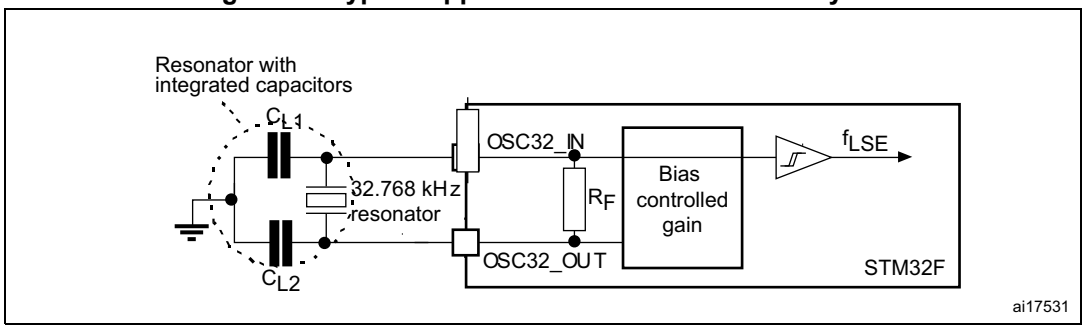
Table 40. LSE oscillator characteristics (f_{LSE} = 32.768 kHz) (1)

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
R_F	Feedback resistor		-	18.4	-	MΩ
I_{DD}	LSE current consumption		-	-	1	µA
ACC_{LSE} (2)	LSE accuracy		− 500	-	500	ppm
G_m_crit_max	Maximum critical crystal g_m	Startup	-	-	0.56	µA/V
$t_{SU(LSE)}$ (3)	startup time	V_{DD} is stabilized	-	2	-	s

1. Guaranteed by design.
2. This parameter depends on the crystal used in the application. Refer to application note AN2867.
3. $t_{SU(LSE)}$ is the startup time measured from the moment it is enabled (by software) to a stabilized 32.768 kHz oscillation is reached. This value is based on characterization and not tested in production. It is measured for a standard crystal resonator and it can vary significantly with the crystal manufacturer.

Note: For information on selecting the crystal, refer to the application note AN2867 "Oscillator design guide for ST microcontrollers" available from the ST website www.st.com.

Figure 30. Typical application with a 32.768 kHz crystal



6.3.10 Internal clock source characteristics

The parameters given in *Table 41* and *Table 42* are derived from tests performed under ambient temperature and V_{DD} supply voltage conditions summarized in *Table 17*.

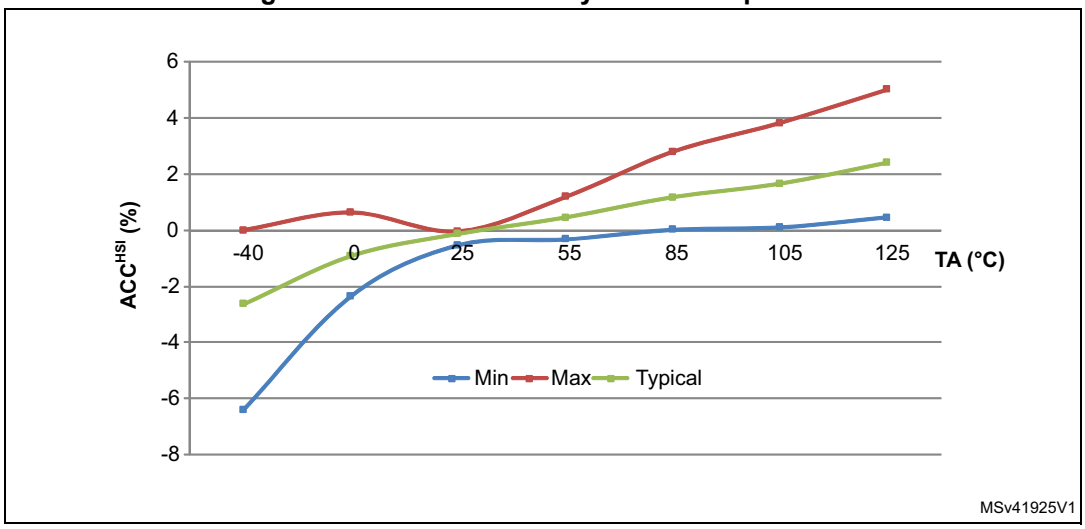
High-speed internal (HSI) RC oscillator

Table 41. HSI oscillator characteristics (1)

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
f_{HSI}	Frequency	-	-	16	-	MHz
ACC_{HSI}	HSI user-trimming step (2)	-	-	-	1	%
	Accuracy of the HSI oscillator	T_A = –40 to 105 °C(3)	− 8	-	4.5	%
		T_A = –10 to 85 °C(3)	− 4	-	4	%
		T_A = 25 °C(4)	− 1	-	1	%
$t_{su(HSI)}$(2)	HSI oscillator startup time	-	-	2.2	4	µs
$I_{DD(HSI)}$(2)	HSI oscillator power consumption	-	-	60	80	µA

1. V_{DD} = 3.3 V, T_A = –40 to 105 °C unless otherwise specified.
2. Guaranteed by design.
3. Guaranteed by characterization results.
4. Factory calibrated, parts not soldered.

Figure 31. ACCHSI accuracy versus temperature



1. Guaranteed by characterization results.

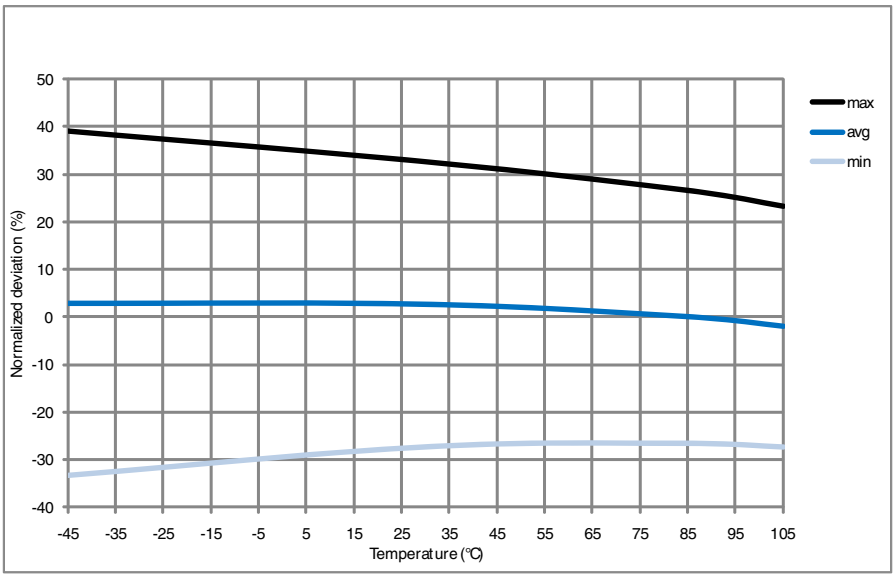
Low-speed internal (LSI) RC oscillator

Table 42. LSI oscillator characteristics (1)

Symbol	Parameter	Min	Typ	Max	Unit
f_{LSI}(2)	Frequency	17	32	47	kHz
$t_{su(LSI)}$(3)	LSI oscillator startup time	-	15	40	µs
$I_{DD(LSI)}$(3)	LSI oscillator power consumption	-	0.4	0.6	µA

1. V_{DD} = 3 V, T_A = –40 to 105 °C unless otherwise specified.
2. Guaranteed by characterization results.
3. Guaranteed by design.

Figure 32. ACC_{LSI} versus temperature



6.3.11 PLL characteristics

The parameters given in *Table 43* and *Table 44* are derived from tests performed under temperature and V_{DD} supply voltage conditions summarized in *Table 17*.

Table 43. Main PLL characteristics

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
f_{PLL_IN}	PLL input clock(1)		0.95(2)	1	2.10	MHz
f_{PLL_OUT}	PLL multiplier output clock		24	-	180	MHz
f_{PLL48_OUT}	48 MHz PLL multiplier output clock		-	48	75	MHz
f_{VCO_OUT}	PLL VCO output		100	-	432	MHz
t_{LOCK}	PLL lock time	VCO freq = 100 MHz	75	-	200	µs
		VCO freq = 432 MHz	100	-	300	

Table 43. Main PLL characteristics (continued)

Symbol	Parameter	Conditions		Min	Typ	Max	Unit
Jitter(3)	Cycle-to-cycle jitter	System clock 120 MHz	RMS	-	25	-	ps
			peak to peak	-	±150	-	
	Period Jitter		RMS	-	15	-	
			peak to peak	-	±200	-	
	Main clock output (MCO) for RMII Ethernet	Cycle to cycle at 50 MHz on 1000 samples		-	32	-	
	Main clock output (MCO) for MII Ethernet	Cycle to cycle at 25 MHz on 1000 samples		-	40	-	
	Bit Time CAN jitter	Cycle to cycle at 1 MHz on 1000 samples		-	330	-	
$I_{DD(PLL)}$(4)	PLL power consumption on VDD	VCO freq = 100 MHz VCO freq = 432 MHz		0.15 0.45	-	0.40 0.75	mA
$I_{DDA(PLL)}$(4)	PLL power consumption on VDDA	VCO freq = 100 MHz VCO freq = 432 MHz		0.30 0.55	-	0.40 0.85	mA

1. Take care of using the appropriate division factor M to obtain the specified PLL input clock values. The M factor is shared between PLL and PLLI2S.
2. Guaranteed by design.
3. The use of 2 PLLs in parallel could degraded the Jitter up to +30%.
4. Guaranteed by characterization results.

Table 44. PLLI2S (audio PLL) characteristics

Symbol	Parameter	Conditions		Min	Typ	Max	Unit
f_{PLLI2S_IN}	PLLI2S input clock(1)			0.95(2)	1	2.10	MHz
f_{PLLI2S_OUT}	PLLI2S multiplier output clock			-	-	216	MHz
f_{VCO_OUT}	PLLI2S VCO output			100	-	432	MHz
t_{LOCK}	PLLI2S lock time	VCO freq = 100 MHz		75	-	200	µs
		VCO freq = 432 MHz		100	-	300	
Jitter(3)	Master I2S clock jitter	Cycle to cycle at 12.288 MHz on 48KHz period, N=432, R=5	RMS	-	90	-	
			peak to peak	-	±280	-	ps
		Average frequency of 12.288 MHz N = 432, R = 5 on 1000 samples		-	90	-	ps
	WS I2S clock jitter	Cycle to cycle at 48 KHz on 1000 samples		-	400	-	ps

Table 44. PLLI2S (audio PLL) characteristics (continued)

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
$I_{DD(PLLI2S)}$(4)	PLLI2S power consumption on V_{DD}	VCO freq = 100 MHz VCO freq = 432 MHz	0.15 0.45	-	0.40 0.75	mA
$I_{DDA(PLLI2S)}$(4)	PLLI2S power consumption on V_{DDA}	VCO freq = 100 MHz VCO freq = 432 MHz	0.30 0.55	-	0.40 0.85	mA

1. Take care of using the appropriate division factor M to have the specified PLL input clock values.
2. Guaranteed by design.
3. Value given with main PLL running.
4. Guaranteed by characterization results.

Table 45. PLLISAI (audio and LCD-TFT PLL) characteristics

Symbol	Parameter	Conditions		Min	Typ	Max	Unit
f_{PLLSAI_IN}	PLLSAI input clock(1)			0.95(2)	1	2.10	MHz
f_{PLLSAI_OUT}	PLLSAI multiplier output clock			-	-	216	MHz
f_{VCO_OUT}	PLLSAI VCO output			100	-	432	MHz
t_{LOCK}	PLLSAI lock time	VCO freq = 100 MHz		75	-	200	µs
		VCO freq = 432 MHz		100	-	300	
Jitter(3)	Main SAI clock jitter	Cycle to cycle at 12.288 MHz on 48KHz period, N=432, R=5	RMS	-	90	-	ps
			peak to peak	-	±280	-	
		Average frequency of 12.288 MHz N = 432, R = 5 on 1000 samples		-	90	-	ps
	FS clock jitter	Cycle to cycle at 48 KHz on 1000 samples		-	400	-	ps
$I_{DD(PLLSAI)}$(4)	PLLSAI power consumption on V_{DD}	VCO freq = 100 MHz VCO freq = 432 MHz		0.15 0.45	-	0.40 0.75	mA
$I_{DDA(PLLSAI)}$(4)	PLLSAI power consumption on V_{DDA}	VCO freq = 100 MHz VCO freq = 432 MHz		0.30 0.55	-	0.40 0.85	mA

1. Take care of using the appropriate division factor M to have the specified PLL input clock values.
2. Guaranteed by design.
3. Value given with main PLL running.
4. Guaranteed by characterization results.

6.3.12 PLL spread spectrum clock generation (SSCG) characteristics

The spread spectrum clock generation (SSCG) feature allows to reduce electromagnetic interferences (see *Table 52: EMI characteristics*). It is available only on the main PLL.

Table 46. SSCG parameters constraint

Symbol	Parameter	Min	Typ	Max(1)	Unit
f_{Mod}	Modulation frequency	-	-	10	KHz
md	Peak modulation depth	0.25	-	2	%
MODEPER * INCSTEP		-	-	$2^{15} - 1$	-

1. Guaranteed by design.

Equation 1

The frequency modulation period (MODEPER) is given by the equation below:

$$MODEPER = round[f_{PLL_IN} / (4 \times f_{Mod})]$$

f_{PLL_IN} and f_{Mod} must be expressed in Hz.

As an example:

If f_{PLL_IN} = 1 MHz, and f_{MOD} = 1 kHz, the modulation depth (MODEPER) is given by equation 1:

$$MODEPER = round[10^6 / (4 \times 10^3)] = 250$$

Equation 2

Equation 2 allows to calculate the increment step (INCSTEP):

$$INCSTEP = round[((2^{15} - 1) \times md \times PLLN) / (100 \times 5 \times MODEPER)]$$

f_{VCO_OUT} must be expressed in MHz.

With a modulation depth (md) = ±2 % (4 % peak to peak), and PLLN = 240 (in MHz):

$$INCSTEP = round[((2^{15} - 1) \times 2 \times 240) / (100 \times 5 \times 250)] = 126md(quantitazed)\%$$

An amplitude quantization error may be generated because the linear modulation profile is obtained by taking the quantized values (rounded to the nearest integer) of MODPER and INCSTEP. As a result, the achieved modulation depth is quantized. The percentage quantized modulation depth is given by the following formula:

$$md_{quantized}\% = (MODEPER \times INCSTEP \times 100 \times 5) / ((2^{15} - 1) \times PLLN)$$

As a result:

$$md_{quantized}\% = (250 \times 126 \times 100 \times 5) / ((2^{15} - 1) \times 240) = 2.002\%(peak)$$

Figure 33 and *Figure 34* show the main PLL output clock waveforms in center spread and down spread modes, where:

F0 is f_{PLL_OUT} nominal.

T_{mode} is the modulation period.

md is the modulation depth.

Figure 33. PLL output clock waveforms in center spread mode

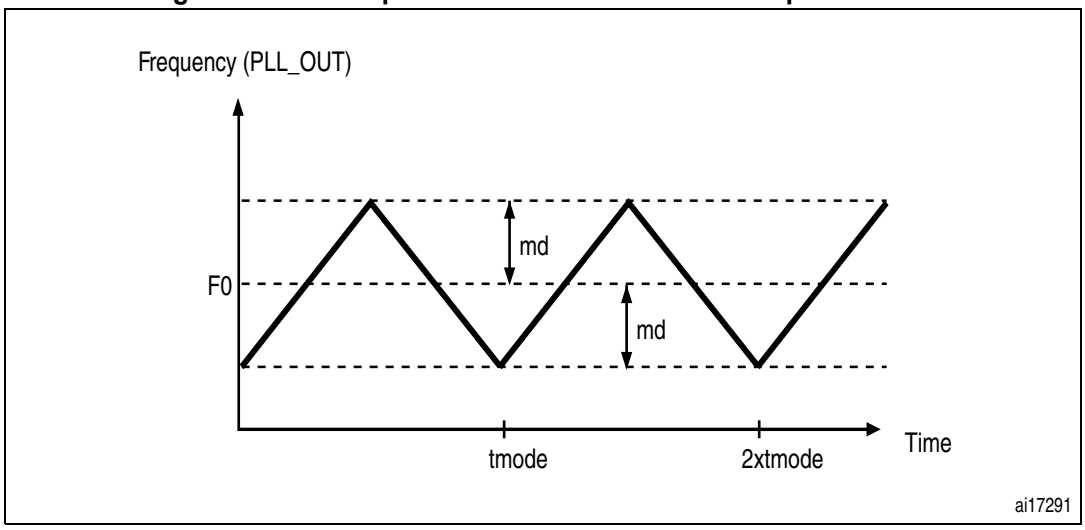
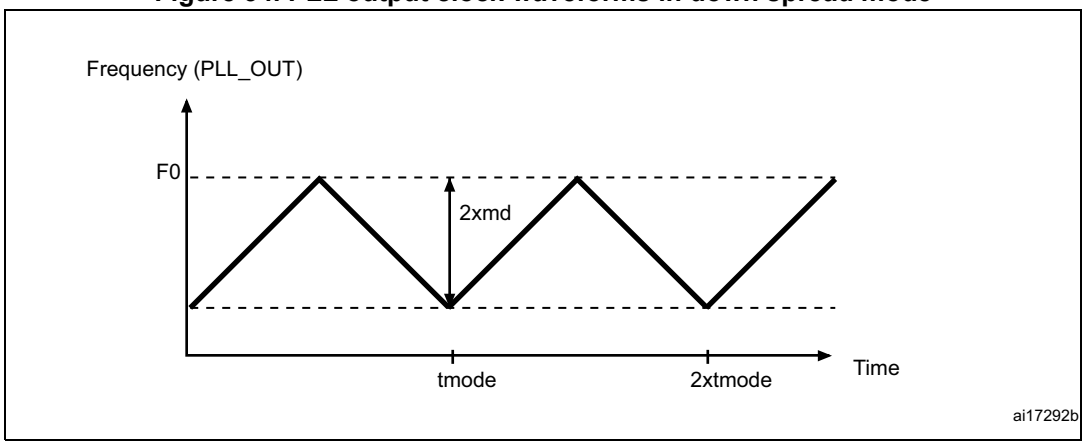


Figure 34. PLL output clock waveforms in down spread mode



6.3.13 Memory characteristics

Flash memory

The characteristics are given at TA = –40 to 105 °C unless otherwise specified.

The devices are shipped to customers with the Flash memory erased.

Table 47. Flash memory characteristics

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
I_{DD}	Supply current	Write / Erase 8-bit mode, V_{DD} = 1.7 V	-	5	-	mA
		Write / Erase 16-bit mode, V_{DD} = 2.1 V	-	8	-	
		Write / Erase 32-bit mode, V_{DD} = 3.3 V	-	12	-	

Table 48. Flash memory programming

Symbol	Parameter	Conditions	Min(1)	Typ	Max(1)	Unit
t_{prog}	Word programming time	Program/erase parallelism (PSIZE) = x 8/16/32	-	16	100(2)	µs
$t_{ERASE16KB}$	Sector (16 KB) erase time	Program/erase parallelism (PSIZE) = x 8	-	400	800	ms
		Program/erase parallelism (PSIZE) = x 16	-	300	600	
		Program/erase parallelism (PSIZE) = x 32	-	250	500	
$t_{ERASE64KB}$	Sector (64 KB) erase time	Program/erase parallelism (PSIZE) = x 8	-	1200	2400	ms
		Program/erase parallelism (PSIZE) = x 16	-	700	1400	
		Program/erase parallelism (PSIZE) = x 32	-	550	1100	
$t_{ERASE128KB}$	Sector (128 KB) erase time	Program/erase parallelism (PSIZE) = x 8	-	2	4	s
		Program/erase parallelism (PSIZE) = x 16	-	1.3	2.6	
		Program/erase parallelism (PSIZE) = x 32	-	1	2	
t_{ME}	Mass erase time	Program/erase parallelism (PSIZE) = x 8	-	16	32	s
		Program/erase parallelism (PSIZE) = x 16	-	11	22	
		Program/erase parallelism (PSIZE) = x 32	-	8	16	

Table 48. Flash memory programming (continued)

Symbol	Parameter	Conditions	Min(1)	Typ	Max(1)	Unit
t_{BE}	Bank erase time	Program/erase parallelism (PSIZE) = x 8	-	16	32	s
		Program/erase parallelism (PSIZE) = x 16	-	11	22	
		Program/erase parallelism (PSIZE) = x 32	-	8	16	
V_{prog}	Programming voltage	32-bit program operation	2.7	-	3.6	V
		16-bit program operation	2.1	-	3.6	V
		8-bit program operation	1.7	-	3.6	V

1. Guaranteed by characterization results.
2. The maximum programming time is measured after 100K erase operations.

Table 49. Flash memory programming with V_{PP}

Symbol	Parameter	Conditions	Min(1)	Typ	Max(1)	Unit
t_{prog}	Double word programming	T_A = 0 to +40 °C V_{DD} = 3.3 V V_{PP} = 8.5 V	-	16	100(2)	µs
$t_{ERASE16KB}$	Sector (16 KB) erase time		-	230	-	ms
$t_{ERASE64KB}$	Sector (64 KB) erase time		-	490	-	
$t_{ERASE128KB}$	Sector (128 KB) erase time		-	875	-	
t_{ME}	Mass erase time		-	6.9	-	s
t_{BE}	Bank erase time		-	6.9	-	s
V_{prog}	Programming voltage		2.7	-	3.6	V
V_{PP}	V_{PP} voltage range		7	-	9	V
I_{PP}	Minimum current sunk on the V_{PP} pin		10	-	-	mA
t_{VPP}(3)	Cumulative time during which V_{PP} is applied		-	-	1	hour

1. Guaranteed by design.
2. The maximum programming time is measured after 100K erase operations.
3. V_{PP} should only be connected during programming/erasing.

Table 50. Flash memory endurance and data retention

Symbol	Parameter	Conditions	Value	Unit
			Min(1)	
N_{END}	Endurance	T_A = –40 to +85 °C (6 suffix versions) T_A = –40 to +105 °C (7 suffix versions)	10	kcycles
t_{RET}	Data retention	1 kcycle(2) at T_A = 85 °C	30	Years
		1 kcycle(2) at T_A = 105 °C	10	
		10 kcycles(2) at T_A = 55 °C	20	

1. Guaranteed by characterization results.
2. Cycling performed over the whole temperature range.

6.3.14 EMC characteristics

Susceptibility tests are performed on a sample basis during device characterization.

Functional EMS (electromagnetic susceptibility)

While a simple application is executed on the device (toggling 2 LEDs through I/O ports). the device is stressed by two electromagnetic events until a failure occurs. The failure is indicated by the LEDs:

- **Electrostatic discharge (ESD)** (positive and negative) is applied to all device pins until a functional disturbance occurs. This test is compliant with the IEC 61000-4-2 standard.
- **FTB**: A burst of fast transient voltage (positive and negative) is applied to V_{DD} and V_{SS} through a 100 pF capacitor, until a functional disturbance occurs. This test is compliant with the IEC 61000-4-4 standard.

A device reset allows normal operations to be resumed.

The test results are given in *Table 51*. They are based on the EMS levels and classes defined in application note AN1709.

Table 51. EMS characteristics

Symbol	Parameter	Conditions	Level/ Class
V_{FESD}	Voltage limits to be applied on any I/O pin to induce a functional disturbance	V_{DD} = 3.3 V, LQFP176, T_A = +25 °C, f_{HCLK} = 168 MHz, conforms to IEC 61000-4-2	2B
V_{EFTB}	Fast transient voltage burst limits to be applied through 100 pF on V_{DD} and V_{SS} pins to induce a functional disturbance	V_{DD} = 3.3 V, LQFP176, T_A =+25 °C, f_{HCLK} = 168 MHz, conforms to IEC 61000-4-2	4A

When the application is exposed to a noisy environment, it is recommended to avoid pin exposition to disturbances. The pins showing a middle range robustness are: PA0, PA1, PA2, PH2, PH3, PH4, PH5, PA3, PA4, PA5, PA6, PA7, PC4, and PC5.

As a consequence, it is recommended to add a serial resistor (1 kΩ) located as close as possible to the MCU to the pins exposed to noise (connected to tracks longer than 50 mm on PCB).

Designing hardened software to avoid noise problems

EMC characterization and optimization are performed at component level with a typical application environment and simplified MCU software. It should be noted that good EMC performance is highly dependent on the user application and the software in particular.

Therefore it is recommended that the user applies EMC software optimization and prequalification tests in relation with the EMC level requested for his application.

Software recommendations

The software flowchart must include the management of runaway conditions such as:

- Corrupted program counter
- Unexpected reset
- Critical Data corruption (control registers...)

Prequalification trials

Most of the common failures (unexpected reset and program counter corruption) can be reproduced by manually forcing a low state on the NRST pin or the Oscillator pins for 1 second.

To complete these trials, ESD stress can be applied directly on the device, over the range of specification values. When unexpected behavior is detected, the software can be hardened to prevent unrecoverable errors occurring (see application note AN1015).

Electromagnetic Interference (EMI)

The electromagnetic field emitted by the device are monitored while a simple application, executing EEMBC? code, is running. This emission test is compliant with SAE IEC61967-2 standard which specifies the test board and the pin loading.

Table 52. EMI characteristics

Symbol	Parameter	Conditions	Monitored frequency band	Max vs. [f_{HSE}/f_{CPU}] 25/168 MHz	Max vs. [f_{HSE}/f_{CPU}] 25/180 MHz	Unit
S_{EMI}	Peak level	V_{DD} = 3.3 V, T_A = 25 °C, LQFP176 package, conforming to SAE J1752/3 EEMBC, ART ON, all peripheral clocks enabled, clock dithering disabled.	0.1 to 30 MHz	16	19	dBµV
			30 to 130 MHz	23	23	
			130 MHz to 1GHz	25	22	
			SAE EMI Level	4	4	-
		V_{DD} = 3.3 V, T_A = 25 °C, LQFP176 package, conforming to SAE J1752/3 EEMBC, ART ON, all peripheral clocks enabled, clock dithering enabled	0.1 to 30 MHz	17	16	dBµV
			30 to 130 MHz	8	10	
			130 MHz to 1GHz	11	16	
			SAE EMI level	3.5	3.5	-

6.3.15 Absolute maximum ratings (electrical sensitivity)

Based on three different tests (ESD, LU) using specific measurement methods, the device is stressed in order to determine its performance in terms of electrical sensitivity.

Electrostatic discharge (ESD)

Electrostatic discharges (a positive then a negative pulse separated by 1 second) are applied to the pins of each sample according to each pin combination. The sample size depends on the number of supply pins in the device (3 parts × (n+1) supply pins). This test conforms to the ANSI/ESDA/JEDEC JS-001 and ANSI/ESD S5.3.1 standards.

Table 53. ESD absolute maximum ratings

Symbol	Ratings	Conditions	Class	Maximum value(1)	Unit
$V_{ESD(HBM)}$	Electrostatic discharge voltage (human body model)	T_A = +25 °C conforming to ANSI/ESDA/JEDEC JS-001	2	2000	V
$V_{ESD(CDM)}$	Electrostatic discharge voltage (charge device model)	T_A = +25 °C conforming to ANSI/ESD S5.3.1, LQFP100/144/176, UFBGA169/176, TFBGA176 and WLCSP143 packages	C3	250	
		T_A = +25 °C conforming to ANSI/ESD S5.3.1, LQFP208 package	C3	250	

1. Guaranteed by characterization results.

Static latchup

Two complementary static tests are required on six parts to assess the latchup performance:

- A supply overvoltage is applied to each power supply pin
- A current injection is applied to each input, output and configurable I/O pin

These tests are compliant with EIA/JESD 78A IC latchup standard.

Table 54. Electrical sensitivities

Symbol	Parameter	Conditions	Class
LU	Static latch-up class	T_A = +105 °C conforming to JESD78A	II level A

6.3.16 I/O current injection characteristics

As a general rule, current injection to the I/O pins, due to external voltage below V_{SS} or above V_{DD} (for standard, 3 V-capable I/O pins) should be avoided during normal product operation. However, in order to give an indication of the robustness of the microcontroller in cases when abnormal injection accidentally happens, susceptibility tests are performed on a sample basis during device characterization.

Functional susceptibilty to I/O current injection

While a simple application is executed on the device, the device is stressed by injecting current into the I/O pins programmed in floating input mode. While current is injected into the I/O pin, one at a time, the device is checked for functional failures.

The failure is indicated by an out of range parameter: ADC error above a certain limit (>5 LSB TUE), out of conventional limits of induced leakage current on adjacent pins (out of –5 µA/+0 µA range), or other functional failure (for example reset, oscillator frequency deviation).

Negative induced leakage current is caused by negative injection and positive induced leakage current by positive injection.

The test results are given in *Table 55*.

Table 55. I/O current injection susceptibility(1)

Symbol	Description	Functional susceptibility		Unit
		Negative injection	Positive injection	
I_{INJ}	Injected current on BOOT0 pin	− 0	NA	mA
	Injected current on NRST pin	− 0	NA	
	Injected current on PA0, PA1, PA2, PA3, PA6, PA7, PB0, PC0, PC1, PC2, PC3, PC4, PC5, PH1, PH2, PH3, PH4, PH5	− 0	NA	
	Injected current on TTa pins: PA4 and PA5	− 0	+5	
	Injected current on any other FT pin	− 5	NA	

1. NA = not applicable.

Note: It is recommended to add a Schottky diode (pin to ground) to analog pins which may potentially inject negative currents.

6.3.17 I/O port characteristics

General input/output characteristics

Unless otherwise specified, the parameters given in *Table 56: I/O static characteristics* are derived from tests performed under the conditions summarized in *Table 17*. All I/Os are CMOS and TTL compliant.

Table 56. I/O static characteristics

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
V_{IL}	FT, TTa and NRST I/O input low level voltage	1.7 V≤V_{DD}≤3.6 V	-	-	$0.35V_{DD}$ − 0.04 (1)	V
					$0.3V_{DD}$ (2)	
	BOOT0 I/O input low level voltage	1.75 V≤V_{DD} ≤3.6 V, – 40 °C≤T_A ≤105 °C	-	-	$0.1V_{DD}$+0.1 (1)	
		1.7 V≤V_{DD} ≤3.6 V, 0 °C≤T_A ≤105 °C	-	-		
V_{IH}	FT, TTa and NRST I/O input high level voltage (5)	1.7 V≤V_{DD}≤3.6 V	$0.45V_{DD}$+0.3 (1)	-	-	V
			$0.7V_{DD}$ (2)			
	BOOT0 I/O input high level voltage	1.75 V≤V_{DD} ≤3.6 V, – 40 °C≤T_A ≤105 °C	$0.17V_{DD}$+0.7 (1)	-	-	
		1.7 V≤V_{DD} ≤3.6 V, 0 °C≤T_A ≤105 °C				
V_{HYS}	FT, TTa and NRST I/O input hysteresis	1.7 V≤V_{DD}≤3.6 V	10%V_{DD} (3)	-	-	V
	BOOT0 I/O input hysteresis	1.75 V≤V_{DD} ≤3.6 V, – 40 °C≤T_A ≤105 °C	0.1	-	-	
		1.7 V≤V_{DD} ≤3.6 V, 0 °C≤T_A ≤105 °C				
I_{lkg}	I/O input leakage current (4)	V_{SS} ≤V_{IN} ≤V_{DD}	-	-	±1	µA
	I/O FT input leakage current (5)	V_{IN} = 5 V	-	-	3	

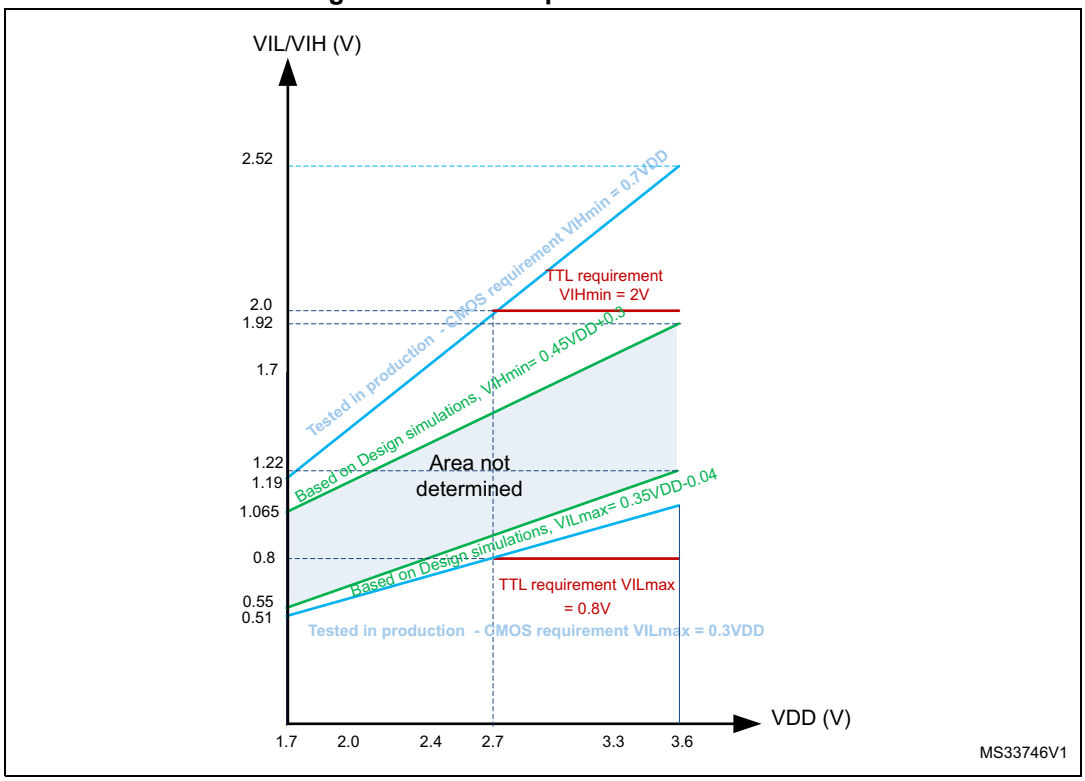
Table 56. I/O static characteristics (continued)

Symbol	Parameter		Conditions	Min	Typ	Max	Unit
R_{PU}	Weak pull-up equivalent resistor(6)	All pins except for PA10/PB12 (OTG_FS_ID, OTG_HS_ID)	$V_{IN} = V_{SS}$	30	40	50	kΩ
		PA10/PB12 (OTG_FS_ID, OTG_HS_ID)		7	10	14	
R_{PD}	Weak pull-down equivalent resistor(7)	All pins except for PA10/PB12 (OTG_FS_ID, OTG_HS_ID)	$V_{IN} = V_{DD}$	30	40	50	
		PA10/PB12 (OTG_FS_ID, OTG_HS_ID)		7	10	14	
C_{IO}(8)	I/O pin capacitance		-	-	5	-	pF

1. Guaranteed by design.
2. Tested in production.
3. With a minimum of 200 mV.
4. Leakage could be higher than the maximum value, if negative current is injected on adjacent pins, Refer to *Table 55: I/O current injection susceptibility*
5. To sustain a voltage higher than VDD +0.3 V, the internal pull-up/pull-down resistors must be disabled. Leakage could be higher than the maximum value, if negative current is injected on adjacent pins.Refer to *Table 55: I/O current injection susceptibility*
6. Pull-up resistors are designed with a true resistance in series with a switchable PMOS. This PMOS contribution to the series resistance is minimum (~10% order).
7. Pull-down resistors are designed with a true resistance in series with a switchable NMOS. This NMOS contribution to the series resistance is minimum (~10% order).
8. Hysteresis voltage between Schmitt trigger switching levels. Guaranteed by characterization results.

All I/Os are CMOS and TTL compliant (no software configuration required). Their characteristics cover more than the strict CMOS-technology or TTL parameters. The coverage of these requirements for FT I/Os is shown in *Figure 35*.

Figure 35. FT I/O input characteristics



Output driving current

The GPIOs (general purpose input/outputs) can sink or source up to ±8 mA, and sink or source up to ±20 mA (with a relaxed V_{OL}/V_{OH}) except PC13, PC14, PC15 and PI8 which can sink or source up to ±3mA. When using the PC13 to PC15 and PI8 GPIOs in output mode, the speed should not exceed 2 MHz with a maximum load of 30 pF.

In the user application, the number of I/O pins which can drive current must be limited to respect the absolute maximum rating specified in *Section 6.2*. In particular:

- The sum of the currents sourced by all the I/Os on $V_{DD,}$ plus the maximum Run consumption of the MCU sourced on $V_{DD,}$ cannot exceed the absolute maximum rating ΣI_{VDD} (see *Table 15*).
- The sum of the currents sunk by all the I/Os on V_{SS} plus the maximum Run consumption of the MCU sunk on V_{SS} cannot exceed the absolute maximum rating ΣI_{VSS} (see *Table 15*).

Output voltage levels

Unless otherwise specified, the parameters given in *Table 57* are derived from tests performed under ambient temperature and V_{DD} supply voltage conditions summarized in *Table 17*. All I/Os are CMOS and TTL compliant.

Table 57. Output voltage characteristics

Symbol	Parameter	Conditions	Min	Max	Unit
V_{OL}(1)	Output low level voltage for an I/O pin	CMOS port(2) I_{IO} = +8 mA 2.7 V ≤V_{DD} ≤3.6 V	-	0.4	V
V_{OH}(3)	Output high level voltage for an I/O pin		V_{DD} − 0.4	-	
V_{OL} (1)	Output low level voltage for an I/O pin	TTL port(2) I_{IO} =+ 8mA 2.7 V ≤V_{DD} ≤3.6 V	-	0.4	V
V_{OH} (3)	Output high level voltage for an I/O pin		2.4	-	
V_{OL}(1)	Output low level voltage for an I/O pin	I_{IO} = +20 mA 2.7 V ≤V_{DD} ≤3.6 V	-	1.3(4)	V
V_{OH}(3)	Output high level voltage for an I/O pin		V_{DD}−1.3(4)	-	
V_{OL}(1)	Output low level voltage for an I/O pin	I_{IO} = +6 mA 1.8 V ≤V_{DD} ≤3.6 V	-	0.4(4)	V
V_{OH}(3)	Output high level voltage for an I/O pin		V_{DD}−0.4(4)	-	
V_{OL}(1)	Output low level voltage for an I/O pin	I_{IO} = +4 mA 1.7 V ≤V_{DD} ≤3.6V	-	0.4(5)	V
V_{OH}(3)	Output high level voltage for an I/O pin		V_{DD}−0.4(5)	-	

1. The I_{IO} current sunk by the device must always respect the absolute maximum rating specified in *Table 15*. and the sum of I_{IO} (I/O ports and control pins) must not exceed I_{VSS}.
2. TTL and CMOS outputs are compatible with JEDEC standards JESD36 and JESD52.
3. The I_{IO} current sourced by the device must always respect the absolute maximum rating specified in *Table 15* and the sum of I_{IO} (I/O ports and control pins) must not exceed I_{VDD}.
4. Based on characterization data.
5. Guaranteed by design.

Input/output AC characteristics

The definition and values of input/output AC characteristics are given in *Figure 36* and *Table 58*, respectively.

Unless otherwise specified, the parameters given in *Table 58* are derived from tests performed under the ambient temperature and V_{DD} supply voltage conditions summarized in *Table 17*.

Table 58. I/O AC characteristics(1)(2)

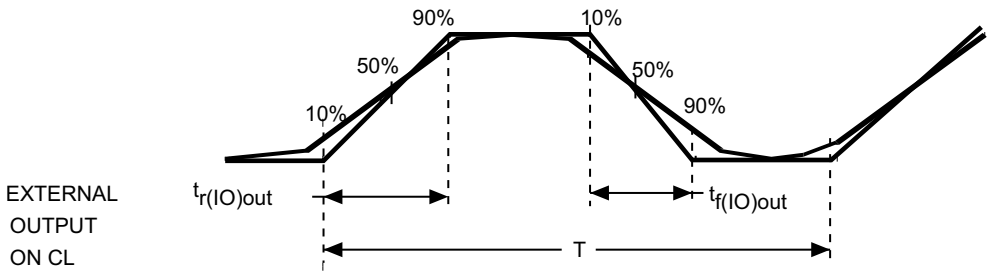
OSPEEDRy [1:0] bit value(1)	Symbol	Parameter	Conditions	Min	Typ	Max	Unit
00	$f_{max(IO)out}$	Maximum frequency(3)	C_L = 50 pF, V_{DD} ≥ 2.7 V	-	-	4	MHz
			C_L = 50 pF, V_{DD} ≥ 1.7 V	-	-	2	
			C_L = 10 pF, V_{DD} ≥ 2.7 V	-	-	8	
			C_L = 10 pF, V_{DD} ≥ 1.8 V	-	-	4	
			C_L = 10 pF, V_{DD} ≥ 1.7 V	-	-	3	
	$t_{f(IO)out}$/ $t_{r(IO)out}$	Output high to low level fall time and output low to high level rise time	C_L = 50 pF, V_{DD} = 1.7 V to 3.6 V	-	-	100	ns
01	$f_{max(IO)out}$	Maximum frequency(3)	C_L = 50 pF, V_{DD}≥ 2.7 V	-	-	25	MHz
			C_L = 50 pF, V_{DD}≥ 1.8 V	-	-	12.5	
			C_L = 50 pF, V_{DD}≥ 1.7 V	-	-	10	
			C_L = 10 pF, V_{DD} ≥ 2.7 V	-	-	50	
			C_L = 10 pF, V_{DD}≥ 1.8 V	-	-	20	
			C_L = 10 pF, V_{DD}≥ 1.7 V	-	-	12.5	
	$t_{f(IO)out}$/ $t_{r(IO)out}$	Output high to low level fall time and output low to high level rise time	C_L = 50 pF, V_{DD} ≥ 2.7 V	-	-	10	ns
			C_L = 10 pF, V_{DD} ≥ 2.7 V	-	-	6	
			C_L = 50 pF, V_{DD} ≥ 1.7 V	-	-	20	
			C_L = 10 pF, V_{DD} ≥ 1.7 V	-	-	10	
10	$f_{max(IO)out}$	Maximum frequency(3)	C_L = 40 pF, V_{DD} ≥ 2.7 V	-	-	50(4)	MHz
			C_L = 10 pF, V_{DD} ≥ 2.7 V	-	-	100(4)	
			C_L = 40 pF, V_{DD} ≥ 1.7 V	-	-	25	
			C_L = 10 pF, V_{DD} ≥ 1.8 V	-	-	50	
			C_L = 10 pF, V_{DD} ≥ 1.7 V	-	-	42.5	
	$t_{f(IO)out}$/ $t_{r(IO)out}$	Output high to low level fall time and output low to high level rise time	C_L = 40 pF, V_{DD} ≥2.7 V	-	-	6	ns
			C_L = 10 pF, V_{DD} ≥ 2.7 V	-	-	4	
			C_L = 40 pF, V_{DD} ≥ 1.7 V	-	-	10	
			C_L = 10 pF, V_{DD} ≥ 1.7 V	-	-	6	

Table 58. I/O AC characteristics[1][2] (continued)

OSPEEDRy [1:0] bit value[1]	Symbol	Parameter	Conditions	Min	Typ	Max	Unit
11	$f_{max(IO)out}$	Maximum frequency[3]	C_L = 30 pF, V_{DD} ≥ 2.7 V	-	-	100[4]	MHz
			C_L = 30 pF, V_{DD} ≥ 1.8 V	-	-	50	
			C_L = 30 pF, V_{DD} ≥ 1.7 V	-	-	42.5	
			C_L = 10 pF, V_{DD}≥ 2.7 V	-	-	180[4]	
			C_L = 10 pF, V_{DD} ≥ 1.8 V	-	-	100	
			C_L = 10 pF, V_{DD} ≥ 1.7 V	-	-	72.5	
	$t_{f(IO)out}$/ $t_{r(IO)out}$	Output high to low level fall time and output low to high level rise time	C_L = 30 pF, V_{DD} ≥ 2.7 V	-	-	4	ns
			C_L = 30 pF, V_{DD} ≥1.8 V	-	-	6	
			C_L = 30 pF, V_{DD} ≥1.7 V	-	-	7	
			C_L = 10 pF, V_{DD} ≥ 2.7 V	-	-	2.5	
			C_L = 10 pF, V_{DD} ≥1.8 V	-	-	3.5	
			C_L = 10 pF, V_{DD} ≥1.7 V	-	-	4	
-	tEXTIpw	Pulse width of external signals detected by the EXTI controller	-	10	-	-	ns

1. Guaranteed by design.
2. The I/O speed is configured using the OSPEEDRy[1:0] bits. Refer to the STM32F4xx reference manual for a description of the GPIOx_SPEEDR GPIO port output speed register.
3. The maximum frequency is defined in *Figure 36*.
4. For maximum frequencies above 50 MHz and V_{DD} > 2.4 V, the compensation cell should be used.

Figure 36. I/O AC characteristics definition



Maximum frequency is achieved if (t_r + t_f) ≤ (2/3)T and if the duty cycle is (45-55%) when loaded by CL specified in the table " *I/O AC characteristics*".

ai14131d

6.3.18 NRST pin characteristics

The NRST pin input driver uses CMOS technology. It is connected to a permanent pull-up resistor, R_{PU} (see *Table 56: I/O static characteristics*).

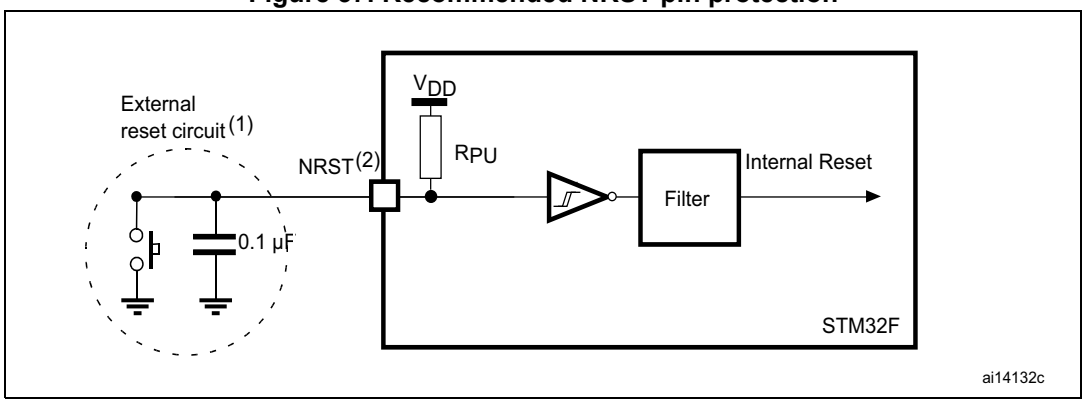
Unless otherwise specified, the parameters given in *Table 59* are derived from tests performed under the ambient temperature and V_{DD} supply voltage conditions summarized in *Table 17*.

Table 59. NRST pin characteristics

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
R_{PU}	Weak pull-up equivalent resistor(1)	$V_{IN} = V_{SS}$	30	40	50	kΩ
$V_{F(NRST)}$(2)	NRST Input filtered pulse		-	-	100	ns
$V_{NF(NRST)}$(2)	NRST Input not filtered pulse	$V_{DD} > 2.7$ V	300	-	-	ns
T_{NRST_OUT}	Generated reset pulse duration	Internal Reset source	20	-	-	µs

1. The pull-up is designed with a true resistance in series with a switchable PMOS. This PMOS contribution to the series resistance must be minimum (~10% order).
2. Guaranteed by design.

Figure 37. Recommended NRST pin protection



1. The reset network protects the device against parasitic resets.
2. The external capacitor must be placed as close as possible to the device.
3. The user must ensure that the level on the NRST pin can go below the $V_{IL(NRST)}$ max level specified in *Table 59*. Otherwise the reset is not taken into account by the device.

6.3.19 TIM timer characteristics

The parameters given in *Table 60* are guaranteed by design.

Refer to *Section 6.3.17: I/O port characteristics* for details on the input/output alternate function characteristics (output compare, input capture, external clock, PWM output).

Table 60. TIMx characteristics(1)(2)

Symbol	Parameter	Conditions(3)	Min	Max	Unit
$t_{res(TIM)}$	Timer resolution time	AHB/APBx prescaler=1 or 2 or 4, $f_{TIMxCLK}$ = 180 MHz	1	-	$t_{TIMxCLK}$
		AHB/APBx prescaler>4, $f_{TIMxCLK}$ = 90 MHz	1	-	$t_{TIMxCLK}$
f_{EXT}	Timer external clock frequency on CH1 to CH4	$f_{TIMxCLK}$ = 180 MHz	0	$f_{TIMxCLK}/2$	MHz
Res_{TIM}	Timer resolution		-	16/32	bit
t_{MAX_COUNT}	Maximum possible count with 32-bit counter		-	65536 × 65536	$t_{TIMxCLK}$

1. TIMx is used as a general term to refer to the TIM1 to TIM12 timers.
2. Guaranteed by design.
3. The maximum timer frequency on APB1 or APB2 is up to 180 MHz, by setting the TIMPRE bit in the RCC_DCKCFGR register, if APBx prescaler is 1 or 2 or 4, then TIMxCLK = HCKL, otherwise TIMxCLK = 4x PCLKx.

6.3.20 Communications interfaces

I^2C interface characteristics

The I^2C interface meets the timings requirements of the I^2C-bus specification and user manual rev. 03 for:

- Standard-mode (Sm): with a bit rate up to 100 kbit/s
- Fast-mode (Fm): with a bit rate up to 400 kbit/s.

The I^2C timings requirements are guaranteed by design when the I2C peripheral is properly configured (refer to RM0090 reference manual).

The SDA and SCL I/O requirements are met with the following restrictions: the SDA and SCL I/O pins are not “true” open-drain. When configured as open-drain, the PMOS connected between the I/O pin and V_{DD} is disabled, but is still present. Refer to *Section 6.3.17: I/O port characteristics* for more details on the I^2C I/O characteristics.

All I^2C SDA and SCL I/Os embed an analog filter. Refer to the table below for the analog filter characteristics:

Table 61. I2C analog filter characteristics(1)

Symbol	Parameter	Min	Max	Unit
t_{AF}	Maximum pulse width of spikes that are suppressed by the analog filter	50(2)	260(3)	ns

1. Guaranteed by design.
2. Spikes with widths below $t_{AF(min)}$ are filtered.
3. Spikes with widths above $t_{AF(max)}$ are not filtered

SPI interface characteristics

Unless otherwise specified, the parameters given in *Table 62* for the SPI interface are derived from tests performed under the ambient temperature, f_{PCLKx} frequency and V_{DD} supply voltage conditions summarized in *Table 17*, with the following configuration:

- Output speed is set to OSPEEDRy[1:0] = 10
- Capacitive load C = 30 pF
- Measurement points are done at CMOS levels: $0.5V_{DD}$

Refer to *Section 6.3.17: I/O port characteristics* for more details on the input/output alternate function characteristics (NSS, SCK, MOSI, MISO for SPI).

Table 62. SPI dynamic characteristics(1)

Symbol	Parameter	Conditions		Min	Typ	Max	Unit
f_{SCK} $1/t_{c(SCK)}$	SPI clock frequency	Master mode, SPI1/4/5/6, 2.7 V≤V_{DD}≤3.6 V		-	-	45	MHz
		Slave mode, SPI1/4/5/6, 2.7 V≤V_{DD}≤3.6 V	Receiver			45	
			Transmitter/ full-duplex			38(2)	
		Master mode, SPI1/2/3/4/5/6, 1.7 V≤V_{DD}≤3.6 V		-	-	22.5	
		Slave mode, SPI1/2/3/4/5/6, 1.7 V≤V_{DD}≤3.6 V				22.5	
Duty(SCK)	Duty cycle of SPI clock frequency	Slave mode		30	50	70	%

Table 62. SPI dynamic characteristics(1) (continued)

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
$t_{w(SCKH)}$	SCK high and low time	Master mode, SPI presc = 2, 2.7 V≤V_{DD}≤3.6 V	T_{PCLK} – 0.5	T_{PCLK}	T_{PCLK}+0.5	ns
$t_{w(SCKL)}$		Master mode, SPI presc = 2, 1.7 V≤V_{DD}≤3.6 V	T_{PCLK} – 2	T_{PCLK}	T_{PCLK}+2	
$t_{su(NSS)}$	NSS setup time	Slave mode, SPI presc = 2	$4T_{PCLK}$	-	-	
$t_{h(NSS)}$	NSS hold time	Slave mode, SPI presc = 2	$2T_{PCLK}$			
$t_{su(MI)}$	Data input setup time	Master mode	3	-	-	
$t_{su(SI)}$		Slave mode	0	-	-	
$t_{h(MI)}$	Data input hold time	Master mode	0.5	-	-	
$t_{h(SI)}$		Slave mode	2	-	-	
$t_{a(SO)}$	Data output access time	Slave mode, SPI presc = 2	0	-	$4T_{PCLK}$	
$t_{dis(SO)}$	Data output disable time	Slave mode, SPI1/4/5/6, 2.7 V≤V_{DD}≤3.6 V	0	-	8.5	
		Slave mode, SPI1/2/3/4/5/6 and 1.7 V≤V_{DD}≤3.6 V	0	-	16.5	
$t_{v(SO)}$ $t_{h(SO)}$	Data output valid/hold time	Slave mode (after enable edge), SPI1/4/5/6 and 2.7V ≤ V_{DD} ≤ 3.6V	-	11	13	ns
		Slave mode (after enable edge), SPI2/3, 2.7 V≤V_{DD}≤3.6 V	-	14	15	
		Slave mode (after enable edge), SPI1/4/5/6, 1.7 V≤V_{DD}≤3.6 V	-	15.5	19	
		Slave mode (after enable edge), SPI2/3, 1.7 V≤V_{DD}≤3.6 V	-	15.5	17.5	
$t_{v(MO)}$	Data output valid time	Master mode (after enable edge), SPI1/4/5/6, 2.7 V≤V_{DD}≤3.6 V	-	-	2.5	
		Master mode (after enable edge), SPI1/2/3/4/5/6, 1.7 V≤V_{DD}≤3.6 V	-	-	4.5	
$t_{h(MO)}$	Data output hold time	Master mode (after enable edge)	0	-	-	

1. Guaranteed by characterization results.
2. Maximum frequency in Slave transmitter mode is determined by the sum of $t_{v(SO)}$ and $t_{su(MI)}$ which has to fit into SCK low or high phase preceding the SCK sampling edge. This value can be achieved when the SPI communicates with a master having $t_{su(MI)}$ = 0 while Duty(SCK) = 50%

Figure 38. SPI timing diagram - slave mode and CPHA = 0

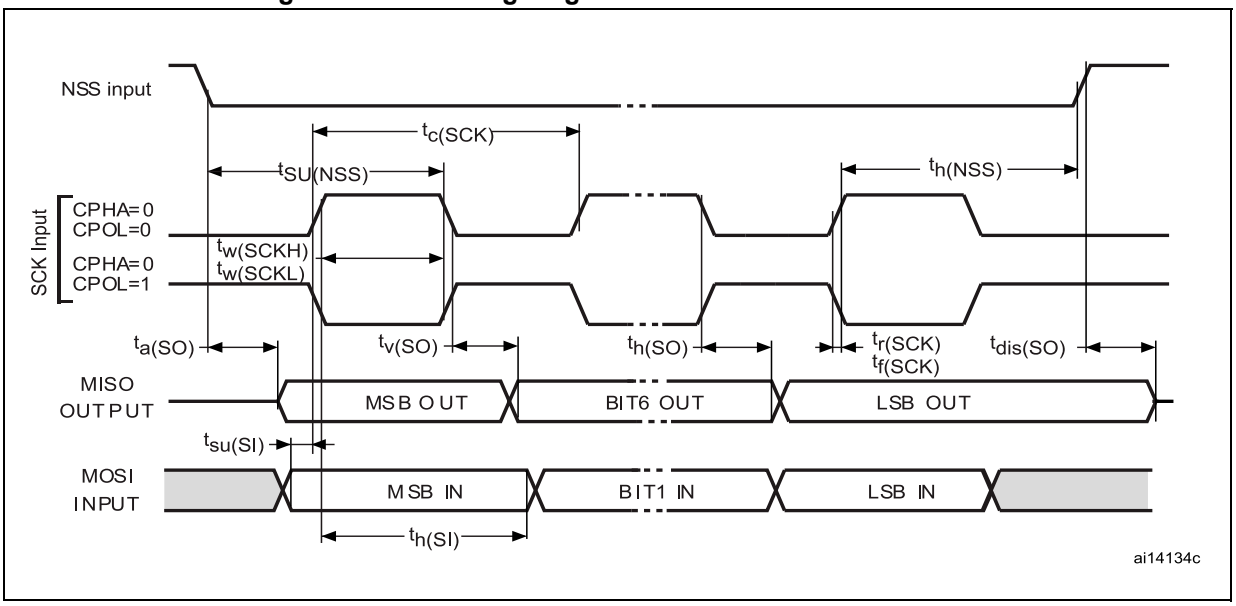


Figure 39. SPI timing diagram - slave mode and CPHA = 1

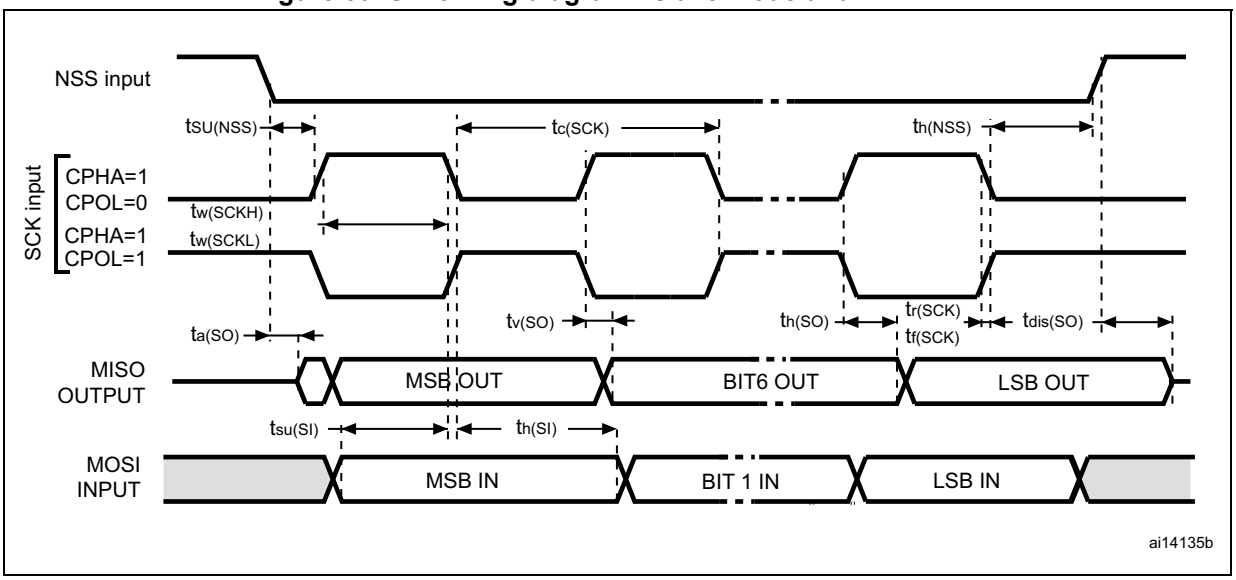
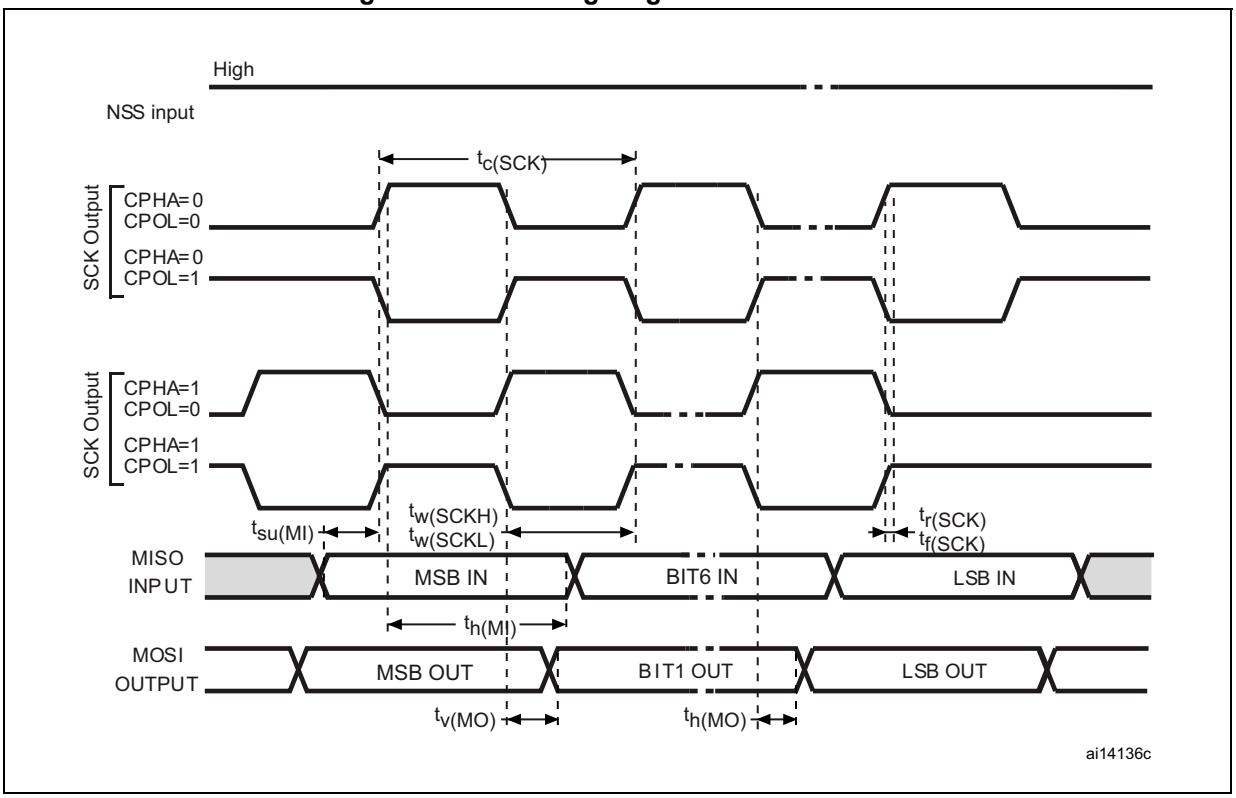


Figure 40. SPI timing diagram - master mode



I²S interface characteristics

Unless otherwise specified, the parameters given in *Table 63* for the I²S interface are derived from tests performed under the ambient temperature, f_{PCLKx} frequency and V_{DD} supply voltage conditions summarized in *Table 17*, with the following configuration:

- Output speed is set to OSPEEDRy[1:0] = 10
- Capacitive load C = 30 pF
- Measurement points are done at CMOS levels: $0.5V_{DD}$

Refer to *Section 6.3.17: I/O port characteristics* for more details on the input/output alternate function characteristics (CK, SD, WS).

Table 63. I²S dynamic characteristics(1)

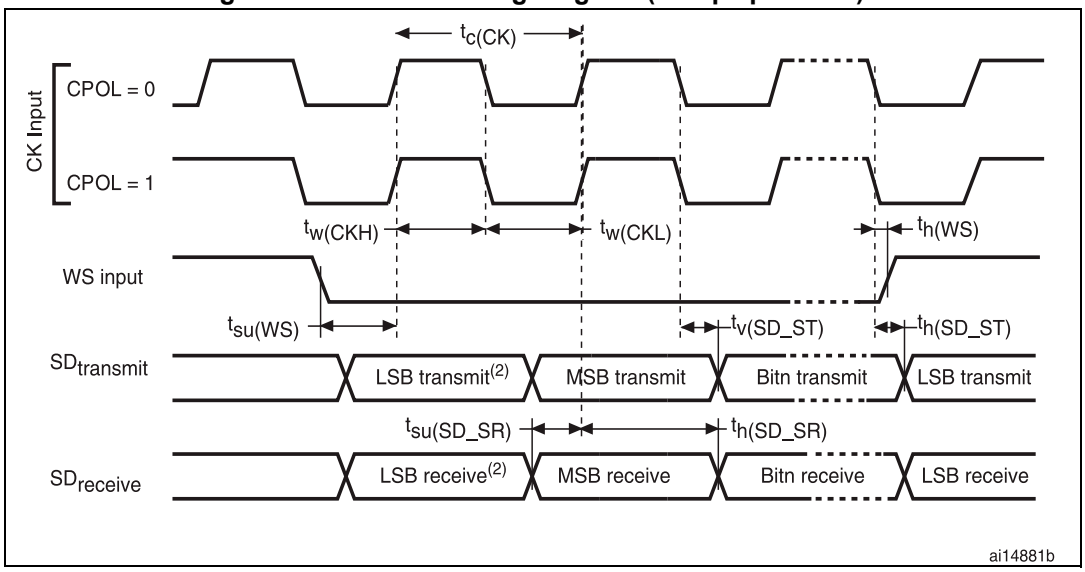
Symbol	Parameter	Conditions	Min	Max	Unit
f_{MCK}	I2S Main clock output	-	256x8K	256xFs(2)	MHz
f_{CK}	I2S clock frequency	Master data: 32 bits	-	64xFs	MHz
		Slave data: 32 bits	-	64xFs	
D_{CK}	I2S clock frequency duty cycle	Slave receiver	30	70	%
$t_{v(WS)}$	WS valid time	Master mode	0	6	ns
$t_{h(WS)}$	WS hold time	Master mode	0	-	
$t_{su(WS)}$	WS setup time	Slave mode	1	-	
$t_{h(WS)}$	WS hold time	Slave mode	0	-	
$t_{su(SD_MR)}$	Data input setup time	Master receiver	7.5	-	
$t_{su(SD_SR)}$		Slave receiver	2	-	
$t_{h(SD_MR)}$	Data input hold time	Master receiver	0	-	
$t_{h(SD_SR)}$		Slave receiver	0	-	
$t_{v(SD_ST)}$ $t_{h(SD_ST)}$	Data output valid time	Slave transmitter (after enable edge)	-	27	
$t_{v(SD_MT)}$		Master transmitter (after enable edge)	-	20	
$t_{h(SD_MT)}$	Data output hold time	Master transmitter (after enable edge)	2.5	-	

1. Guaranteed by characterization results.
2. The maximum value of 256xFs is 45 MHz (APB1 maximum frequency).

Note: Refer to the I2S section of RM0090 reference manual for more details on the sampling frequency (F_S).

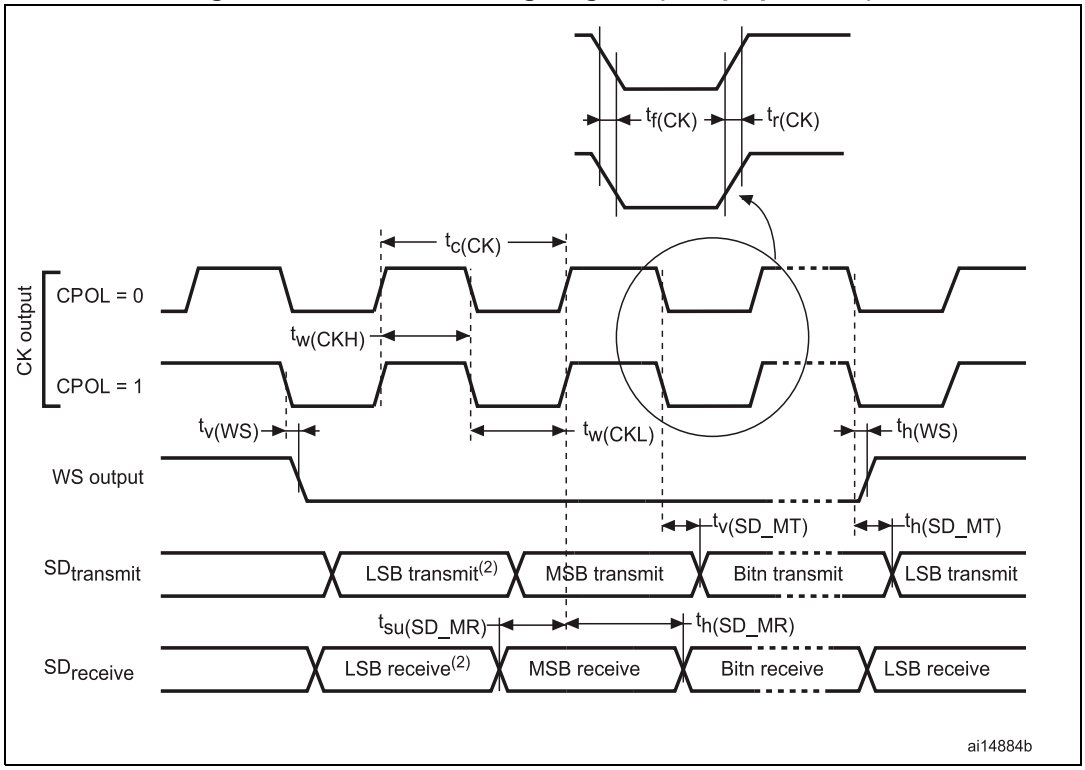
*f_{MCK}, f_{CK}, and D_{CK} values reflect only the digital peripheral behavior. The values of these parameters might be slightly impacted by the source clock precision. D_{CK} depends mainly on the value of ODD bit. The digital contribution leads to a minimum value of (I2SDIV/(2*I2SDIV+ODD) and a maximum value of (I2SDIV+ODD)/(2*I2SDIV+ODD). F_S maximum value is supported for each mode/condition.*

Figure 41. I²S slave timing diagram (Philips protocol)(1)



1. .LSB transmit/receive of the previously transmitted byte. No LSB transmit/receive is sent before the first byte.

Figure 42. I²S master timing diagram (Philips protocol)(1)



1. LSB transmit/receive of the previously transmitted byte. No LSB transmit/receive is sent before the first byte.

SAI characteristics

Unless otherwise specified, the parameters given in *Table 64* for SAI are derived from tests performed under the ambient temperature, f_{PCLKx} frequency and VDD supply voltage conditions summarized in *Table 17*, with the following configuration:

- Output speed is set to OSPEEDRy[1:0] = 10
- Capacitive load C=30 pF
- Measurement points are performed at CMOS levels: $0.5V_{DD}$

Refer to *Section 6.3.17: I/O port characteristics* for more details on the input/output alternate function characteristics (SCK,SD,WS).

Table 64. SAI characteristics(1)

Symbol	Parameter	Conditions	Min	Max	Unit
f_{MCKL}	SAI Main clock output	-	256 x 8K	256xFs(2)	MHz
F_{SCK}	SAI clock frequency	Master data: 32 bits	-	64xFs	MHz
		Slave data: 32 bits	-	64xFs	
D_{SCK}	SAI clock frequency duty cycle	Slave receiver	30	70	%
$t_{v(FS)}$	FS valid time	Master mode	8	22	ns
$t_{su(FS)}$	FS setup time	Slave mode	2	-	
$t_{h(FS)}$	FS hold time	Master mode	8	-	
		Slave mode	0	-	
$t_{su(SD_MR)}$	Data input setup time	Master receiver	5	-	
$t_{su(SD_SR)}$		Slave receiver	3	-	
$t_{h(SD_MR)}$	Data input hold time	Master receiver	0	-	
$t_{h(SD_SR)}$		Slave receiver	0	-	
$t_{v(SD_ST)}$ $t_{h(SD_ST)}$	Data output valid time	Slave transmitter (after enable edge)	-	22	
$t_{v(SD_MT)}$		Master transmitter (after enable edge)	-	20	
$t_{h(SD_MT)}$	Data output hold time	Master transmitter (after enable edge)	8	-	

1. Guaranteed by characterization results.
2. 256xFs maximum corresponds to 45 MHz (APB2 xaximum frequency)

Figure 43. SAI master timing waveforms

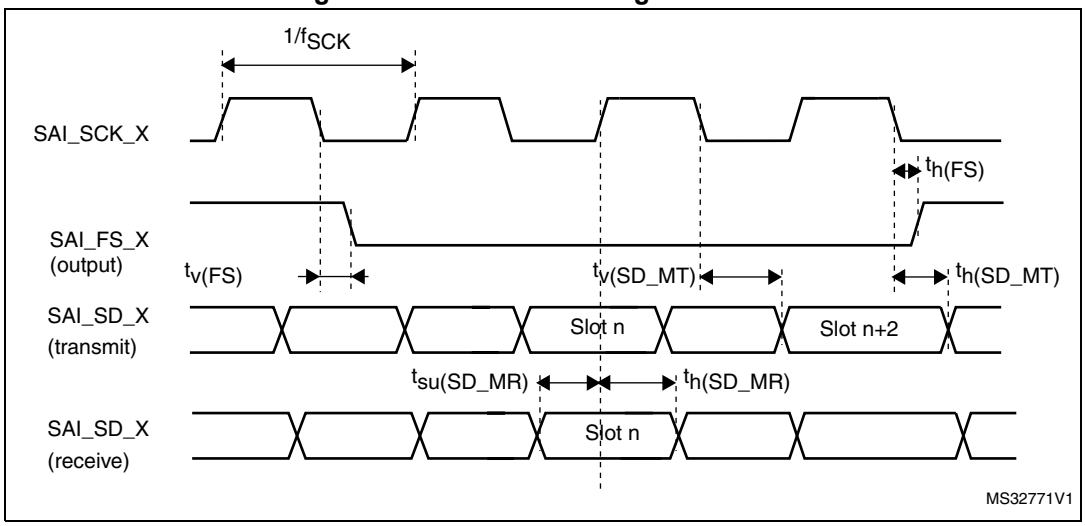
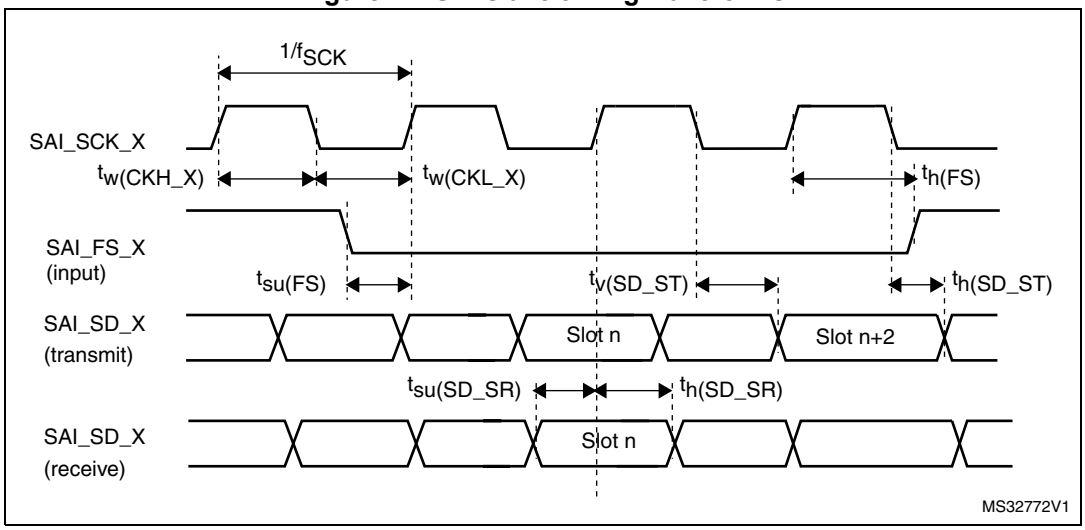


Figure 44. SAI slave timing waveforms



USB OTG full speed (FS) characteristics

This interface is present in both the USB OTG HS and USB OTG FS controllers.

Table 65. USB OTG full speed startup time

Symbol	Parameter	Max	Unit
$t_{STARTUP}$(1)	USB OTG full speed transceiver startup time	1	µs

1. Guaranteed by design.

Table 66. USB OTG full speed DC electrical characteristics

Symbol		Parameter	Conditions	Min.(1)	Typ.	Max.(1)	Unit
Input levels	V_{DD}	USB OTG full speed transceiver operating voltage		3.0(2)	-	3.6	V
	V_{DI}(3)	Differential input sensitivity	I(USB_FS_DP/DM, USB_HS_DP/DM)	0.2	-	-	V
	V_{CM}(3)	Differential common mode range	Includes V_{DI} range	0.8	-	2.5	
	V_{SE}(3)	Single ended receiver threshold		1.3	-	2.0	
Output levels	V_{OL}	Static output level low	R_L of 1.5 kΩ to 3.6 V(4)	-	-	0.3	V
	V_{OH}	Static output level high	R_L of 15 kΩ to V_{SS}(4)	2.8	-	3.6	
R_{PD}		PA11, PA12, PB14, PB15 (USB_FS_DP/DM, USB_HS_DP/DM)	$V_{IN} = V_{DD}$	17	21	24	kΩ
		PA9, PB13 (OTG_FS_VBUS, OTG_HS_VBUS)		0.65	1.1	2.0	
R_{PU}		PA12, PB15 (USB_FS_DP, USB_HS_DP)	$V_{IN} = V_{SS}$	1.5	1.8	2.1	
		PA9, PB13 (OTG_FS_VBUS, OTG_HS_VBUS)	$V_{IN} = V_{SS}$	0.25	0.37	0.55	

1. All the voltages are measured from the local ground potential.
2. The USB OTG full speed transceiver functionality is ensured down to 2.7 V but not the full USB full speed electrical characteristics which are degraded in the 2.7-to-3.0 V V_{DD} voltage range.
3. Guaranteed by design.
4. R_L is the load connected on the USB OTG full speed drivers.

Note: When VBUS sensing feature is enabled, PA9 and PB13 should be left at their default state (floating input), not as alternate function. A typical 200 µA current consumption of the sensing block (current to voltage conversion to determine the different sessions) can be observed on PA9 and PB13 when the feature is enabled.

Figure 45. USB OTG full speed timings: definition of data signal rise and fall time

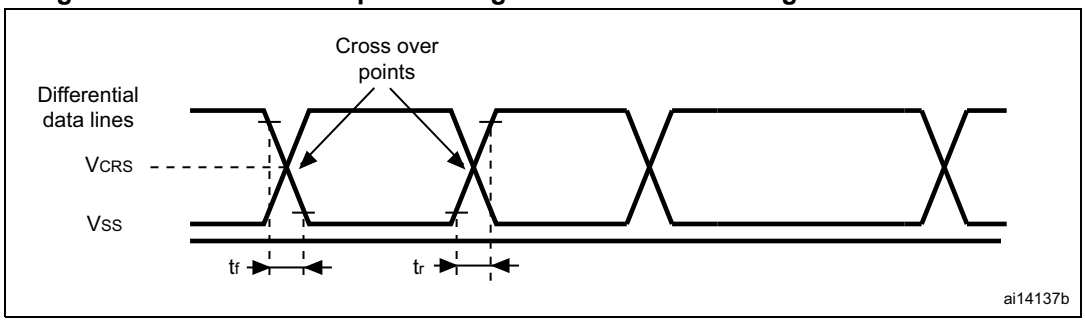


Table 67. USB OTG full speed electrical characteristics(1)

Driver characteristics					
Symbol	**Parameter**	**Conditions**	**Min**	**Max**	**Unit**
t_r	Rise time(2)	C_L = 50 pF	4	20	ns
t_f	Fall time(2)	C_L = 50 pF	4	20	ns
t_{rfm}	Rise/ fall time matching	t_r/t_f	90	110	%
V_{CRS}	Output signal crossover voltage		1.3	2.0	V
Z_{DRV}	Output driver impedance(3)	Driving high or low	28	44	Ω

1. Guaranteed by design.
2. Measured from 10% to 90% of the data signal. For more detailed informations, please refer to USB Specification - Chapter 7 (version 2.0).
3. No external termination series resistors are required on DP (D+) and DM (D-) pins since the matching impedance is included in the embedded driver.

USB high speed (HS) characteristics

Unless otherwise specified, the parameters given in *Table 70* for ULPI are derived from tests performed under the ambient temperature, f_{HCLK} frequency summarized in *Table 69* and V_{DD} supply voltage conditions summarized in *Table 68*, with the following configuration:

- Output speed is set to OSPEEDRy[1:0] = 10, unless otherwise specified
- Capacitive load C = 30 pF, unless otherwise specified
- Measurement points are done at CMOS levels: $0.5V_{DD}$.

Refer to *Section 6.3.17: I/O port characteristics* for more details on the input/output characteristics.

Table 68. USB HS DC electrical characteristics

Symbol		Parameter	Min.(1)	Max.(1)	Unit
Input level	V_{DD}	USB OTG HS operating voltage	1.7	3.6	V

1. All the voltages are measured from the local ground potential.

Table 69. USB HS clock timing parameters(1)

Symbol	Parameter		Min	Typ	Max	Unit
	f_{HCLK} value to guarantee proper operation of USB HS interface		30	-	-	MHz
F_{START_8BIT}	Frequency (first transition)	8-bit ±10%	54	60	66	MHz
F_{STEADY}	Frequency (steady state) ±500 ppm		59.97	60	60.03	MHz
D_{START_8BIT}	Duty cycle (first transition)	8-bit ±10%	40	50	60	%
D_{STEADY}	Duty cycle (steady state) ±500 ppm		49.975	50	50.025	%
t_{STEADY}	Time to reach the steady state frequency and duty cycle after the first transition		-	-	1.4	ms
t_{START_DEV}	Clock startup time after the de-assertion of SuspendM	Peripheral	-	-	5.6	ms
t_{START_HOST}		Host	-	-	-	
t_{PREP}	PHY preparation time after the first transition of the input clock		-	-	-	µs

1. Guaranteed by design.

Figure 46. ULPI timing diagram

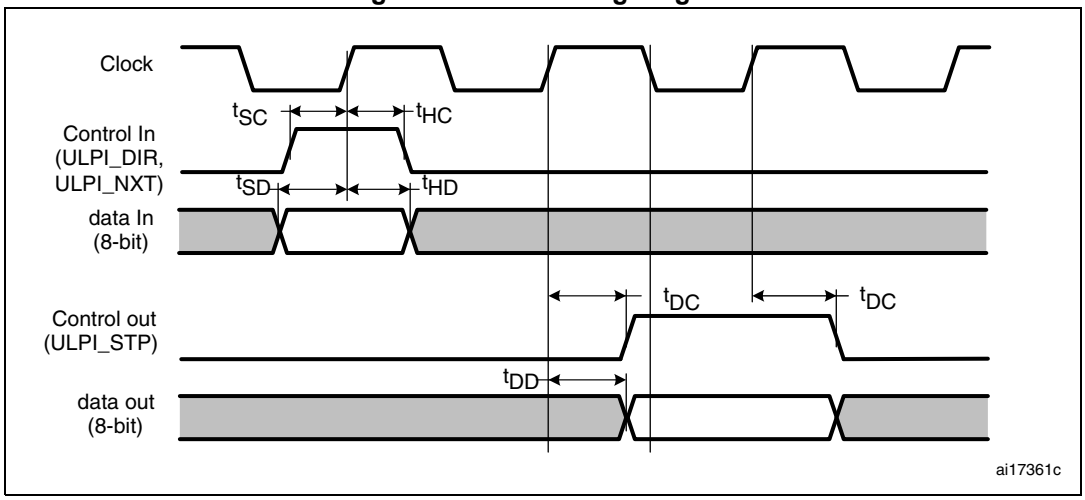


Table 70. Dynamic characteristics: USB ULPI(1)

Symbol	Parameter	Conditions	Min.	Typ.	Max.	Unit
t_{SC}	Control in (ULPI_DIR, ULPI_NXT) setup time		2	-	-	ns
t_{HC}	Control in (ULPI_DIR, ULPI_NXT) hold time		0.5	-	-	
t_{SD}	Data in setup time		1.5	-	-	
t_{HD}	Data in hold time		2	-	-	
t_{DC}/t_{DD}	Data/control output delay	2.7 V < V_{DD} < 3.6 V, C_L = 15 pF and OSPEEDRy[1:0] = 11	-	9	9.5	
		2.7 V < V_{DD} < 3.6 V, C_L = 20 pF and OSPEEDRy[1:0] = 10	-	12	15	
		1.7 V < V_{DD} < 3.6 V, C_L = 15 pF and OSPEEDRy[1:0] = 11	-			

1. Guaranteed by characterization results.

Ethernet characteristics

Unless otherwise specified, the parameters given in *Table 71*, *Table 72* and *Table 73* for SMI, RMII and MII are derived from tests performed under the ambient temperature, f_{HCLK} frequency summarized in *Table 17* with the following configuration:

- Output speed is set to OSPEEDRy[1:0] = 10
- Capacitive load C = 30 pF for 2.7 V < V_{DD} < 3.6 V
- Capacitive load C = 20 pF for 1.71 V < V_{DD} < 3.6 V
- Measurement points are done at CMOS levels: 0.5V_{DD}.

Refer to *Section 6.3.17: I/O port characteristics* for more details on the input/output characteristics.

Table 71 gives the list of Ethernet MAC signals for the SMI (station management interface) and *Figure 47* shows the corresponding timing diagram.

Figure 47. Ethernet SMI timing diagram

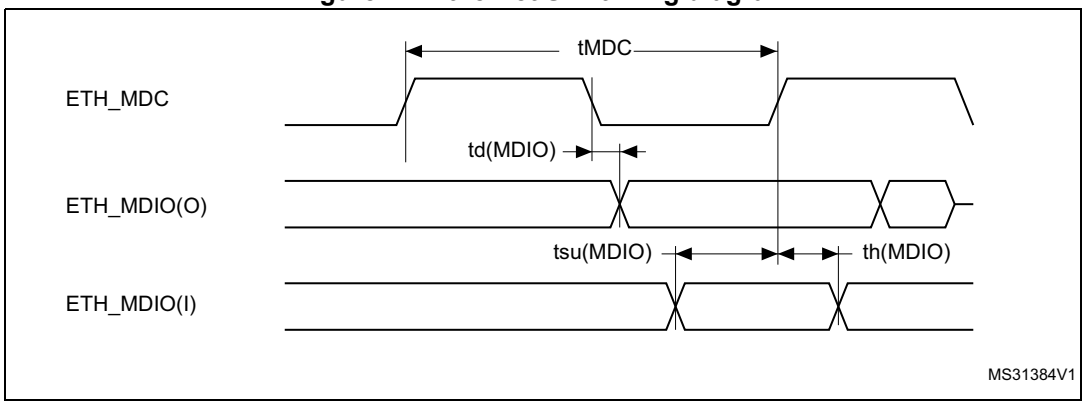


Table 71. Dynamics characteristics: Ethernet MAC signals for SMI(1)

Symbol	Parameter	Min	Typ	Max	Unit
t_{MDC}	MDC cycle time(2.38 MHz)	411	420	425	ns
$T_{d(MDIO)}$	Write data valid time	6	10	13	
$t_{su(MDIO)}$	Read data setup time	12	-	-	
$t_{h(MDIO)}$	Read data hold time	0	-	-	

1. Guaranteed by characterization results.

Table 72 gives the list of Ethernet MAC signals for the RMII and *Figure 48* shows the corresponding timing diagram.

Figure 48. Ethernet RMII timing diagram

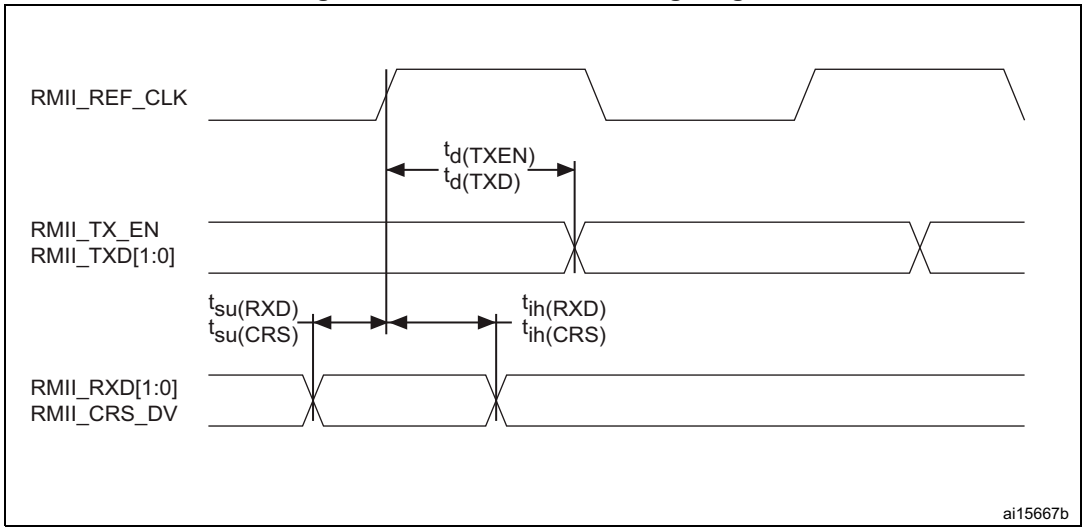


Table 72. Dynamics characteristics: Ethernet MAC signals for RMII(1)

Symbol	Parameter	Condition	Min	Typ	Max	Unit
$t_{su(RXD)}$	Receive data setup time	1.71 V < V_{DD} < 3.6 V	1.5	-	-	ns
$t_{ih(RXD)}$	Receive data hold time		0	-	-	
$t_{su(CRS)}$	Carrier sense setup time		1	-	-	
$t_{ih(CRS)}$	Carrier sense hold time		1	-	-	
$t_{d(TXEN)}$	Transmit enable valid delay time	2.7 V < V_{DD} < 3.6 V	8	10.5	12	
		1.71 V < V_{DD} < 3.6 V	8	10.5	14	
$t_{d(TXD)}$	Transmit data valid delay time	2.7 V < V_{DD} < 3.6 V	8	11	12.5	
		1.71 V < V_{DD} < 3.6 V	8	11	14.5	

1. Guaranteed by characterization results.

Table 73 gives the list of Ethernet MAC signals for MII and *Figure 48* shows the corresponding timing diagram.

Figure 49. Ethernet MII timing diagram

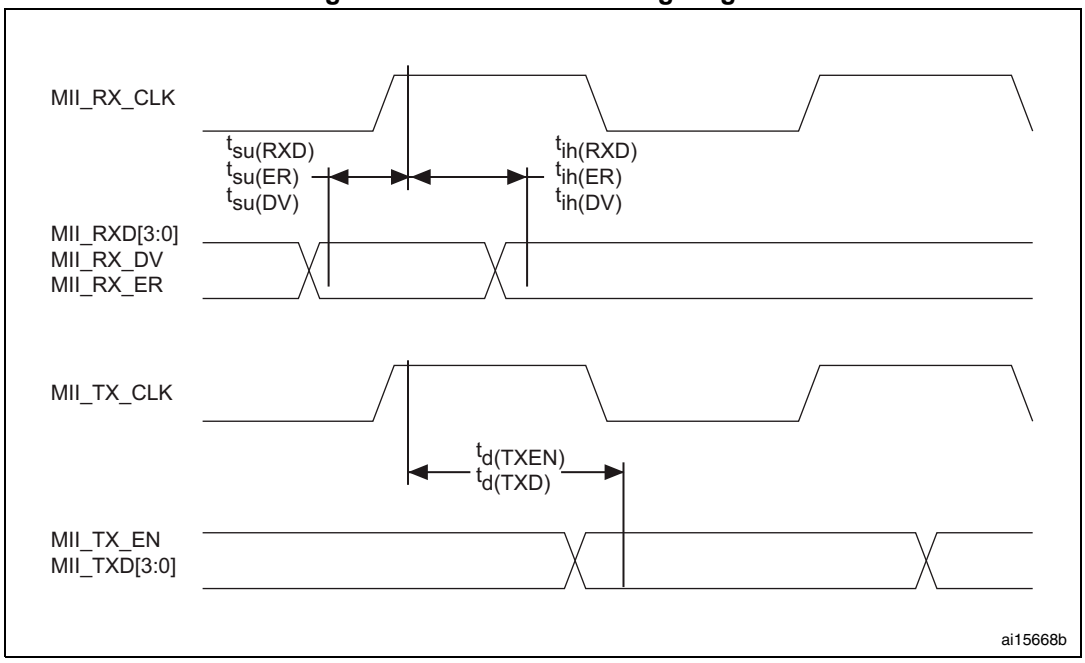


Table 73. Dynamics characteristics: Ethernet MAC signals for MII(1)

Symbol	Parameter	Condition	Min	Typ	Max	Unit
$t_{su(RXD)}$	Receive data setup time	1.71 V < V_{DD} < 3.6 V	9	-	-	ns
$t_{ih(RXD)}$	Receive data hold time		10	-	-	
$t_{su(DV)}$	Data valid setup time		9	-	-	
$t_{ih(DV)}$	Data valid hold time		8	-	-	
$t_{su(ER)}$	Error setup time		6	-	-	
$t_{ih(ER)}$	Error hold time		8	-	-	
$t_{d(TXEN)}$	Transmit enable valid delay time	2.7 V < V_{DD} < 3.6 V	8	10	14	
		1.71 V < V_{DD} < 3.6 V	8	10	16	
$t_{d(TXD)}$	Transmit data valid delay time	2.7 V < V_{DD} < 3.6 V	7.5	10	15	
		1.71 V < V_{DD} < 3.6 V	7.5	10	17	

1. Guaranteed by characterization results.

CAN (controller area network) interface

Refer to *Section 6.3.17: I/O port characteristics* for more details on the input/output alternate function characteristics (CANx_TX and CANx_RX).

6.3.21 12-bit ADC characteristics

Unless otherwise specified, the parameters given in *Table 74* are derived from tests performed under the ambient temperature, f_{PCLK2} frequency and V_{DDA} supply voltage conditions summarized in *Table 17*.

Table 74. ADC characteristics

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
V_{DDA}	Power supply	$V_{DDA} - V_{REF+} < 1.2$ V	1.7(1)	-	3.6	V
V_{REF+}	Positive reference voltage		1.7(1)	-	V_{DDA}	
V_{REF-}	Negative reference voltage	-	-	0	-	
f_{ADC}	ADC clock frequency	V_{DDA} = 1.7(1) to 2.4 V	0.6	15	18	MHz
		V_{DDA} = 2.4 to 3.6 V	0.6	30	36	MHz
f_{TRIG}(2)	External trigger frequency	f_{ADC} = 30 MHz, 12-bit resolution	-	-	1764	kHz
			-	-	17	$1/f_{ADC}$
V_{AIN}	Conversion voltage range(3)		0 (V_{SSA} or V_{REF-} tied to ground)	-	V_{REF+}	V
R_{AIN}(2)	External input impedance	See *Equation 1* for details	-	-	50	kΩ
R_{ADC}(2)(4)	Sampling switch resistance		1.5	-	6	kΩ
C_{ADC}(2)	Internal sample and hold capacitor		-	4	7	pF
t_{lat}(2)	Injection trigger conversion latency	f_{ADC} = 30 MHz	-	-	0.100	µs
			-	-	3(5)	$1/f_{ADC}$
t_{latr}(2)	Regular trigger conversion latency	f_{ADC} = 30 MHz	-	-	0.067	µs
			-	-	2(5)	$1/f_{ADC}$
t_S(2)	Sampling time	f_{ADC} = 30 MHz	0.100	-	16	µs
			3	-	480	$1/f_{ADC}$
t_{STAB}(2)	Power-up time		-	2	3	µs
t_{CONV}(2)	Total conversion time (including sampling time)	f_{ADC} = 30 MHz 12-bit resolution	0.50	-	16.40	µs
		f_{ADC} = 30 MHz 10-bit resolution	0.43	-	16.34	µs
		f_{ADC} = 30 MHz 8-bit resolution	0.37	-	16.27	µs
		f_{ADC} = 30 MHz 6-bit resolution	0.30	-	16.20	µs
		9 to 492 (t_S for sampling +n-bit resolution for successive approximation)				$1/f_{ADC}$

Table 74. ADC characteristics (continued)

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
f_S(2)	Sampling rate (f_{ADC} = 30 MHz, and t_S = 3 ADC cycles)	12-bit resolution Single ADC	-	-	2	Msps
		12-bit resolution Interleave Dual ADC mode	-	-	3.75	Msps
		12-bit resolution Interleave Triple ADC mode	-	-	6	Msps
I_{VREF+}(2)	ADC V_{REF} DC current consumption in conversion mode		-	300	500	µA
I_{VDDA}(2)	ADC V_{DDA} DC current consumption in conversion mode		-	1.6	1.8	mA

1. V_{DDA} minimum value of 1.7 V is obtained with the use of an external power supply supervisor (refer to *Section 3.17.2: Internal reset OFF*).
2. Guaranteed by characterization results.
3. V_{REF+} is internally connected to V_{DDA} and V_{REF-} is internally connected to V_{SSA}.
4. R_{ADC} maximum value is given for V_{DD}=1.7 V, and minimum value for V_{DD}=3.3 V.
5. For external triggers, a delay of $1/f_{PCLK2}$ must be added to the latency specified in *Table 74*.

Equation 1: R_{AIN} max formula

$$R_{AIN} = \frac{(k - 0.5)}{f_{ADC} \times C_{ADC} \times \ln(2^{N+2})} - R_{ADC}$$

The formula above (*Equation 1*) is used to determine the maximum external impedance allowed for an error below 1/4 of LSB. N = 12 (from 12-bit resolution) and k is the number of sampling periods defined in the ADC_SMPR1 register.

Table 75. ADC static accuracy at f_{ADC} = 18 MHz

Symbol	Parameter	Test conditions	Typ	Max(1)	Unit
ET	Total unadjusted error	f_{ADC} =18 MHz V_{DDA} = 1.7 to 3.6 V V_{REF} = 1.7 to 3.6 V $V_{DDA} - V_{REF} < 1.2$ V	±3	±4	LSB
EO	Offset error		±2	±3	
EG	Gain error		±1	±3	
ED	Differential linearity error		±1	±2	
EL	Integral linearity error		±2	±3	

1. Guaranteed by characterization results.

Table 76. ADC static accuracy at f_{ADC} = 30 MHz

Symbol	Parameter	Test conditions	Typ	Max(1)	Unit
ET	Total unadjusted error	f_{ADC} = 30 MHz, R_{AIN} < 10 kΩ, V_{DDA} = 2.4 to 3.6 V, V_{REF} = 1.7 to 3.6 V, $V_{DDA} - V_{REF}$ < 1.2 V	±2	±5	LSB
EO	Offset error		±1.5	±2.5	
EG	Gain error		±1.5	±3	
ED	Differential linearity error		±1	±2	
EL	Integral linearity error		±1.5	±3	

1. Guaranteed by characterization results.

Table 77. ADC static accuracy at f_{ADC} = 36 MHz

Symbol	Parameter	Test conditions	Typ	Max(1)	Unit
ET	Total unadjusted error	f_{ADC} =36 MHz, V_{DDA} = 2.4 to 3.6 V, V_{REF} = 1.7 to 3.6 V, $V_{DDA} - V_{REF}$ < 1.2 V	±4	±7	LSB
EO	Offset error		±2	±3	
EG	Gain error		±3	±6	
ED	Differential linearity error		±2	±3	
EL	Integral linearity error		±3	±6	

1. Guaranteed by characterization results.

Table 78. ADC dynamic accuracy at f_{ADC} = 18 MHz - limited test conditions(1)

Symbol	Parameter	Test conditions	Min	Typ	Max	Unit
ENOB	Effective number of bits	f_{ADC} =18 MHz, $V_{DDA} = V_{REF+}$= 1.7 V, Input Frequency = 20 KHz, Temperature = 25 °C	10.3	10.4	-	bits
SINAD	Signal-to-noise and distortion ratio		64	64.2	-	dB
SNR	Signal-to-noise ratio		64	65	-	
THD	Total harmonic distortion		− 67	− 72	-	

1. Guaranteed by characterization results.

Table 79. ADC dynamic accuracy at f_{ADC} = 36 MHz - limited test conditions(1)

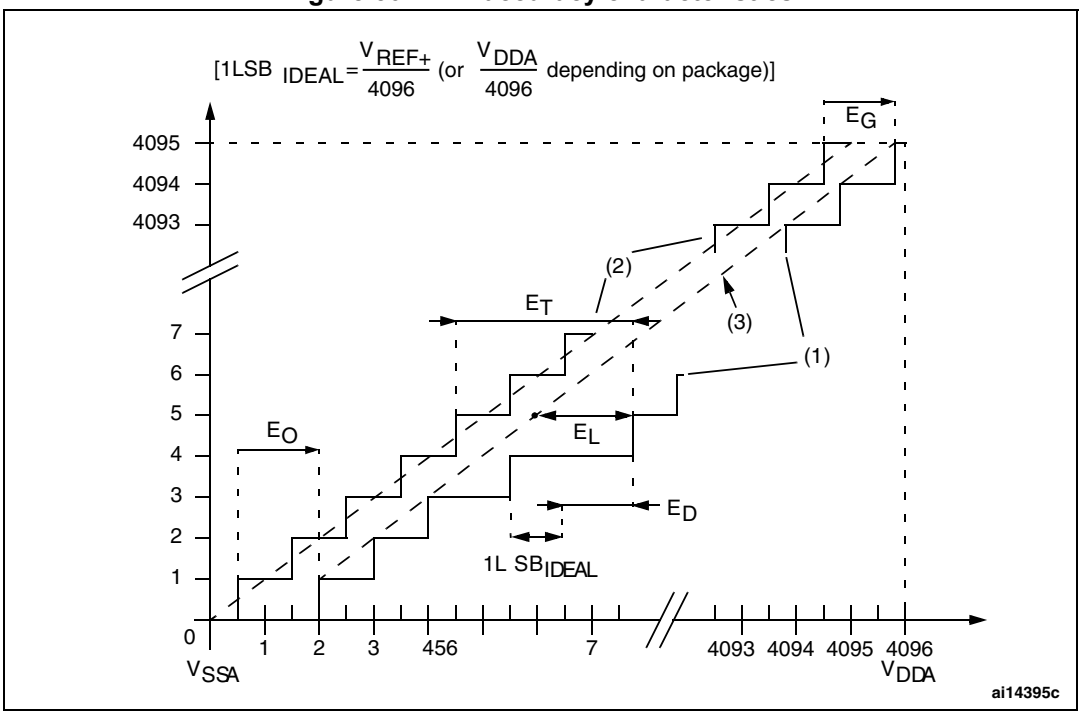
Symbol	Parameter	Test conditions	Min	Typ	Max	Unit
ENOB	Effective number of bits	f_{ADC} =36 MHz, $V_{DDA} = V_{REF+}$ = 3.3 V, Input Frequency = 20 KHz, Temperature = 25 °C	10.6	10.8	-	bits
SINAD	Signal-to noise and distortion ratio		66	67	-	dB
SNR	Signal-to noise ratio		64	68	-	
THD	Total harmonic distortion		− 70	− 72	-	

1. Guaranteed by characterization results.

Note: ADC accuracy vs. negative injection current: injecting a negative current on any analog input pins should be avoided as this significantly reduces the accuracy of the conversion being performed on another analog input. It is recommended to add a Schottky diode (pin to ground) to analog pins which may potentially inject negative currents.

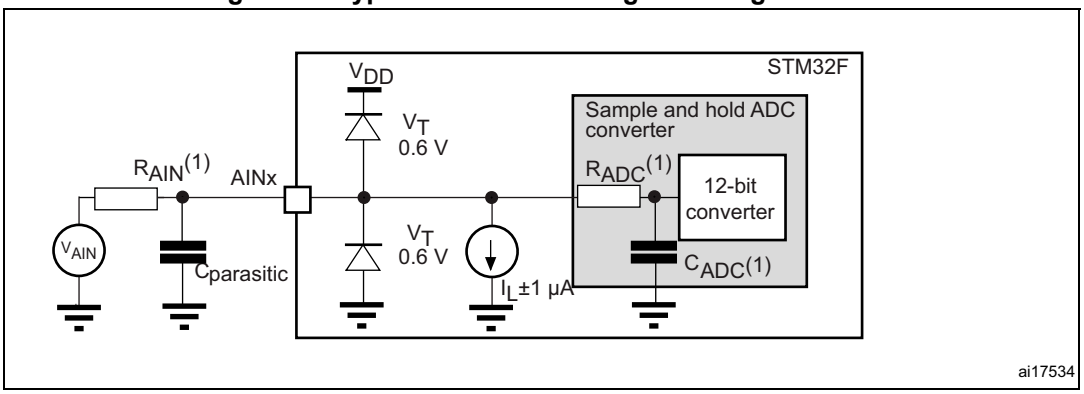
Any positive injection current within the limits specified for $I_{INJ(PIN)}$ and $\Sigma I_{INJ(PIN)}$ in *Section 6.3.17* does not affect the ADC accuracy.

Figure 50. ADC accuracy characteristics



1. See also *Table 76*.
2. Example of an actual transfer curve.
3. Ideal transfer curve.
4. End point correlation line.
5. E_T = Total Unadjusted Error: maximum deviation between the actual and the ideal transfer curves.
   EO = Offset Error: deviation between the first actual transition and the first ideal one.
   EG = Gain Error: deviation between the last ideal transition and the last actual one.
   ED = Differential Linearity Error: maximum deviation between actual steps and the ideal one.
   EL = Integral Linearity Error: maximum deviation between any actual transition and the end point correlation line.

Figure 51. Typical connection diagram using the ADC

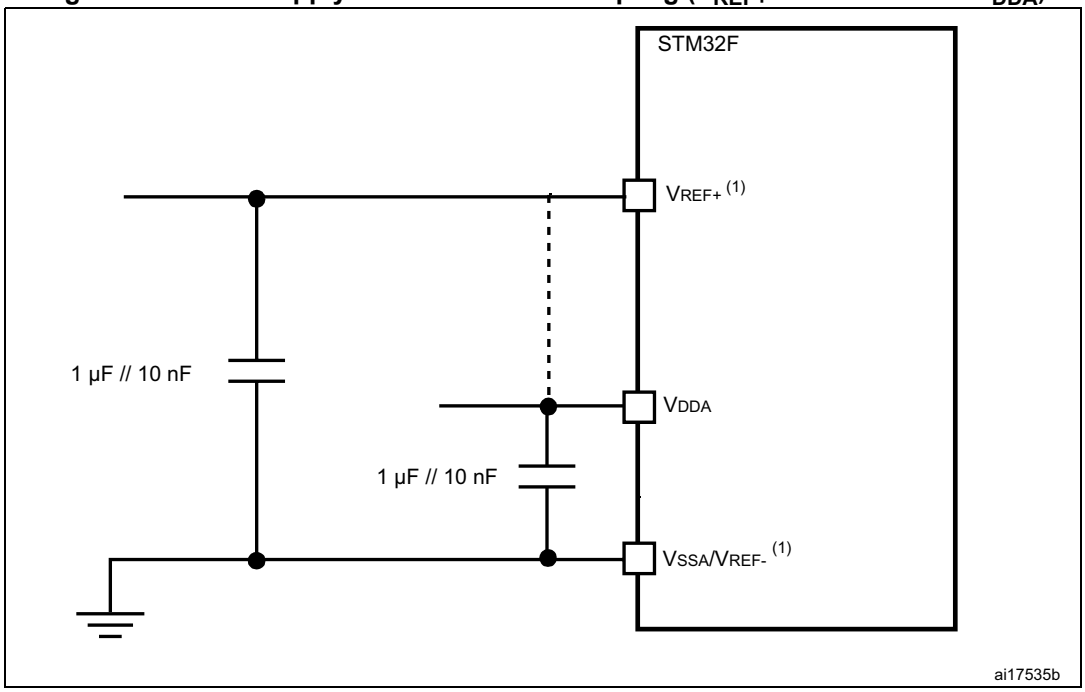


1. Refer to *Table 74* for the values of R_{AIN}, R_{ADC} and C_{ADC}.
2. $C_{parasitic}$ represents the capacitance of the PCB (dependent on soldering and PCB layout quality) plus the pad capacitance (roughly 5 pF). A high $C_{parasitic}$ value downgrades conversion accuracy. To remedy this, f_{ADC} should be reduced.

General PCB design guidelines

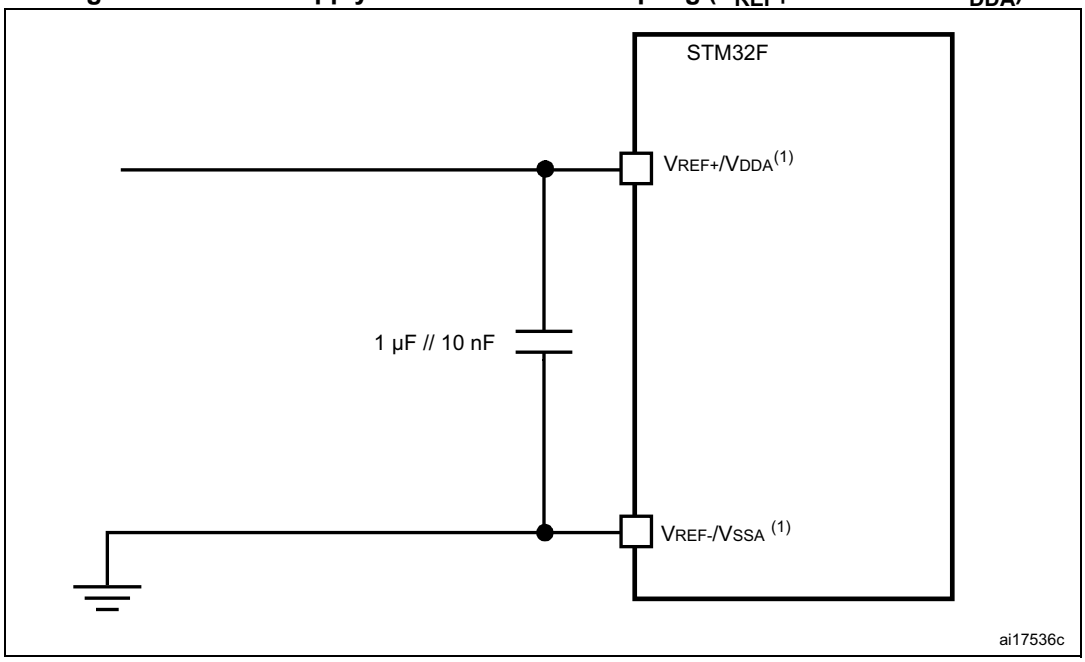
Power supply decoupling should be performed as shown in *Figure 52* or *Figure 53*, depending on whether V_{REF+} is connected to V_{DDA} or not. The 10 nF capacitors should be ceramic (good quality). They should be placed them as close as possible to the chip.

Figure 52. Power supply and reference decoupling (V_{REF+} not connected to V_{DDA})



1. V_{REF+} and V_{REF-} inputs are both available on UFBGA176. V_{REF+} is also available on LQFP100, LQFP144, and LQFP176. When V_{REF+} and V_{REF-} are not available, they are internally connected to V_{DDA} and V_{SSA}.

Figure 53. Power supply and reference decoupling (V_{REF+} connected to V_{DDA})



1. V_{REF+} and V_{REF-} inputs are both available on UFBGA176. V_{REF+} is also available on LQFP100, LQFP144, and LQFP176. When V_{REF+} and V_{REF-} are not available, they are internally connected to V_{DDA} and V_{SSA}.

6.3.22 Temperature sensor characteristics

Table 80. Temperature sensor characteristics

Symbol	Parameter	Min	Typ	Max	Unit
T_L(1)	V_{SENSE} linearity with temperature	-	±1	±2	°C
Avg_Slope(1)	Average slope	-	2.5		mV/°C
V_{25}(1)	Voltage at 25 °C	-	0.76		V
t_{START}(2)	Startup time	-	6	10	µs
T_{S_temp}(2)	ADC sampling time when reading the temperature (1 °C accuracy)	10	-	-	µs

1. Guaranteed by characterization results.
2. Guaranteed by design.

Table 81. Temperature sensor calibration values

Symbol	Parameter	Memory address
TS_CAL1	TS ADC raw data acquired at temperature of 30 °C, V_{DDA}= 3.3 V	0x1FFF 7A2C - 0x1FFF 7A2D
TS_CAL2	TS ADC raw data acquired at temperature of 110 °C, V_{DDA}= 3.3 V	0x1FFF 7A2E - 0x1FFF 7A2F

6.3.23 V_{BAT} monitoring characteristics

Table 82. V_{BAT} monitoring characteristics

Symbol	Parameter	Min	Typ	Max	Unit
R	Resistor bridge for V_{BAT}	-	50	-	KΩ
Q	Ratio on V_{BAT} measurement	-	4	-	
Er(1)	Error on Q	–1	-	+1	%
T_{S_vbat}(2)(2)	ADC sampling time when reading the V_{BAT} 1 mV accuracy	5	-	-	µs

1. Guaranteed by design.
2. Shortest sampling time can be determined in the application by multiple iterations.

6.3.24 Reference voltage

The parameters given in *Table 83* are derived from tests performed under ambient temperature and V_{DD} supply voltage conditions summarized in *Table 17*.

Table 83. internal reference voltage

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
V_{REFINT}	Internal reference voltage	–40 °C < T_A < +105 °C	1.18	1.21	1.24	V
$T_{S_vrefint}$(1)	ADC sampling time when reading the internal reference voltage		10	-	-	µs
V_{RERINT_s}(2)	Internal reference voltage spread over the temperature range	V_{DD} = 3V ± 10mV	-	3	5	mV
T_{Coeff}(2)	Temperature coefficient		-	30	50	ppm/°C
t_{START}(2)	Startup time		-	6	10	µs

1. Shortest sampling time can be determined in the application by multiple iterations.
2. Guaranteed by design, not tested in production

Table 84. Internal reference voltage calibration values

Symbol	Parameter	Memory address
V_{REFIN_CAL}	Raw data acquired at temperature of 30 °C $_{VDDA}$ = 3.3 V	0x1FFF 7A2A - 0x1FFF 7A2B

6.3.25 DAC electrical characteristics

Table 85. DAC characteristics

Symbol	Parameter	Conditions		Min	Typ	Max	Unit	Comments
V_{DDA}	Analog supply voltage	-		$1.7^{(1)}$	-	3.6	V	-
V_{REF+}	Reference supply voltage	-		$1.7^{(1)}$	-	3.6	V	$V_{REF+} \leq V_{DDA}$
V_{SSA}	Ground	-		0	-	0	V	-
$R_{LOAD}^{(2)}$	Resistive load	DAC output buffer ON	R_{LOAD} connected to V_{SSA}	5	-	-	kΩ	-
			R_{LOAD} connected to V_{DDA}	25				-
$R_O^{(2)}$	Impedance output with buffer OFF	-		-	-	15	kΩ	When the buffer is OFF, the Minimum resistive load between DAC_OUT and V_{SS} to have a 1% accuracy is 1.5 MΩ
$C_{LOAD}^{(2)}$	Capacitive load	-		-	-	50	pF	Maximum capacitive load at DAC_OUT pin (when the buffer is ON).
DAC_OUT min$^{(2)}$	Lower DAC_OUT voltage with buffer ON	-		0.2	-	-	V	It gives the maximum output excursion of the DAC. It corresponds to 12-bit input code (0x0E0) to (0xF1C) at V_{REF+} = 3.6 V and (0x1C7) to (0xE38) at V_{REF+} = 1.7 V
DAC_OUT max$^{(2)}$	Higher DAC_OUT voltage with buffer ON	-		-	-	V_{DDA} – 0.2	V	
DAC_OUT min$^{(2)}$	Lower DAC_OUT voltage with buffer OFF	-		-	0.5	-	mV	It gives the maximum output excursion of the DAC.
DAC_OUT max$^{(2)}$	Higher DAC_OUT voltage with buffer OFF	-		-	-	V_{REF+} – 1LSB	V	
$I_{VREF+}^{(4)}$	DAC DC V_{REF} current consumption in quiescent mode (Standby mode)	-		-	170	240	µA	With no load, worst code (0x800) at V_{REF+} = 3.6 V in terms of DC consumption on the inputs
		-		-	50	75		With no load, worst code (0xF1C) at V_{REF+} = 3.6 V in terms of DC consumption on the inputs

Table 85. DAC characteristics (continued)

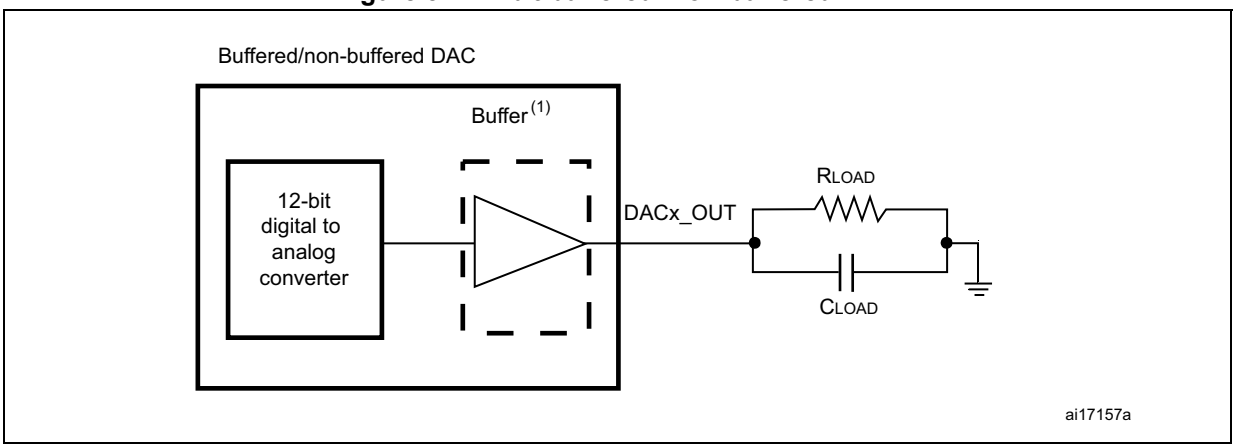
Symbol	Parameter	Conditions	Min	Typ	Max	Unit	Comments
I_{DDA}(4)	DAC DC VDDA current consumption in quiescent mode(3)	-	-	280	380	µA	With no load, middle code (0x800) on the inputs
		-	-	475	625	µA	With no load, worst code (0xF1C) at V_{REF+} = 3.6 V in terms of DC consumption on the inputs
DNL(4)	Differential non linearity Difference between two consecutive code-1LSB)	-	-	-	±0.5	LSB	Given for the DAC in 10-bit configuration.
		-	-	-	±2	LSB	Given for the DAC in 12-bit configuration.
INL(4)	Integral non linearity (difference between measured value at Code i and the value at Code i on a line drawn between Code 0 and last Code 1023)	-	-	-	±1	LSB	Given for the DAC in 10-bit configuration.
		-	-	-	±4	LSB	Given for the DAC in 12-bit configuration.
Offset(4)	Offset error (difference between measured value at Code (0x800) and the ideal value = V_{REF+}/2)	-	-	-	±10	mV	Given for the DAC in 12-bit configuration
		-	-	-	±3	LSB	Given for the DAC in 10-bit at V_{REF+} = 3.6 V
		-	-	-	±12	LSB	Given for the DAC in 12-bit at V_{REF+} = 3.6 V
Gain error(4)	Gain error	-	-	-	±0.5	%	Given for the DAC in 12-bit configuration
$t_{SETTLING}$(4)	Settling time (full scale: for a 10-bit input code transition between the lowest and the highest input codes when DAC_OUT reaches final value ±4LSB	-	-	3	6	µs	$C_{LOAD} \leq 50$ pF, $R_{LOAD} \geq 5$ kΩ
THD(4)	Total Harmonic Distortion Buffer ON	-	-	-	-	dB	$C_{LOAD} \leq 50$ pF, $R_{LOAD} \geq 5$ kΩ
Update rate(2)	Max frequency for a correct DAC_OUT change when small variation in the input code (from code i to i+1LSB)	-	-	-	1	MS/s	$C_{LOAD} \leq 50$ pF, $R_{LOAD} \geq 5$ kΩ

Table 85. DAC characteristics (continued)

Symbol	Parameter	Conditions	Min	Typ	Max	Unit	Comments
t_{WAKEUP}(4)	Wakeup time from off state (Setting the ENx bit in the DAC Control register)	-	-	6.5	10	µs	$C_{LOAD} \leq 50$ pF, $R_{LOAD} \geq 5$ kΩ input code between lowest and highest possible ones.
PSRR+ (2)	Power supply rejection ratio (to V_{DDA}) (static DC measurement)	-	-	–67	–40	dB	No R_{LOAD}, C_{LOAD} = 50 pF

1. V_{DDA} minimum value of 1.7 V is obtained with the use of an external power supply supervisor (refer to *Section 3.17.2: Internal reset OFF*).
2. Guaranteed by design.
3. The quiescent mode corresponds to a state where the DAC maintains a stable output level to ensure that no dynamic consumption occurs.
4. Guaranteed by characterization.

Figure 54. 12-bit buffered /non-buffered DAC



1. The DAC integrates an output buffer that can be used to reduce the output impedance and to drive external loads directly without the use of an external operational amplifier. The buffer can be bypassed by configuring the BOFFx bit in the DAC_CR register.

6.3.26 FMC characteristics

Unless otherwise specified, the parameters given in *Table 86* to *Table 101* for the FMC interface are derived from tests performed under the ambient temperature, f_{HCLK} frequency and V_{DD} supply voltage conditions summarized in *Table 17*, with the following configuration:

- Output speed is set to OSPEEDRy[1:0] = 10 except at V_{DD} range 1.7 to 2.1V where OSPEEDRy[1:0] = 11
- Measurement points are done at CMOS levels: $0.5V_{DD}$

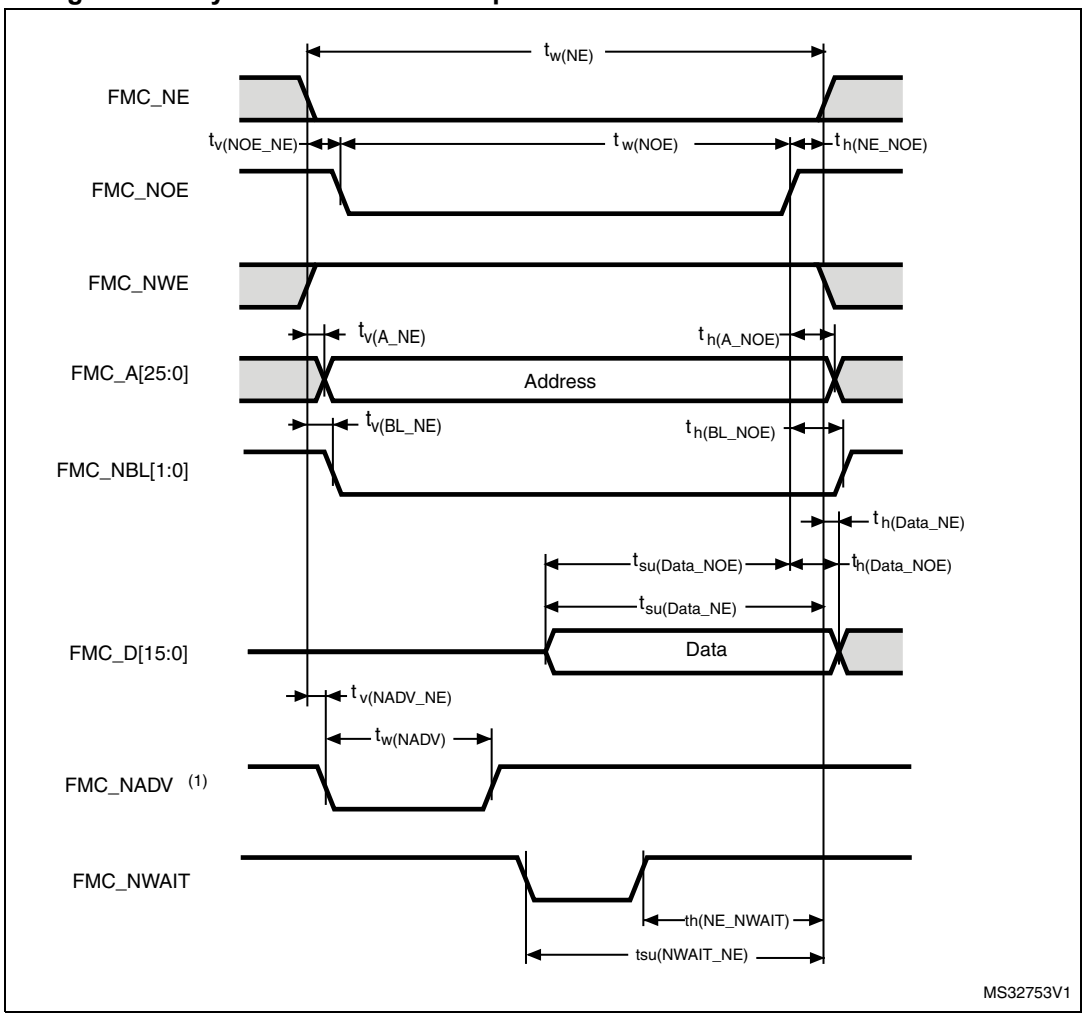
Refer to *Section 6.3.17: I/O port characteristics* for more details on the input/output characteristics.

Asynchronous waveforms and timings

Figure 55 through *Figure 58* represent asynchronous waveforms and *Table 86* through *Table 93* provide the corresponding timings. The results shown in these tables are obtained with the following FMC configuration:

- AddressSetupTime = 0x1
- AddressHoldTime = 0x1
- DataSetupTime = 0x1 (except for asynchronous NWAIT mode , DataSetupTime = 0x5)
- BusTurnAroundDuration = 0x0
- For SDRAM memories, V_{DD} ranges from 2.7 to 3.6 V and maximum frequency FMC_SDCLK = 90 MHz
- For Mobile LPSDR SDRAM memories, V_{DD} ranges from 1.7 to 1.95 V and maximum frequency FMC_SDCLK = 84 MHz

Figure 55. Asynchronous non-multiplexed SRAM/PSRAM/NOR read waveforms



1. Mode 2/B, C and D only. In Mode 1, FMC_NADV is not used.

Table 86. Asynchronous non-multiplexed SRAM/PSRAM/NOR - read timings(1)(2)

Symbol	Parameter	Min	Max	Unit
$t_{w(NE)}$	FMC_NE low time	$2T_{HCLK} - 0.5$	$2\ T_{HCLK}+0.5$	ns
$t_{v(NOE_NE)}$	FMC_NEx low to FMC_NOE low	0	1	ns
$t_{w(NOE)}$	FMC_NOE low time	$2T_{HCLK}$	$2T_{HCLK}+ 0.5$	ns
$t_{h(NE_NOE)}$	FMC_NOE high to FMC_NE high hold time	0	-	ns
$t_{v(A_NE)}$	FMC_NEx low to FMC_A valid	-	2	ns
$t_{h(A_NOE)}$	Address hold time after FMC_NOE high	0	-	ns
$t_{v(BL_NE)}$	FMC_NEx low to FMC_BL valid	-	2	ns
$t_{h(BL_NOE)}$	FMC_BL hold time after FMC_NOE high	0	-	ns
$t_{su(Data_NE)}$	Data to FMC_NEx high setup time	$T_{HCLK} + 2.5$	-	ns
$t_{su(Data_NOE)}$	Data to FMC_NOEx high setup time	$T_{HCLK} +2$	-	ns

Table 86. Asynchronous non-multiplexed SRAM/PSRAM/NOR - read timings(1)(2) (continued)

Symbol	Parameter	Min	Max	Unit
$t_{h(Data_NOE)}$	Data hold time after FMC_NOE high	0	-	ns
$t_{h(Data_NE)}$	Data hold time after FMC_NEx high	0	-	ns
$t_{v(NADV_NE)}$	FMC_NEx low to FMC_NADV low	-	0	ns
$t_{w(NADV)}$	FMC_NADV low time	-	T_{HCLK} +1	ns

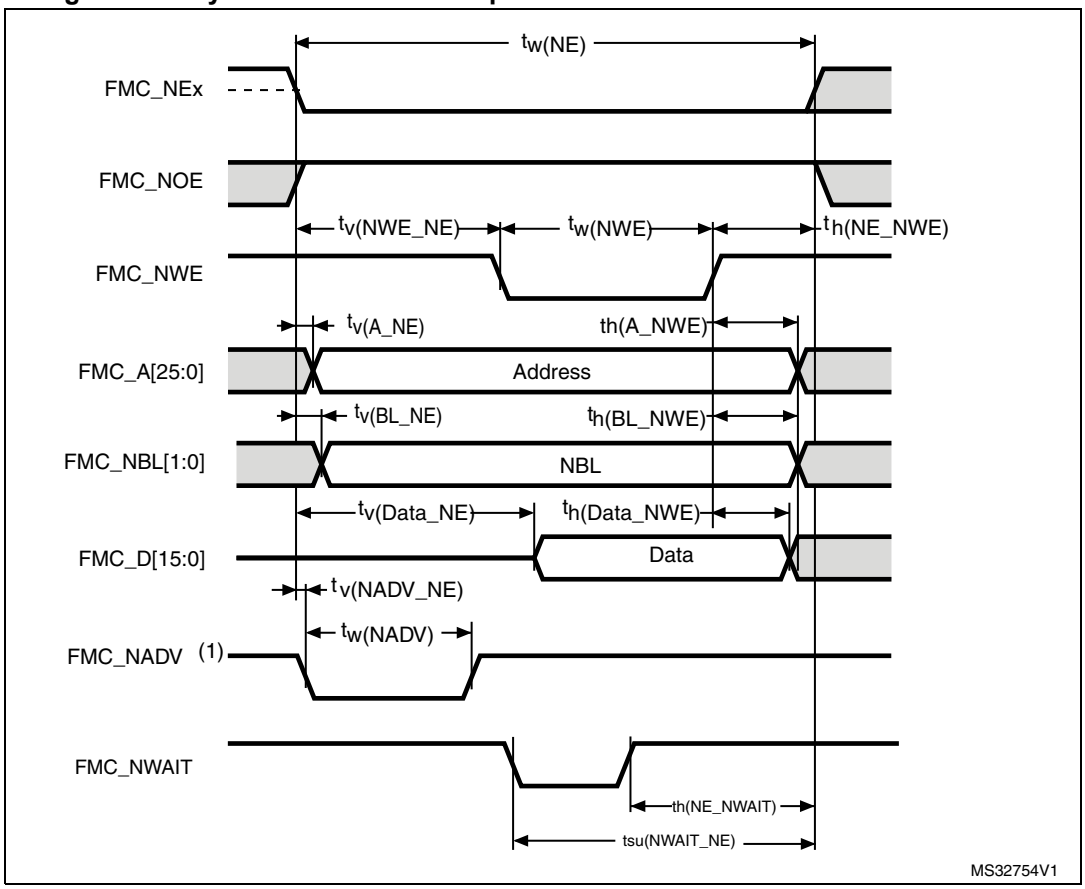
1. C_L = 30 pF.
2. Guaranteed by characterization results.

Table 87. Asynchronous non-multiplexed SRAM/PSRAM/NOR read - NWAIT timings(1)(2)

Symbol	Parameter	Min	Max	Unit
$t_{w(NE)}$	FMC_NE low time	$7T_{HCLK}$+0.5	$7T_{HCLK}$+1	ns
$t_{w(NOE)}$	FMC_NWE low time	$5T_{HCLK}$ − 1.5	$5T_{HCLK}$ +2	
$t_{su(NWAIT_NE)}$	FMC_NWAIT valid before FMC_NEx high	$5T_{HCLK}$+1.5	-	
$t_{h(NE_NWAIT)}$	FMC_NEx hold time after FMC_NWAIT invalid	$4T_{HCLK}$+1	-	

1. C_L = 30 pF.
2. Guaranteed by characterization results.

Figure 56. Asynchronous non-multiplexed SRAM/PSRAM/NOR write waveforms



1. Mode 2/B, C and D only. In Mode 1, FMC_NADV is not used.

Table 88. Asynchronous non-multiplexed SRAM/PSRAM/NOR write timings(1)(2)

Symbol	Parameter	Min	Max	Unit
$t_{w(NE)}$	FMC_NE low time	$3T_{HCLK}$	$3T_{HCLK}+1$	ns
$t_{v(NWE_NE)}$	FMC_NEx low to FMC_NWE low	$T_{HCLK} - 0.5$	$T_{HCLK}+ 0.5$	ns
$t_{w(NWE)}$	FMC_NWE low time	T_{HCLK}	$T_{HCLK}+ 0.5$	ns
$t_{h(NE_NWE)}$	FMC_NWE high to FMC_NE high hold time	T_{HCLK} +1.5	-	ns
$t_{v(A_NE)}$	FMC_NEx low to FMC_A valid	-	0	ns
$t_{h(A_NWE)}$	Address hold time after FMC_NWE high	T_{HCLK}+0.5	-	ns
$t_{v(BL_NE)}$	FMC_NEx low to FMC_BL valid	-	1.5	ns
$t_{h(BL_NWE)}$	FMC_BL hold time after FMC_NWE high	T_{HCLK}+0.5	-	ns
$t_{v(Data_NE)}$	Data to FMC_NEx low to Data valid	-	$T_{HCLK}+ 2$	ns
$t_{h(Data_NWE)}$	Data hold time after FMC_NWE high	T_{HCLK}+0.5	-	ns
$t_{v(NADV_NE)}$	FMC_NEx low to FMC_NADV low	-	0.5	ns
$t_{w(NADV)}$	FMC_NADV low time	-	$T_{HCLK}+ 0.5$	ns

1. C_L = 30 pF.
2. Guaranteed by characterization results.

Table 89. Asynchronous non-multiplexed SRAM/PSRAM/NOR write - NWAIT timings(1)(2)

Symbol	Parameter	Min	Max	Unit
$t_{w(NE)}$	FMC_NE low time	$8T_{HCLK}+1$	$8T_{HCLK}+2$	ns
$t_{w(NWE)}$	FMC_NWE low time	$6T_{HCLK} - 1$	$6T_{HCLK}+2$	ns
$t_{su(NWAIT_NE)}$	FMC_NWAIT valid before FMC_NEx high	$6T_{HCLK}+1.5$	-	ns
$t_{h(NE_NWAIT)}$	FMC_NEx hold time after FMC_NWAIT invalid	$4T_{HCLK}+1$		ns

1. C_L = 30 pF.
2. Guaranteed by characterization results.

Figure 57. Asynchronous multiplexed PSRAM/NOR read waveforms

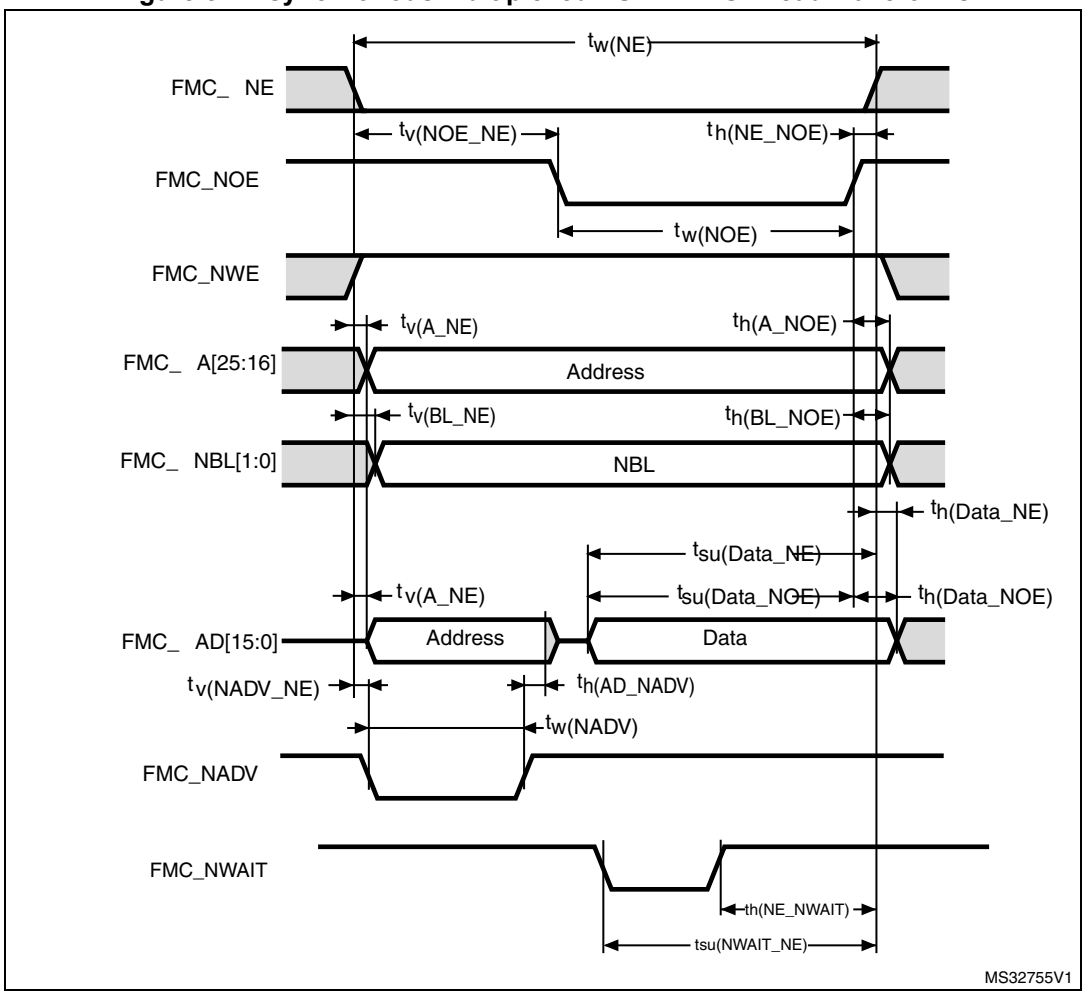


Table 90. Asynchronous multiplexed PSRAM/NOR read timings(1)(2)

Symbol	Parameter	Min	Max	Unit
$t_{w(NE)}$	FMC_NE low time	$3T_{HCLK} - 1$	$3T_{HCLK}+0.5$	ns
$t_{v(NOE_NE)}$	FMC_NEx low to FMC_NOE low	$2T_{HCLK} - 0.5$	$2T_{HCLK}$	ns
$t_{tw(NOE)}$	FMC_NOE low time	$T_{HCLK} - 1$	$T_{HCLK}+1$	ns
$t_{h(NE_NOE)}$	FMC_NOE high to FMC_NE high hold time	1	-	ns
$t_{v(A_NE)}$	FMC_NEx low to FMC_A valid	-	2	ns
$t_{v(NADV_NE)}$	FMC_NEx low to FMC_NADV low	0	2	ns
$t_{w(NADV)}$	FMC_NADV low time	$T_{HCLK} - 0.5$	$T_{HCLK}+0.5$	ns
$t_{h(AD_NADV)}$	FMC_AD(address) valid hold time after FMC_NADV high)	0	-	ns
$t_{h(A_NOE)}$	Address hold time after FMC_NOE high	$T_{HCLK} - 0.5$	-	ns
$t_{h(BL_NOE)}$	FMC_BL time after FMC_NOE high	0	-	ns
$t_{v(BL_NE)}$	FMC_NEx low to FMC_BL valid	-	2	ns
$t_{su(Data_NE)}$	Data to FMC_NEx high setup time	$T_{HCLK}+1.5$	-	ns
$t_{su(Data_NOE)}$	Data to FMC_NOE high setup time	$T_{HCLK}+1$	-	ns
$t_{h(Data_NE)}$	Data hold time after FMC_NEx high	0	-	ns
$t_{h(Data_NOE)}$	Data hold time after FMC_NOE high	0	-	ns

1. C_L = 30 pF.
2. Guaranteed by characterization results.

Table 91. Asynchronous multiplexed PSRAM/NOR read-NWAIT timings(1)(2)

Symbol	Parameter	Min	Max	Unit
$t_{w(NE)}$	FMC_NE low time	$8T_{HCLK}+0.5$	$8T_{HCLK}+2$	ns
$t_{w(NOE)}$	FMC_NWE low time	$5T_{HCLK} - 1$	$5T_{HCLK}+1.5$	ns
$t_{su(NWAIT_NE)}$	FMC_NWAIT valid before FMC_NEx high	$5T_{HCLK}+1.5$	-	ns
$t_{h(NE_NWAIT)}$	FMC_NEx hold time after FMC_NWAIT invalid	$4T_{HCLK}+1$		ns

1. C_L = 30 pF.
2. Guaranteed by characterization results.

Figure 58. Asynchronous multiplexed PSRAM/NOR write waveforms

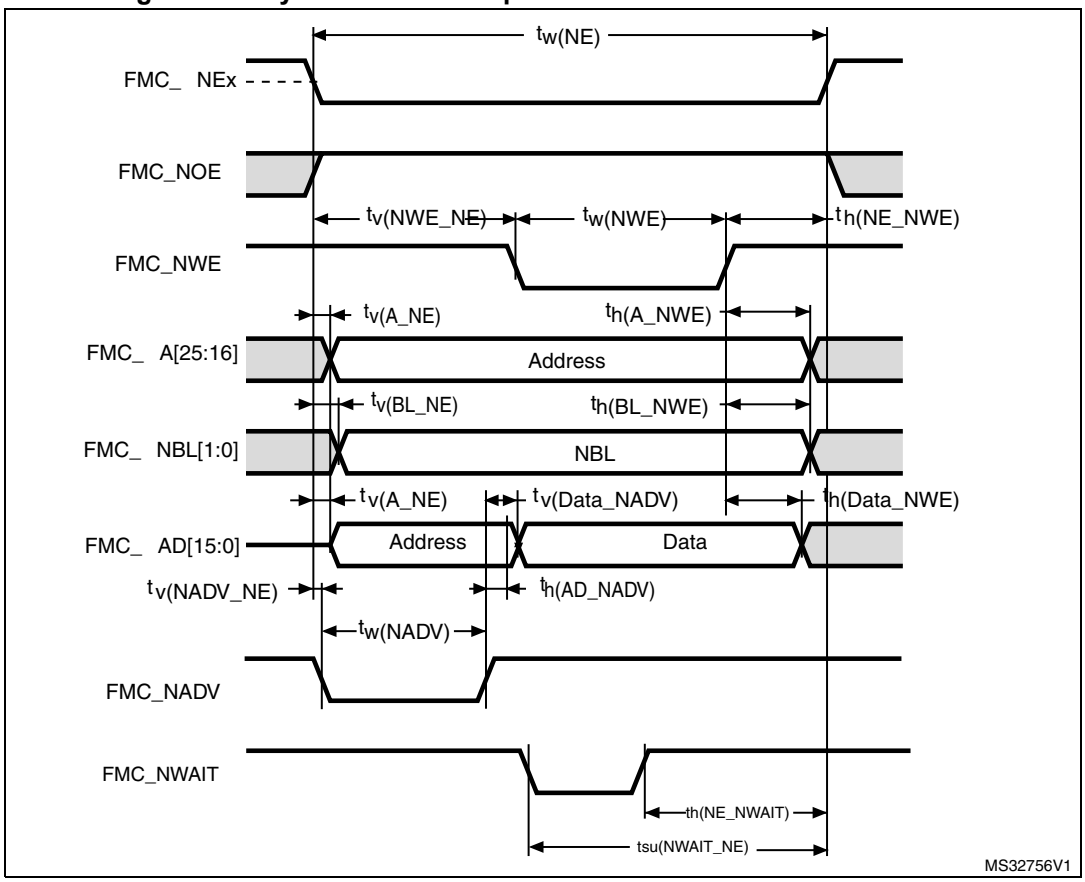


Table 92. Asynchronous multiplexed PSRAM/NOR write timings(1)(2)

Symbol	Parameter	Min	Max	Unit
$t_{w(NE)}$	FMC_NE low time	$4T_{HCLK}$	$4T_{HCLK}+0.5$	ns
$t_{v(NWE_NE)}$	FMC_NEx low to FMC_NWE low	$T_{HCLK} - 1$	$T_{HCLK}+0.5$	ns
$t_{w(NWE)}$	FMC_NWE low time	$2T_{HCLK}$	$2T_{HCLK}+0.5$	ns
$t_{h(NE_NWE)}$	FMC_NWE high to FMC_NE high hold time	T_{HCLK}	-	ns
$t_{v(A_NE)}$	FMC_NEx low to FMC_A valid	-	0	ns
$t_{v(NADV_NE)}$	FMC_NEx low to FMC_NADV low	0.5	1	ns
$t_{w(NADV)}$	FMC_NADV low time	$T_{HCLK} - 0.5$	$T_{HCLK}+ 0.5$	ns
$t_{h(AD_NADV)}$	FMC_AD(adress) valid hold time after FMC_NADV high)	$T_{HCLK} - 2$	-	ns
$t_{h(A_NWE)}$	Address hold time after FMC_NWE high	T_{HCLK}	-	ns
$t_{h(BL_NWE)}$	FMC_BL hold time after FMC_NWE high	$T_{HCLK} - 2$	-	ns
$t_{v(BL_NE)}$	FMC_NEx low to FMC_BL valid	-	2	ns
$t_{v(Data_NADV)}$	FMC_NADV high to Data valid	-	$T_{HCLK}+1.5$	ns
$t_{h(Data_NWE)}$	Data hold time after FMC_NWE high	$T_{HCLK}+0.5$	-	ns

1. C_L = 30 pF.
2. Guaranteed by characterization results.

Table 93. Asynchronous multiplexed PSRAM/NOR write-NWAIT timings(1)(2)

Symbol	Parameter	Min	Max	Unit
$t_{w(NE)}$	FMC_NE low time	$9T_{HCLK}$	$9T_{HCLK}+0.5$	ns
$t_{w(NWE)}$	FMC_NWE low time	$7T_{HCLK}$	$7T_{HCLK}+2$	ns
$t_{su(NWAIT_NE)}$	FMC_NWAIT valid before FMC_NEx high	$6T_{HCLK}+1.5$	-	ns
$t_{h(NE_NWAIT)}$	FMC_NEx hold time after FMC_NWAIT invalid	$4T_{HCLK}-1$	-	ns

1. C_L = 30 pF.
2. Guaranteed by characterization results.

Synchronous waveforms and timings

Figure 59 through *Figure 62* represent synchronous waveforms and *Table 94* through *Table 97* provide the corresponding timings. The results shown in these tables are obtained with the following FMC configuration:

- BurstAccessMode = FMC_BurstAccessMode_Enable;
- MemoryType = FMC_MemoryType_CRAM;
- WriteBurst = FMC_WriteBurst_Enable;
- CLKDivision = 1; (0 is not supported, see the STM32F4xx reference manual : RM0090)
- DataLatency = 1 for NOR Flash; DataLatency = 0 for PSRAM

In all timing tables, the T_{HCLK} is the HCLK clock period (with maximum FMC_CLK = 90 MHz).

Figure 59. Synchronous multiplexed NOR/PSRAM read timings

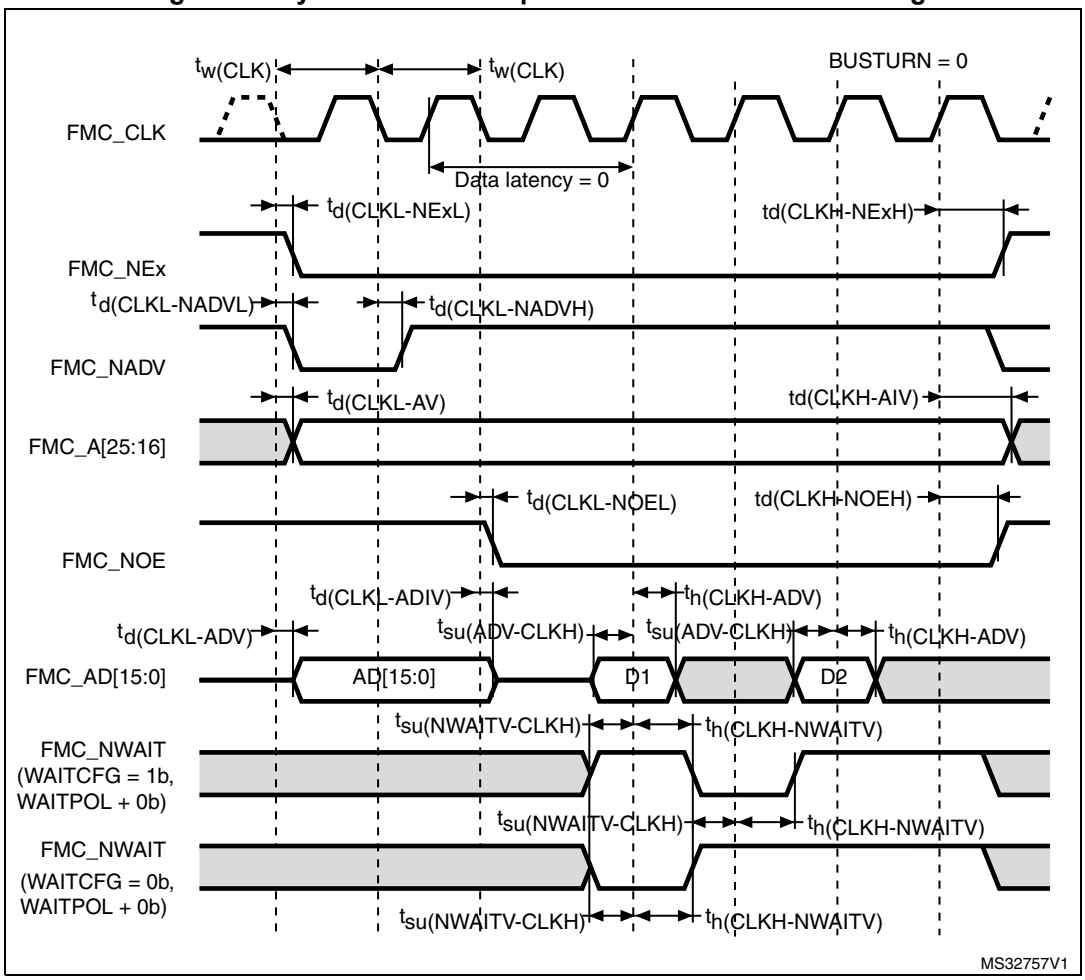


Table 94. Synchronous multiplexed NOR/PSRAM read timings(1)(2)

Symbol	Parameter	Min	Max	Unit
$t_{w(CLK)}$	FMC_CLK period	$2T_{HCLK} - 1$	-	ns
$t_{d(CLKL-NExL)}$	FMC_CLK low to FMC_NEx low (x=0..2)	-	0	ns
$t_{d(CLKH_NExH)}$	FMC_CLK high to FMC_NEx high (x= 0…2)	T_{HCLK}	-	ns
$t_{d(CLKL-NADVL)}$	FMC_CLK low to FMC_NADV low	-	0	ns
$t_{d(CLKL-NADVH)}$	FMC_CLK low to FMC_NADV high	0	-	ns
$t_{d(CLKL-AV)}$	FMC_CLK low to FMC_Ax valid (x=16…25)	-	0	ns
$t_{d(CLKH-AIV)}$	FMC_CLK high to FMC_Ax invalid (x=16…25)	0	-	ns
$t_{d(CLKL-NOEL)}$	FMC_CLK low to FMC_NOE low	-	T_{HCLK}+0.5	ns
$t_{d(CLKH-NOEH)}$	FMC_CLK high to FMC_NOE high	$T_{HCLK} - 0.5$	-	ns
$t_{d(CLKL-ADV)}$	FMC_CLK low to FMC_AD[15:0] valid	-	0.5	ns
$t_{d(CLKL-ADIV)}$	FMC_CLK low to FMC_AD[15:0] invalid	0	-	ns

Table 94. Synchronous multiplexed NOR/PSRAM read timings(1)(2) (continued)

Symbol	Parameter	Min	Max	Unit
$t_{su(ADV-CLKH)}$	FMC_A/D[15:0] valid data before FMC_CLK high	5	-	ns
$t_{h(CLKH-ADV)}$	FMC_A/D[15:0] valid data after FMC_CLK high	0	-	ns
$t_{su(NWAIT-CLKH)}$	FMC_NWAIT valid before FMC_CLK high	4	-	ns
$t_{h(CLKH-NWAIT)}$	FMC_NWAIT valid after FMC_CLK high	0	-	ns

1. C_L = 30 pF.
2. Guaranteed by characterization results.

Figure 60. Synchronous multiplexed PSRAM write timings

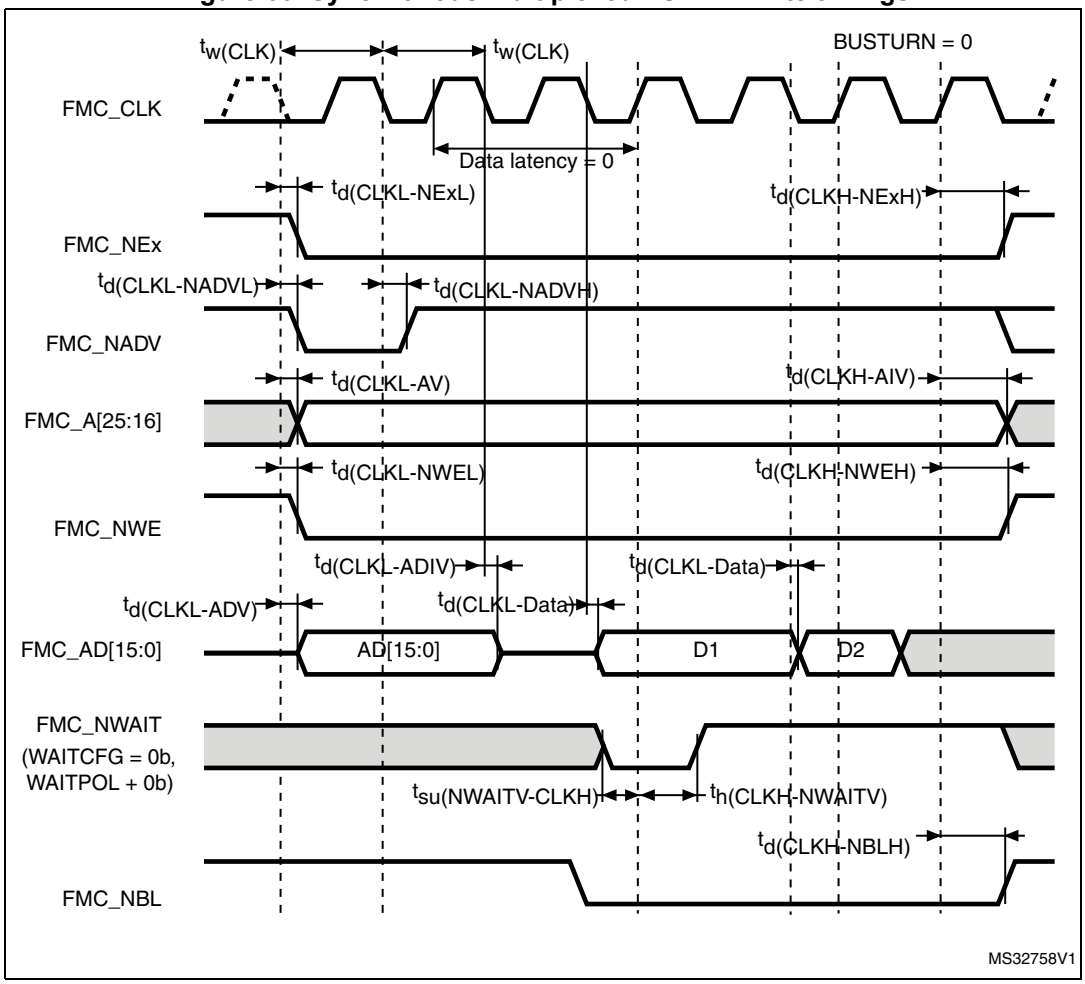


Table 95. Synchronous multiplexed PSRAM write timings(1)(2)

Symbol	Parameter	Min	Max	Unit
$t_{w(CLK)}$	FMC_CLK period, VDD range= 2.7 to 3.6 V	$2T_{HCLK} - 1$	-	ns
$t_{d(CLKL-NExL)}$	FMC_CLK low to FMC_NEx low (x=0..2)	-	1.5	ns
$t_{d(CLKH-NExH)}$	FMC_CLK high to FMC_NEx high (x= 0…2)	T_{HCLK}	-	ns
$t_{d(CLKL-NADVL)}$	FMC_CLK low to FMC_NADV low	-	0	ns
$t_{d(CLKL-NADVH)}$	FMC_CLK low to FMC_NADV high	0	-	ns
$t_{d(CLKL-AV)}$	FMC_CLK low to FMC_Ax valid (x=16…25)	-	0	ns
$t_{d(CLKH-AIV)}$	FMC_CLK high to FMC_Ax invalid (x=16…25)	T_{HCLK}	-	ns
$t_{d(CLKL-NWEL)}$	FMC_CLK low to FMC_NWE low	-	0	ns
$t_{(CLKH-NWEH)}$	FMC_CLK high to FMC_NWE high	$T_{HCLK}-0.5$	-	ns
$t_{d(CLKL-ADV)}$	FMC_CLK low to FMC_AD[15:0] valid	-	3	ns
$t_{d(CLKL-ADIV)}$	FMC_CLK low to FMC_AD[15:0] invalid	0	-	ns
$t_{d(CLKL-DATA)}$	FMC_A/D[15:0] valid data after FMC_CLK low	-	3	ns
$t_{d(CLKL-NBLL)}$	FMC_CLK low to FMC_NBL low	0	-	ns
$t_{d(CLKH-NBLH)}$	FMC_CLK high to FMC_NBL high	$T_{HCLK}-0.5$	-	ns
$t_{su(NWAIT-CLKH)}$	FMC_NWAIT valid before FMC_CLK high	4	-	ns
$t_{h(CLKH-NWAIT)}$	FMC_NWAIT valid after FMC_CLK high	0	-	ns

1. C_L = 30 pF.
2. Guaranteed by characterization results.

Figure 61. Synchronous non-multiplexed NOR/PSRAM read timings

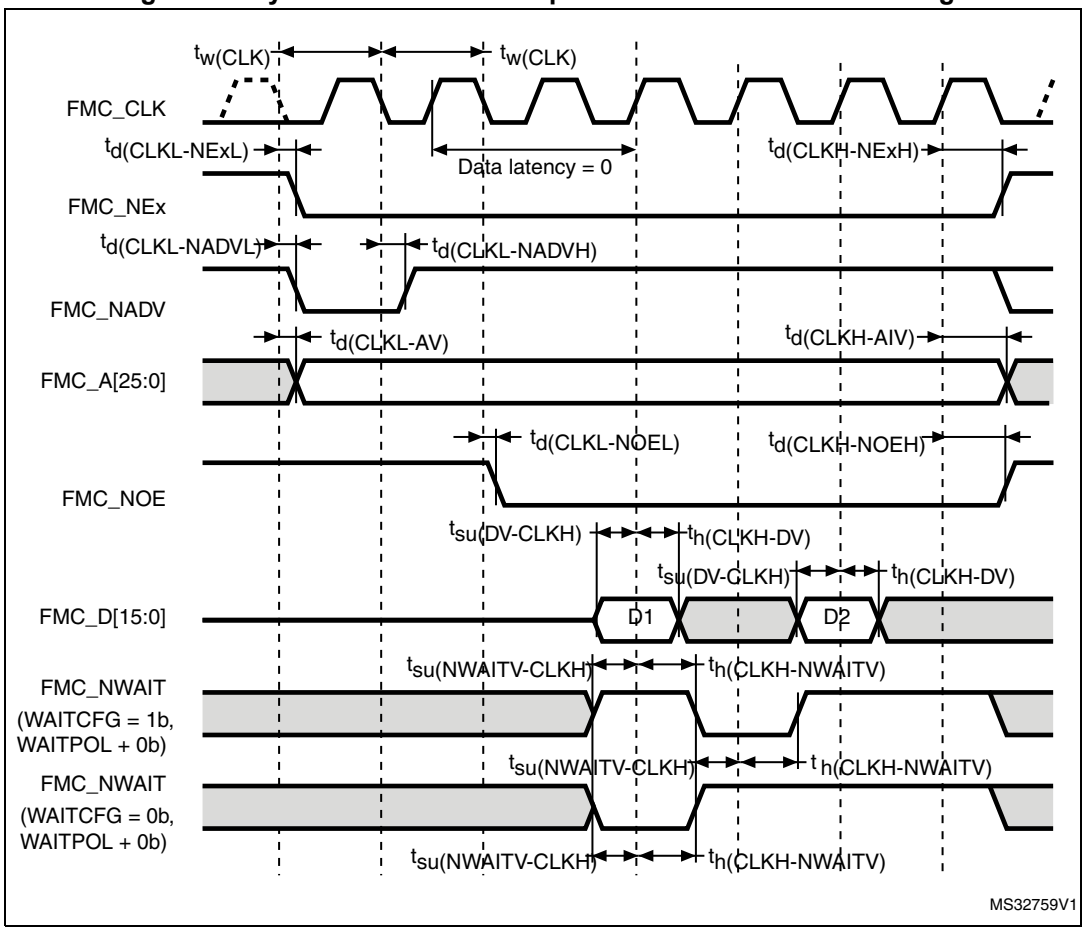


Table 96. Synchronous non-multiplexed NOR/PSRAM read timings(1)(2)

Symbol	Parameter	Min	Max	Unit
$t_{w(CLK)}$	FMC_CLK period	$2T_{HCLK} - 1$	-	ns
$t_{(CLKL-NExL)}$	FMC_CLK low to FMC_NEx low (x=0..2)	-	0.5	ns
$t_{d(CLKH-NExH)}$	FMC_CLK high to FMC_NEx high (x= 0…2)	T_{HCLK}	-	ns
$t_{d(CLKL-NADVL)}$	FMC_CLK low to FMC_NADV low	-	0	ns
$t_{d(CLKL-NADVH)}$	FMC_CLK low to FMC_NADV high	0	-	ns
$t_{d(CLKL-AV)}$	FMC_CLK low to FMC_Ax valid (x=16…25)	-	0	ns
$t_{d(CLKH-AIV)}$	FMC_CLK high to FMC_Ax invalid (x=16…25)	$T_{HCLK} - 0.5$	-	ns
$t_{d(CLKL-NOEL)}$	FMC_CLK low to FMC_NOE low	-	$T_{HCLK}+2$	ns
$t_{d(CLKH-NOEH)}$	FMC_CLK high to FMC_NOE high	$T_{HCLK} - 0.5$	-	ns
$t_{su(DV-CLKH)}$	FMC_D[15:0] valid data before FMC_CLK high	5	-	ns

Table 96. Synchronous non-multiplexed NOR/PSRAM read timings(1)(2) (continued)

Symbol	Parameter	Min	Max	Unit
$t_{h(CLKH-DV)}$	FMC_D[15:0] valid data after FMC_CLK high	0	-	ns
$t_{(NWAIT-CLKH)}$	FMC_NWAIT valid before FMC_CLK high	4		
$t_{h(CLKH-NWAIT)}$	FMC_NWAIT valid after FMC_CLK high	0		

1. C_L = 30 pF.
2. Guaranteed by characterization results.

Figure 62. Synchronous non-multiplexed PSRAM write timings

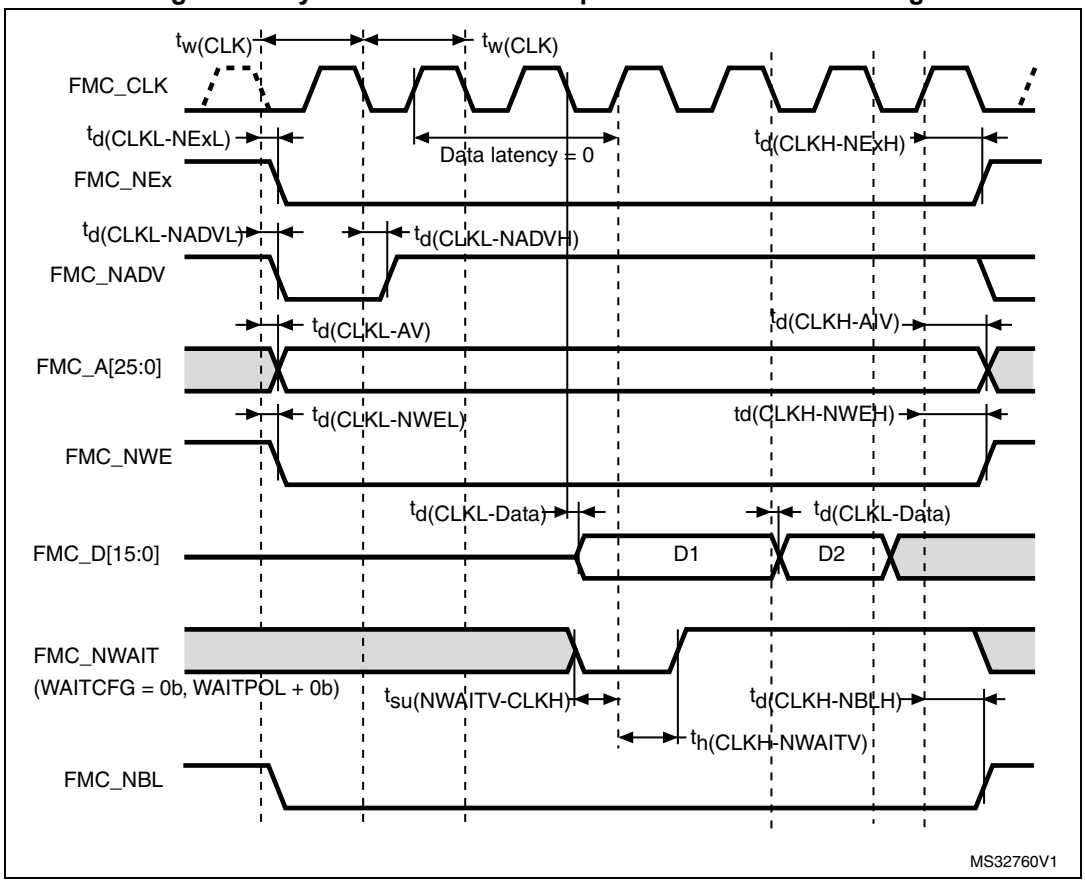


Table 97. Synchronous non-multiplexed PSRAM write timings(1)(2)

Symbol	Parameter	Min	Max	Unit
$t_{(CLK)}$	FMC_CLK period	$2T_{HCLK} - 1$	-	ns
$t_{d(CLKL-NExL)}$	FMC_CLK low to FMC_NEx low (x=0..2)	-	0.5	ns
$t_{(CLKH-NExH)}$	FMC_CLK high to FMC_NEx high (x= 0…2)	T_{HCLK}	-	ns
$t_{d(CLKL-NADVL)}$	FMC_CLK low to FMC_NADV low	-	0	ns
$t_{d(CLKL-NADVH)}$	FMC_CLK low to FMC_NADV high	0	-	ns
$t_{d(CLKL-AV)}$	FMC_CLK low to FMC_Ax valid (x=16…25)	-	0	ns

Table 97. Synchronous non-multiplexed PSRAM write timings(1)(2) (continued)

Symbol	Parameter	Min	Max	Unit
$t_{d(CLKH-AIV)}$	FMC_CLK high to FMC_Ax invalid (x=16…25)	0	-	ns
$t_{d(CLKL-NWEL)}$	FMC_CLK low to FMC_NWE low	-	0	ns
$t_{d(CLKH-NWEH)}$	FMC_CLK high to FMC_NWE high	T_{HCLK}−0.5	-	ns
$t_{d(CLKL-Data)}$	FMC_D[15:0] valid data after FMC_CLK low	-	2.5	ns
$t_{d(CLKL-NBLL)}$	FMC_CLK low to FMC_NBL low	0	-	ns
$t_{d(CLKH-NBLH)}$	FMC_CLK high to FMC_NBL high	T_{HCLK}−0.5	-	ns
$t_{su(NWAIT-CLKH)}$	FMC_NWAIT valid before FMC_CLK high	4		
$t_{h(CLKH-NWAIT)}$	FMC_NWAIT valid after FMC_CLK high	0		

1. C_L = 30 pF.
2. Guaranteed by characterization results.

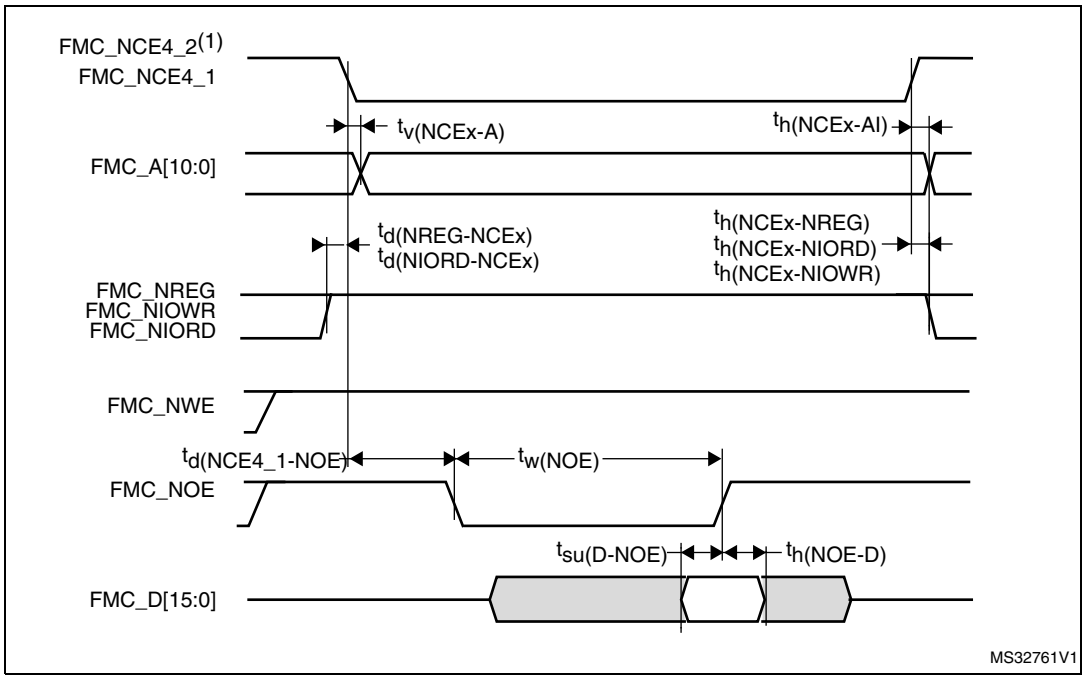
PC Card/CompactFlash controller waveforms and timings

Figure 63 through *Figure 68* represent synchronous waveforms, and *Table 98* and *Table 99* provide the corresponding timings. The results shown in this table are obtained with the following FMC configuration:

- COM.FMC_SetupTime = 0x04;
- COM.FMC_WaitSetupTime = 0x07;
- COM.FMC_HoldSetupTime = 0x04;
- COM.FMC_HiZSetupTime = 0x00;
- ATT.FMC_SetupTime = 0x04;
- ATT.FMC_WaitSetupTime = 0x07;
- ATT.FMC_HoldSetupTime = 0x04;
- ATT.FMC_HiZSetupTime = 0x00;
- IO.FMC_SetupTime = 0x04;
- IO.FMC_WaitSetupTime = 0x07;
- IO.FMC_HoldSetupTime = 0x04;
- IO.FMC_HiZSetupTime = 0x00;
- TCLRSetupTime = 0;
- TARSetupTime = 0.

In all timing tables, the T_{HCLK} is the HCLK clock period.

Figure 63. PC Card/CompactFlash controller waveforms for common memory read access



1. FMC_NCE4_2 remains high (inactive during 8-bit access.

Figure 64. PC Card/CompactFlash controller waveforms for common memory write access

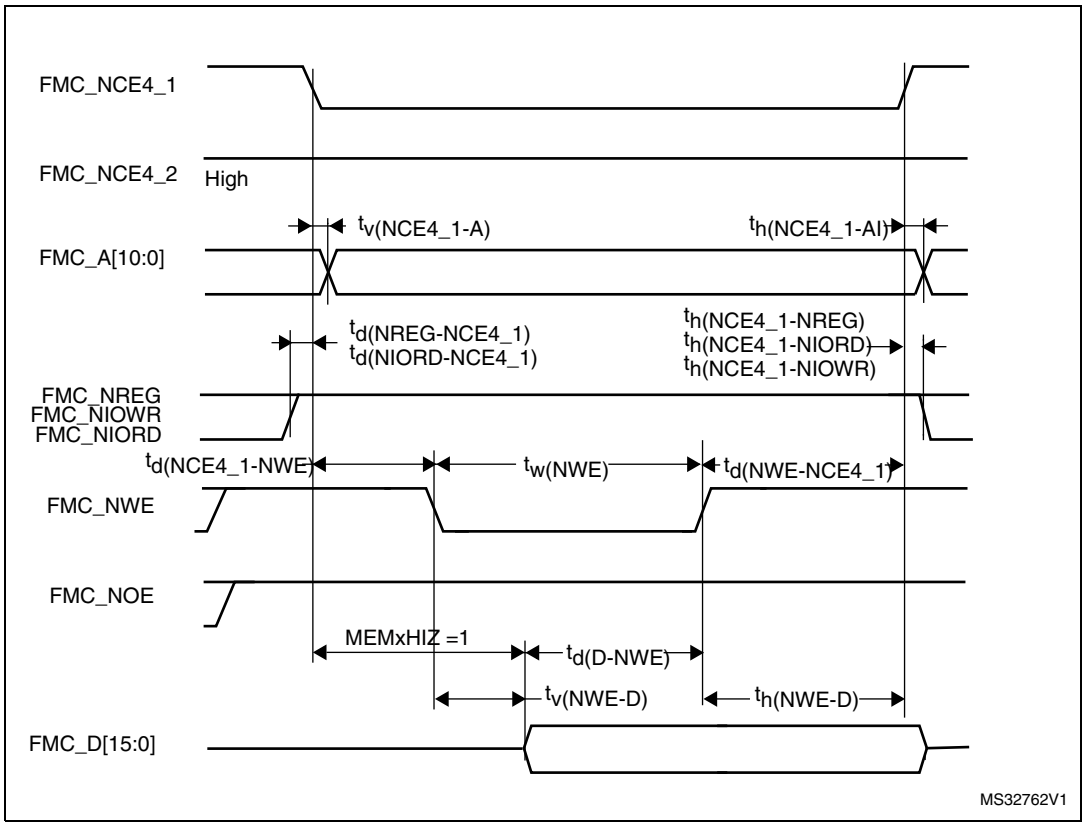
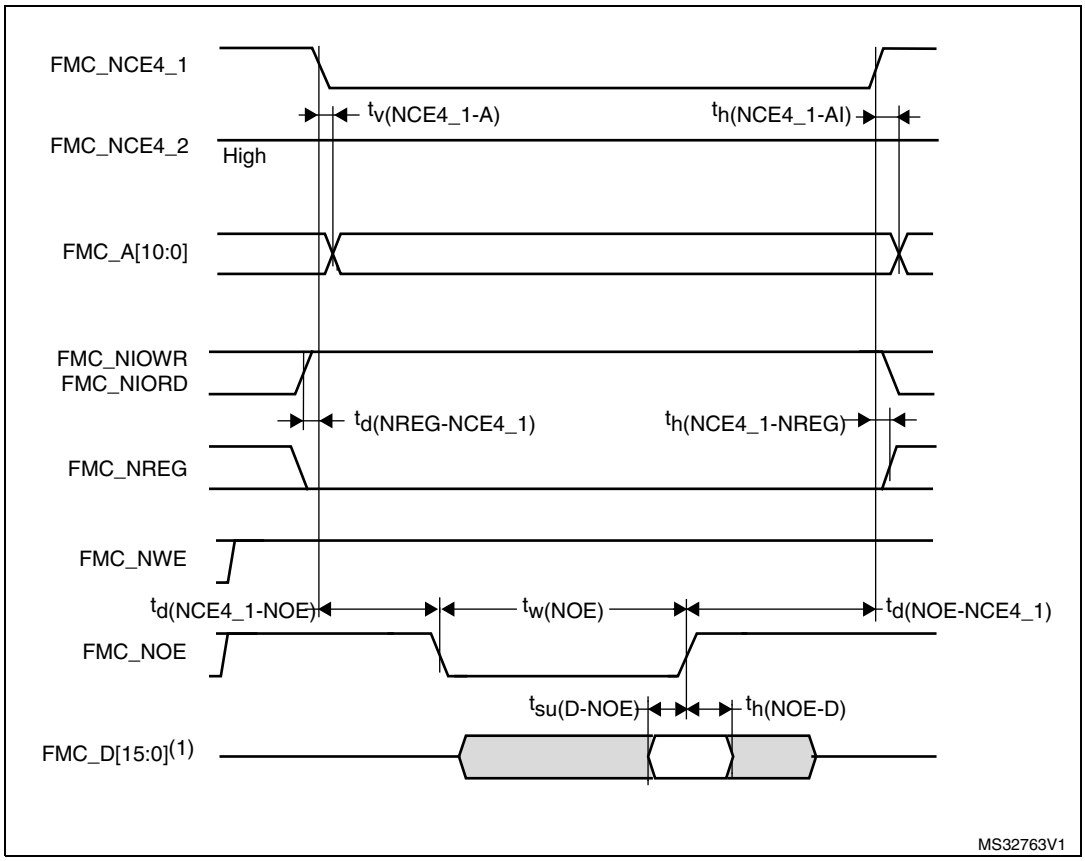
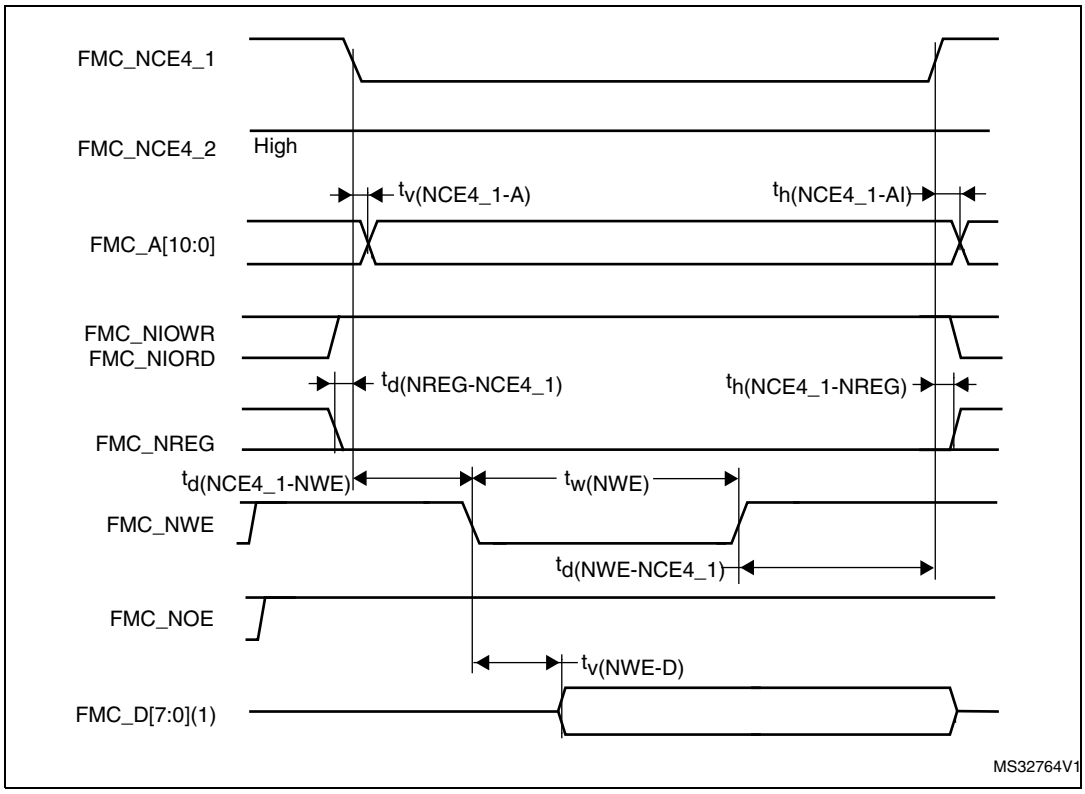


Figure 65. PC Card/CompactFlash controller waveforms for attribute memory read access



1. Only data bits 0...7 are read (bits 8...15 are disregarded).

Figure 66. PC Card/CompactFlash controller waveforms for attribute memory write access



1. Only data bits 0...7 are driven (bits 8...15 remains Hi-Z).

Figure 67. PC Card/CompactFlash controller waveforms for I/O space read access

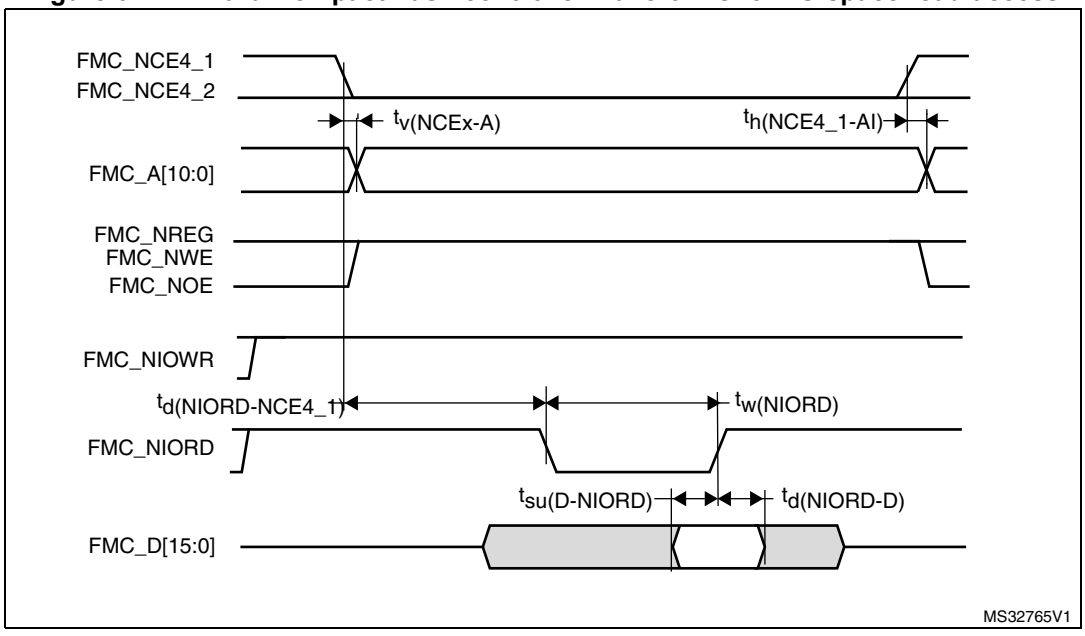


Figure 68. PC Card/CompactFlash controller waveforms for I/O space write access

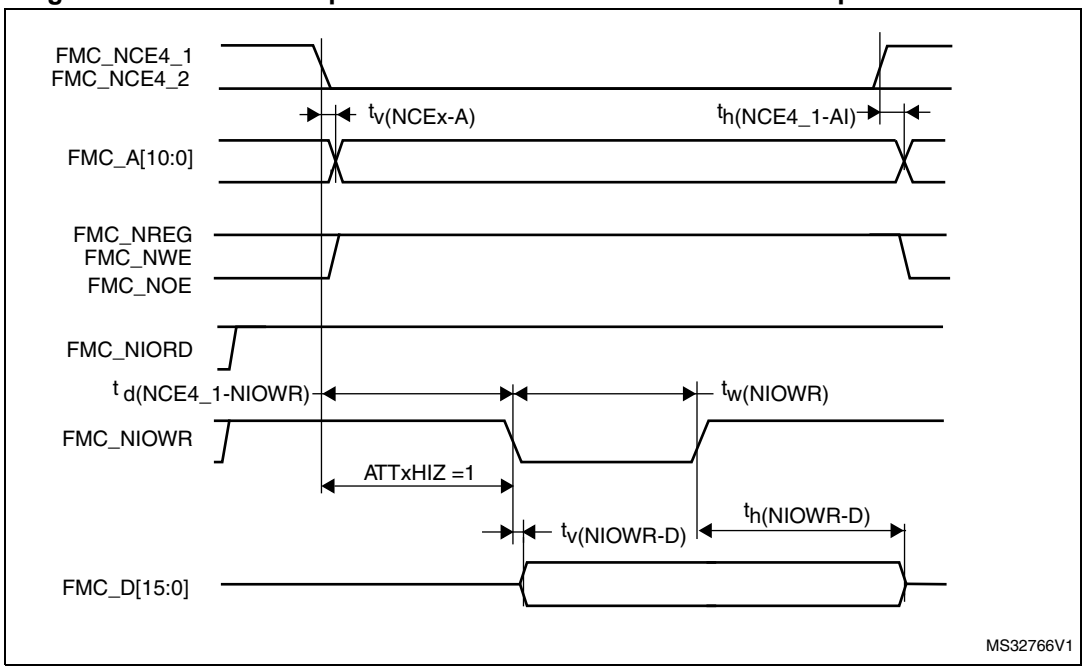


Table 98. Switching characteristics for PC Card/CF read and write cycles in attribute/common space(1)(2)

Symbol	Parameter	Min	Max	Unit
$t_{v(NCEx-A)}$	FMC_Ncex low to FMC_Ay valid	-	0	ns
$t_{h(NCEx_AI)}$	FMC_NCEx high to FMC_Ax invalid	0	-	ns
$t_{d(NREG-NCEx)}$	FMC_NCEx low to FMC_NREG valid	-	1	ns
$t_{h(NCEx-NREG)}$	FMC_NCEx high to FMC_NREG invalid	$T_{HCLK} - 2$	-	ns
$t_{d(NCEx-NWE)}$	FMC_NCEx low to FMC_NWE low	-	$5T_{HCLK}$	ns
$t_{w(NWE)}$	FMC_NWE low width	$8T_{HCLK} - 0.5$	$8T_{HCLK}+0.5$	ns
$t_{d(NWE_NCEx)}$	FMC_NWE high to FMC_NCEx high	$5T_{HCLK}+1$	-	ns
$t_{V(NWE-D)}$	FMC_NWE low to FMC_D[15:0] valid	-	0	ns
$t_{h(NWE-D)}$	FMC_NWE high to FMC_D[15:0] invalid	$9T_{HCLK} - 0.5$	-	ns
$t_{d(D-NWE)}$	FMC_D[15:0] valid before FMC_NWE high	$13T_{HCLK} - 3$		ns
$t_{d(NCEx-NOE)}$	FMC_NCEx low to FMC_NOE low	-	$5T_{HCLK}$	ns
$t_{w(NOE)}$	FMC_NOE low width	$8\ T_{HCLK} - 0.5$	$8\ T_{HCLK}+0.5$	ns
$t_{d(NOE_NCEx)}$	FMC_NOE high to FMC_NCEx high	$5T_{HCLK} - 1$	-	ns
$t_{su\ (D-NOE)}$	FMC_D[15:0] valid data before FMC_NOE high	T_{HCLK}	-	ns
$t_{h(NOE-D)}$	FMC_NOE high to FMC_D[15:0] invalid	0	-	ns

1. C_L = 30 pF.
2. Guaranteed by characterization results.

Table 99. Switching characteristics for PC Card/CF read and write cycles in I/O space(1)(2)

Symbol	Parameter	Min	Max	Unit
tw(NIOWR)	FMC_NIOWR low width	$8T_{HCLK} - 0.5$	-	ns
tv(NIOWR-D)	FMC_NIOWR low to FMC_D[15:0] valid	-	0	ns
th(NIOWR-D)	FMC_NIOWR high to FMC_D[15:0] invalid	$9T_{HCLK} - 2$	-	ns
td(NCE4_1-NIOWR)	FMC_NCE4_1 low to FMC_NIOWR valid	-	$5T_{HCLK}$	ns
th(NCEx-NIOWR)	FMC_NCEx high to FMC_NIOWR invalid	$5T_{HCLK}$	-	ns
td(NIORD-NCEx)	FMC_NCEx low to FMC_NIORD valid	-	$5T_{HCLK}$	ns
th(NCEx-NIORD)	FMC_NCEx high to FMC_NIORD) valid	$6T_{HCLK}+2$	-	ns
tw(NIORD)	FMC_NIORD low width	$8T_{HCLK} - 0.5$	$8T_{HCLK}+0.5$	ns
tsu(D-NIORD)	FMC_D[15:0] valid before FMC_NIORD high	T_{HCLK}	-	ns
td(NIORD-D)	FMC_D[15:0] valid after FMC_NIORD high	0	-	ns

1. C_L = 30 pF.
2. Guaranteed by characterization results.

NAND controller waveforms and timings

Figure 69 through *Figure 72* represent synchronous waveforms, and *Table 100* and *Table 101* provide the corresponding timings. The results shown in this table are obtained with the following FMC configuration:

- COM.FMC_SetupTime = 0x01;
- COM.FMC_WaitSetupTime = 0x03;
- COM.FMC_HoldSetupTime = 0x02;
- COM.FMC_HiZSetupTime = 0x01;
- ATT.FMC_SetupTime = 0x01;
- ATT.FMC_WaitSetupTime = 0x03;
- ATT.FMC_HoldSetupTime = 0x02;
- ATT.FMC_HiZSetupTime = 0x01;
- Bank = FMC_Bank_NAND;
- MemoryDataWidth = FMC_MemoryDataWidth_16b;
- ECC = FMC_ECC_Enable;
- ECCPageSize = FMC_ECCPageSize_512Bytes;
- TCLRSetupTime = 0;
- TARSetupTime = 0.

In all timing tables, the T_{HCLK} is the HCLK clock period.

Figure 69. NAND controller waveforms for read access

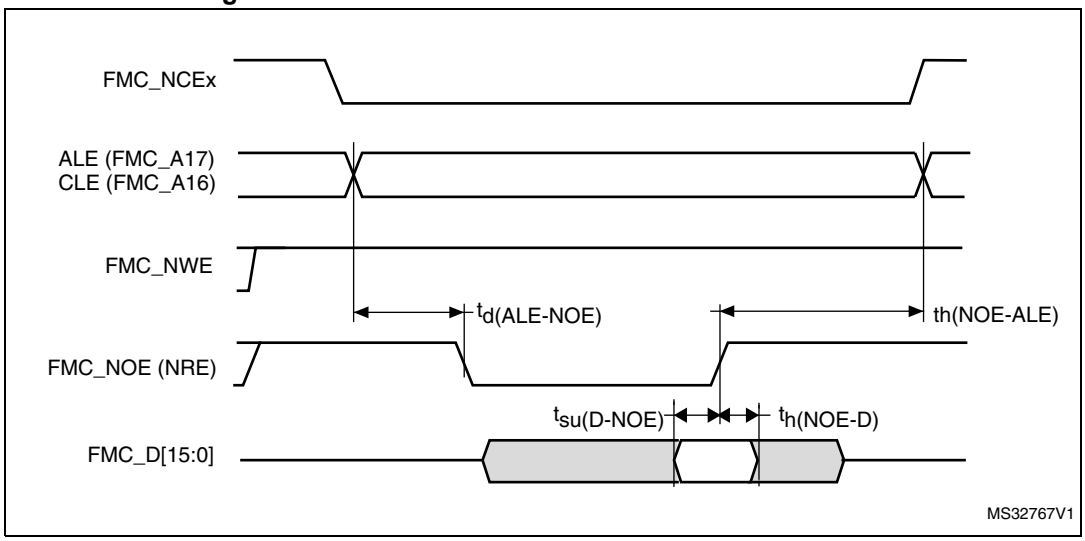


Figure 70. NAND controller waveforms for write access

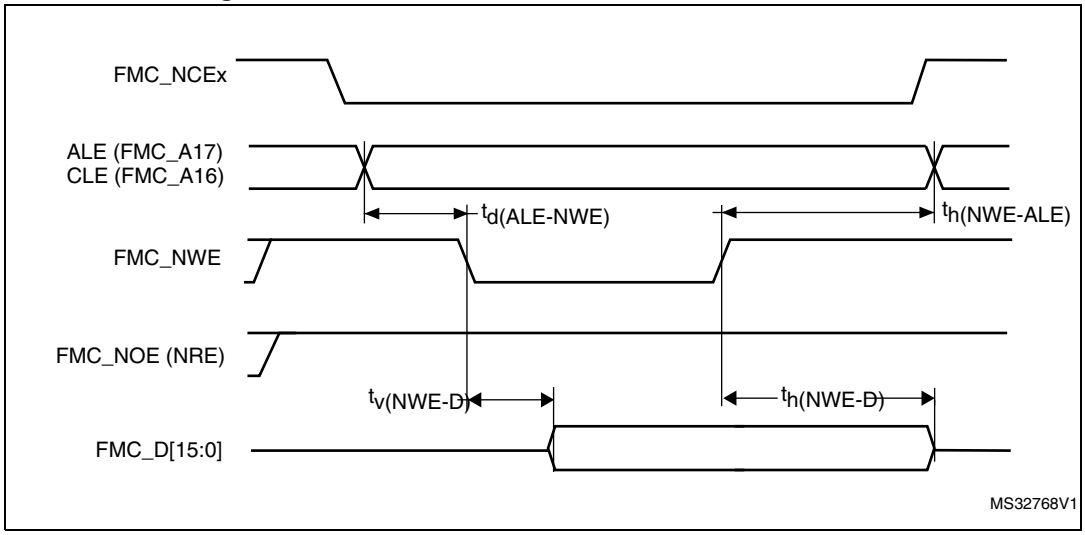


Figure 71. NAND controller waveforms for common memory read access

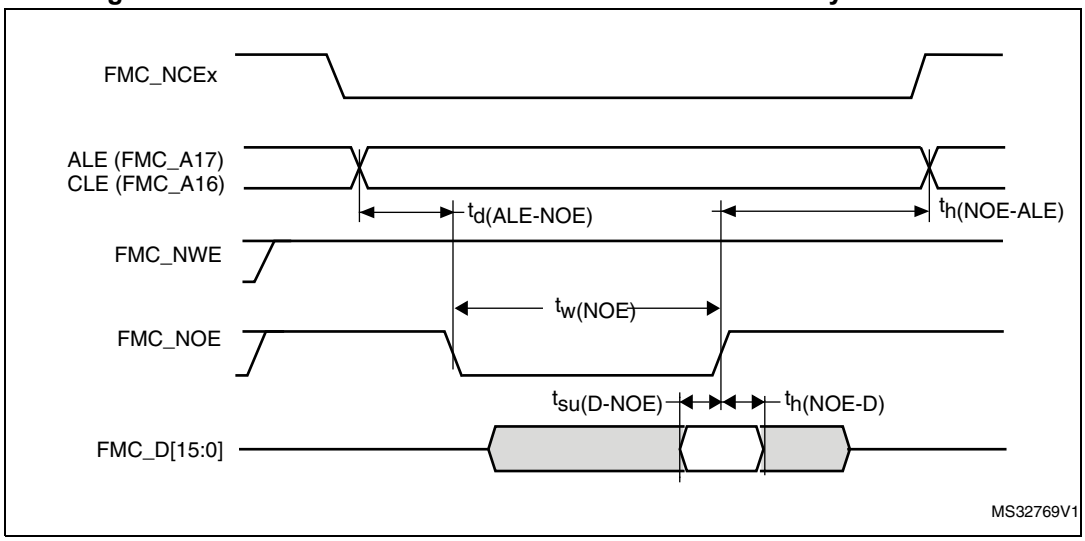


Figure 72. NAND controller waveforms for common memory write access

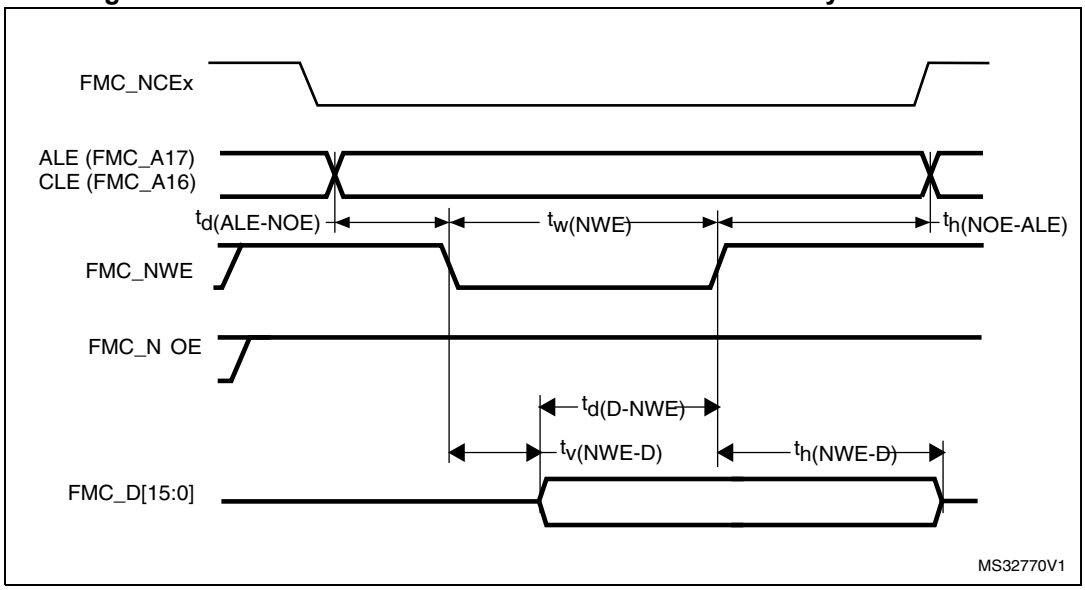


Table 100. Switching characteristics for NAND Flash read cycles(1)

Symbol	Parameter	Min	Max	Unit
$t_{w(N0E)}$	FMC_NOE low width	$4T_{HCLK} - 0.5$	$4T_{HCLK}+0.5$	ns
$t_{su(D-NOE)}$	FMC_D[15-0] valid data before FMC_NOE high	9	-	ns
$t_{h(NOE-D)}$	FMC_D[15-0] valid data after FMC_NOE high	0	-	ns
$t_{d(ALE-NOE)}$	FMC_ALE valid before FMC_NOE low	-	$3T_{HCLK} - 0.5$	ns
$t_{h(NOE-ALE)}$	FMC_NWE high to FMC_ALE invalid	$3T_{HCLK} - 2$	-	ns

1. C_L = 30 pF.

Table 101. Switching characteristics for NAND Flash write cycles(1)

Symbol	Parameter	Min	Max	Unit
$t_{w(NWE)}$	FMC_NWE low width	$4T_{HCLK}$	$4T_{HCLK}+1$	ns
$t_{v(NWE-D)}$	FMC_NWE low to FMC_D[15-0] valid	0	-	ns
$t_{h(NWE-D)}$	FMC_NWE high to FMC_D[15-0] invalid	$3T_{HCLK} - 1$	-	ns
$t_{d(D-NWE)}$	FMC_D[15-0] valid before FMC_NWE high	$5T_{HCLK} - 3$	-	ns
$t_{d(ALE-NWE)}$	FMC_ALE valid before FMC_NWE low	-	$3T_{HCLK}-0.5$	ns
$t_{h(NWE-ALE)}$	FMC_NWE high to FMC_ALE invalid	$3T_{HCLK} - 1$	-	ns

1. C_L = 30 pF.

SDRAM waveforms and timings

Figure 73. SDRAM read access waveforms (CL = 1)

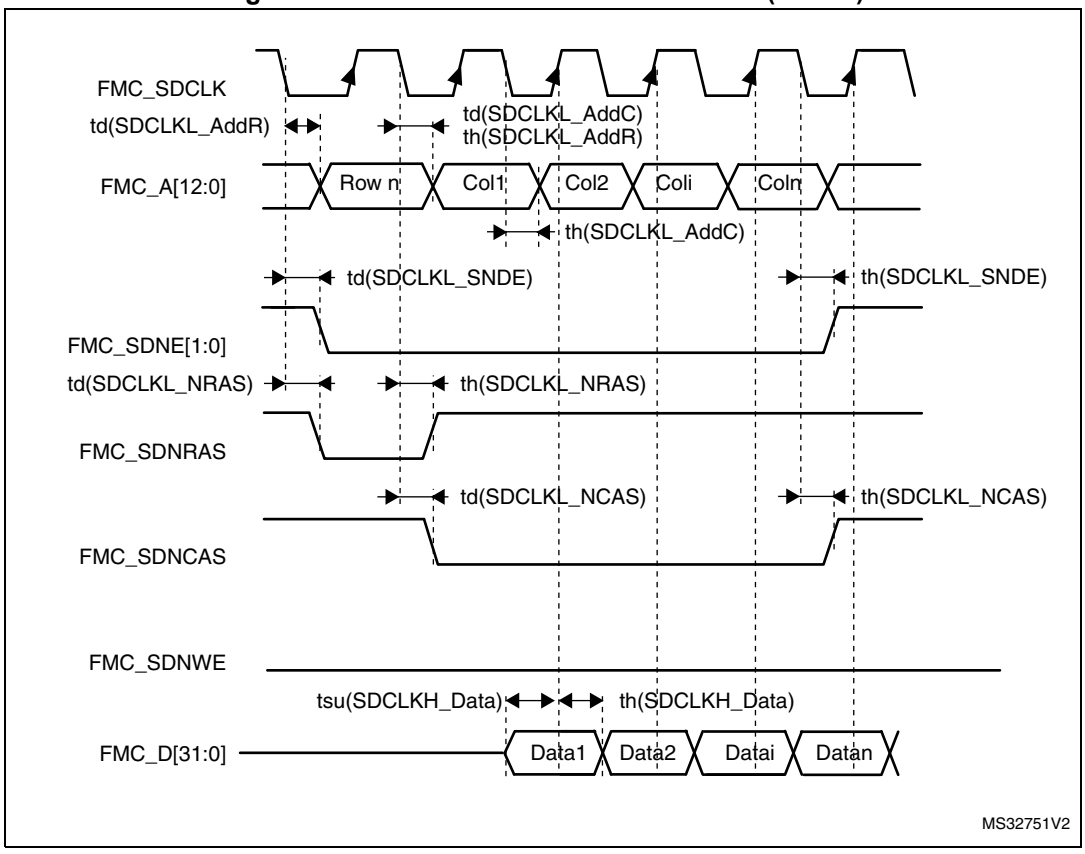


Table 102. SDRAM read timings(1)(2)

Symbol	Parameter	Min	Max	Unit
$t_{w(SDCLK)}$	FMC_SDCLK period	$2T_{HCLK} - 0.5$	$2T_{HCLK}+0.5$	ns
$t_{su(SDCLKH_Data)}$	Data input setup time	2	-	
$t_{h(SDCLKH_Data)}$	Data input hold time	0	-	
$t_{d(SDCLKL_Add)}$	Address valid time	-	1.5	
$t_{d(SDCLKL-SDNE)}$	Chip select valid time	-	0.5	
$t_{h(SDCLKL_SDNE)}$	Chip select hold time	0	-	
$t_{d(SDCLKL_SDNRAS)}$	SDNRAS valid time	-	0.5	
$t_{h(SDCLKL_SDNRAS)}$	SDNRAS hold time	0	-	
$t_{d(SDCLKL_SDNCAS)}$	SDNCAS valid time	-	0.5	
$t_{h(SDCLKL_SDNCAS)}$	SDNCAS hold time	0	-	

1. CL = 30 pF on data and address lines. CL=15pF on FMC_SDCLK.
2. Guaranteed by characterization results.

Table 103. LPSDR SDRAM read timings(1)(2)

Symbol	Parameter	Min	Max	Unit
$t_{W(SDCLK)}$	FMC_SDCLK period	$2T_{HCLK} - 0.5$	$2T_{HCLK}+0.5$	ns
$t_{su(SDCLKH_Data)}$	Data input setup time	2.5	-	
$t_{h(SDCLKH_Data)}$	Data input hold time	0	-	
$t_{d(SDCLKL_Add)}$	Address valid time	-	1	
$t_{d(SDCLKL_SDNE)}$	Chip select valid time	-	1	
$t_{h(SDCLKL_SDNE)}$	Chip select hold time	1	-	
$t_{d(SDCLKL_SDNRAS}$	SDNRAS valid time	-	1	
$t_{h(SDCLKL_SDNRAS)}$	SDNRAS hold time	1	-	
$t_{d(SDCLKL_SDNCAS)}$	SDNCAS valid time	-	1	
$t_{h(SDCLKL_SDNCAS)}$	SDNCAS hold time	1	-	

1. CL = 10 pF.
2. Guaranteed by characterization results.

Figure 74. SDRAM write access waveforms

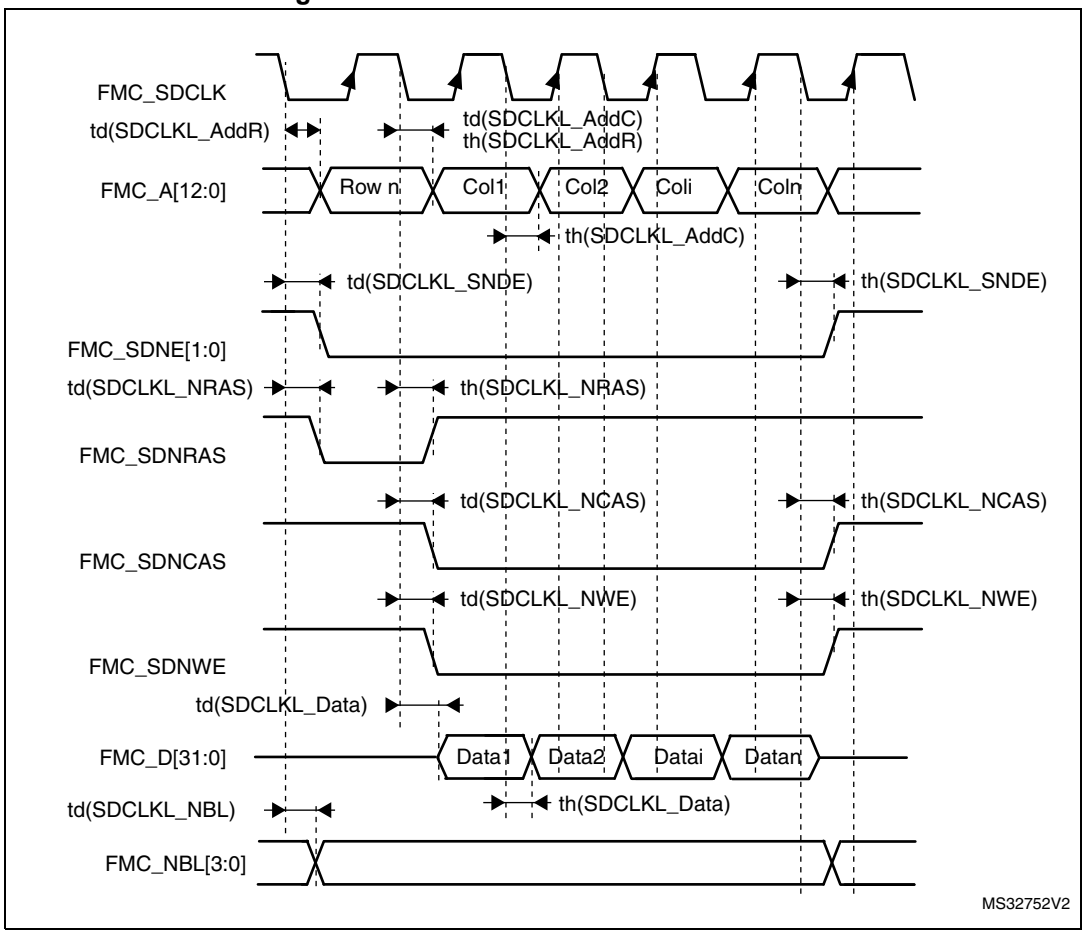


Table 104. SDRAM write timings(1)(2)

Symbol	Parameter	Min	Max	Unit
$t_{w(SDCLK)}$	FMC_SDCLK period	$2T_{HCLK} - 0.5$	$2T_{HCLK}+0.5$	ns
$t_{d(SDCLKL_Data)}$	Data output valid time	-	3.5	
$t_{h(SDCLKL_Data)}$	Data output hold time	0	-	
$t_{d(SDCLKL_Add)}$	Address valid time	-	1.5	
$t_{d(SDCLKL_SDNWE)}$	SDNWE valid time	-	1	
$t_{h(SDCLKL_SDNWE)}$	SDNWE hold time	0	-	
$t_{d(SDCLKL_SDNE)}$	Chip select valid time	-	0.5	
$t_{h(SDCLKL-_SDNE)}$	Chip select hold time	0	-	
$t_{d(SDCLKL_SDNRAS)}$	SDNRAS valid time	-	2	
$t_{h(SDCLKL_SDNRAS)}$	SDNRAS hold time	0	-	
$t_{d(SDCLKL_SDNCAS)}$	SDNCAS valid time	-	0.5	
$t_{d(SDCLKL_SDNCAS)}$	SDNCAS hold time	0	-	
$t_{d(SDCLKL_NBL)}$	NBL valid time	-	0.5	
$t_{h(SDCLKL_NBL)}$	NBLoutput time	0	-	

1. CL = 30 pF on data and address lines. CL=15pF on FMC_SDCLK.
2. Guaranteed by characterization results.

Table 105. LPSDR SDRAM write timings(1)(2)

Symbol	Parameter	Min	Max	Unit
$t_{w(SDCLK)}$	FMC_SDCLK period	$2T_{HCLK} - 0.5$	$2T_{HCLK}+0.5$	ns
$t_{d(SDCLKL_Data)}$	Data output valid time	-	5	
$t_{h(SDCLKL_Data)}$	Data output hold time	2	-	
$t_{d(SDCLKL_Add)}$	Address valid time	-	2.8	
$t_{d(SDCLKL-SDNWE)}$	SDNWE valid time	-	2	
$t_{h(SDCLKL-SDNWE)}$	SDNWE hold time	1	-	
$t_{d(SDCLKL-\ SDNE)}$	Chip select valid time	-	1.5	
$t_{h(SDCLKL-\ SDNE)}$	Chip select hold time	1	-	
$t_{d(SDCLKL-SDNRAS)}$	SDNRAS valid time	-	1.5	
$t_{h(SDCLKL-SDNRAS)}$	SDNRAS hold time	1.5	-	
$t_{d(SDCLKL-SDNCAS)}$	SDNCAS valid time	-	1.5	
$t_{d(SDCLKL-SDNCAS)}$	SDNCAS hold time	1.5	-	
$t_{d(SDCLKL_NBL)}$	NBL valid time	-	1.5	
$t_{h(SDCLKL-NBL)}$	NBL output time	1.5	-	

1. CL = 10 pF.
2. Guaranteed by characterization results.

6.3.27 Camera interface (DCMI) timing specifications

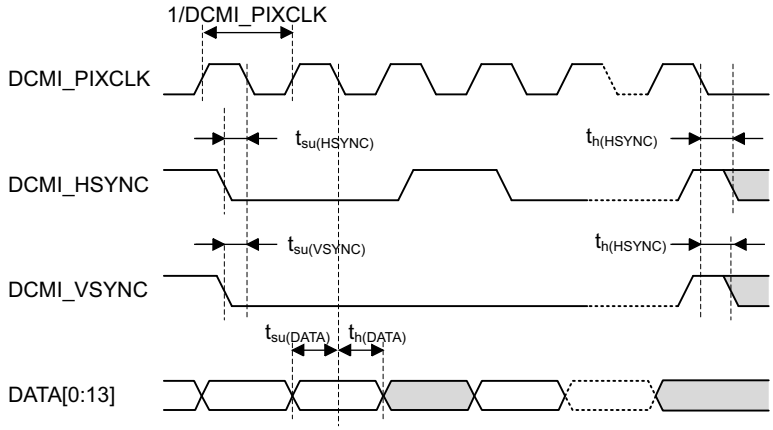
Unless otherwise specified, the parameters given in *Table 106* for DCMI are derived from tests performed under the ambient temperature, f_{HCLK} frequency and V_{DD} supply voltage summarized in *Table 17*, with the following configuration:

- DCMI_PIXCLK polarity: falling
- DCMI_VSYNC and DCMI_HSYNC polarity: high
- Data formats: 14 bits

Table 106. DCMI characteristics

Symbol	Parameter	Min	Max	Unit
	Frequency ratio DCMI_PIXCLK/f_{HCLK}	-	0.4	
DCMI_PIXCLK	Pixel clock input	-	54	MHz
D_{Pixel}	Pixel clock input duty cycle	30	70	%
$t_{su(DATA)}$	Data input setup time	2	-	ns
$t_{h(DATA)}$	Data input hold time	2.5	-	
$t_{su(HSYNC)}$ $t_{su(VSYNC)}$	DCMI_HSYNC/DCMI_VSYNC input setup time	0.5	-	
$t_{h(HSYNC)}$ $t_{h(VSYNC)}$	DCMI_HSYNC/DCMI_VSYNC input hold time	1	-	

Figure 75. DCMI timing diagram



6.3.28 LCD-TFT controller (LTDC) characteristics

Unless otherwise specified, the parameters given in *Table 107* for LCD-TFT are derived from tests performed under the ambient temperature, f_{HCLK} frequency and VDD supply voltage summarized in *Table 17*, with the following configuration:

- LCD_CLK polarity: high
- LCD_DE polarity : low
- LCD_VSYNC and LCD_HSYNC polarity: high
- Pixel formats: 24 bits

Table 107. LTDC characteristics

Symbol	Parameter	Min	Max	Unit
f_{CLK}	LTDC clock output frequency	-	83	MHz
D_{CLK}	LTDC clock output duty cycle	45	55	%
$t_{w(CLKH)}$ $t_{w(CLKL)}$	Clock High time, low time	tw(CLK)/2 − 0.5	tw(CLK)/2+0.5	ns
$t_{v(DATA)}$	Data output valid time	-	3.5	
$t_{h(DATA)}$	Data output hold time	1.5	-	
$t_{v(HSYNC)}$	HSYNC/VSYNC/DE output valid time	-	2.5	
$t_{v(VSYNC)}$				
$t_{v(DE)}$				
$t_{h(HSYNC)}$	HSYNC/VSYNC/DE output hold time	2	-	
$t_{h(VSYNC)}$				
th(DE)				

Figure 76. LCD-TFT horizontal timing diagram

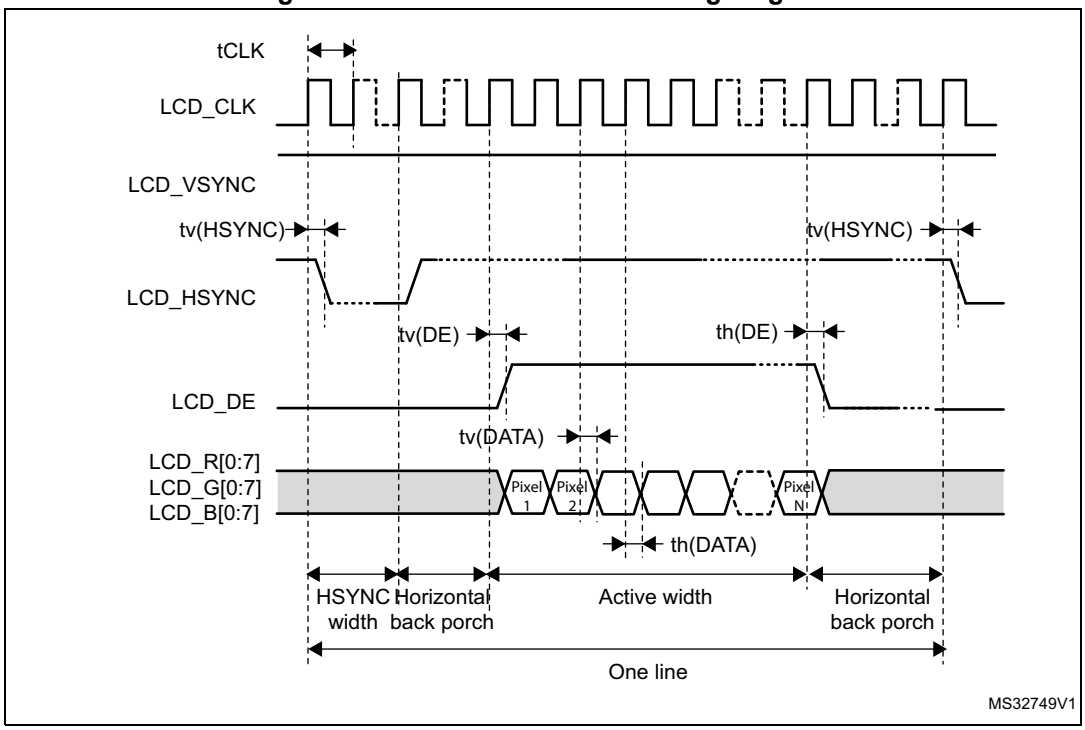
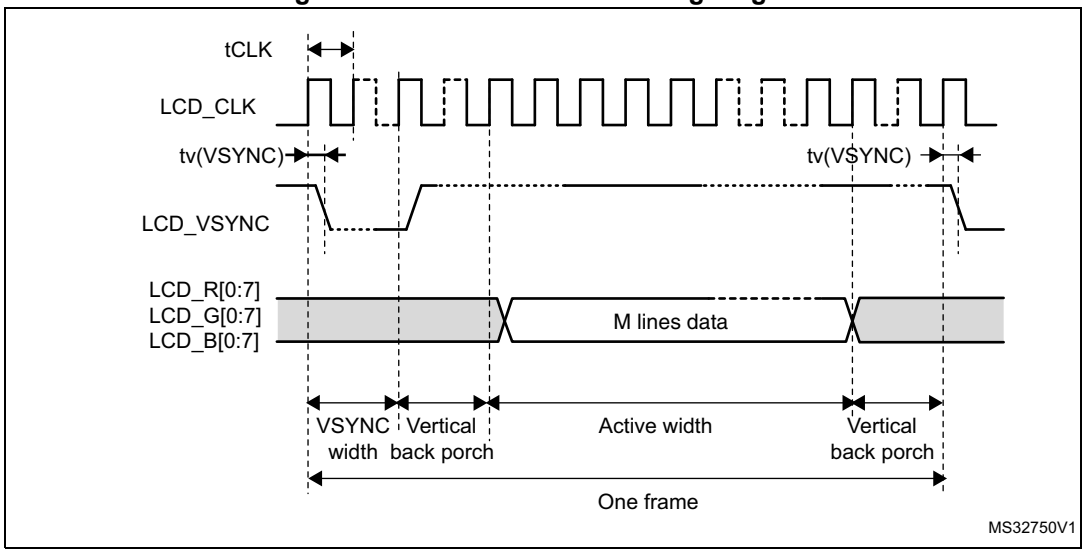


Figure 77. LCD-TFT vertical timing diagram



6.3.29 SD/SDIO MMC card host interface (SDIO) characteristics

Unless otherwise specified, the parameters given in *Table 108* for the SDIO/MMC interface are derived from tests performed under the ambient temperature, f_{PCLK2} frequency and V_{DD} supply voltage conditions summarized in *Table 17*, with the following configuration:

- Output speed is set to OSPEEDRy[1:0] = 10
- Capacitive load C = 30 pF
- Measurement points are done at CMOS levels: $0.5V_{DD}$

Refer to *Section 6.3.17: I/O port characteristics* for more details on the input/output characteristics.

Figure 78. SDIO high-speed mode

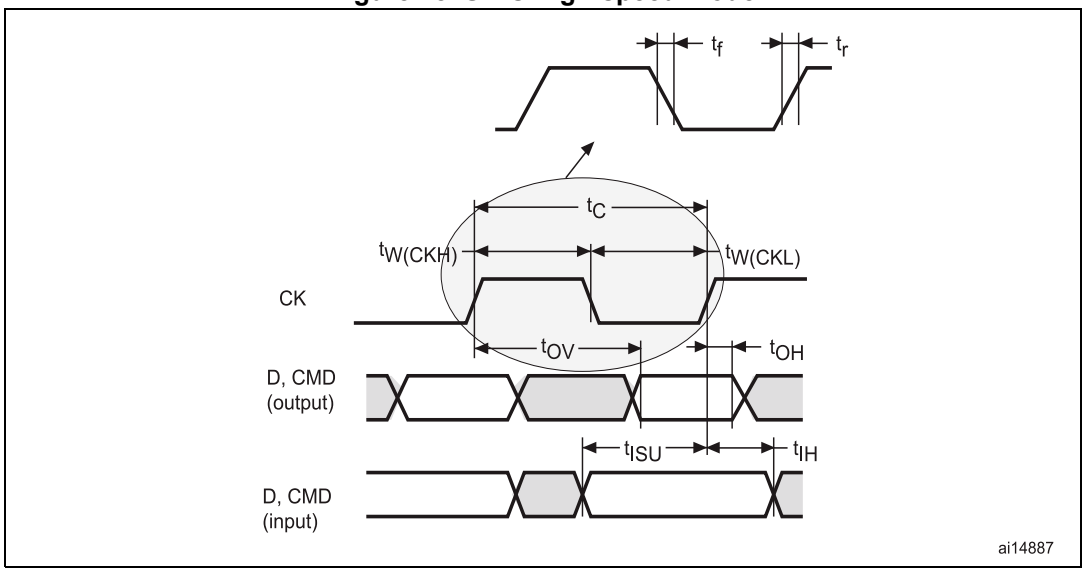


Figure 79. SD default mode

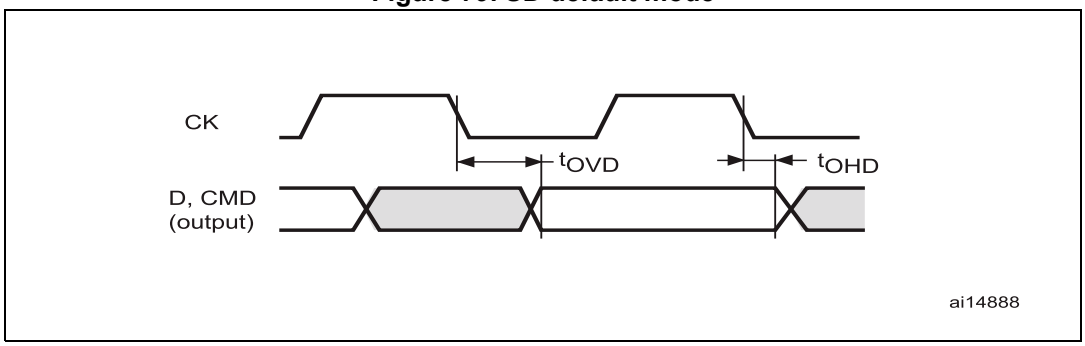


Table 108. Dynamic characteristics: SD / MMC characteristics(1)(2)

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
f_{PP}	Clock frequency in data transfer mode		0		48	MHz
-	SDIO_CK/fPCLK2 frequency ratio		-	-	8/3	-
$t_{W(CKL)}$	Clock low time	fpp =48 MHz	8.5	9	-	ns
$t_{W(CKH)}$	Clock high time	fpp =48 MHz	8.3	10	-	
CMD, D inputs (referenced to CK) in MMC and SD HS mode						
t_{ISU}	Input setup time HS	fpp =48 MHz	3.5	-	-	ns
t_{IH}	Input hold time HS	fpp =48 MHz	0	-	-	
CMD, D outputs (referenced to CK) in MMC and SD HS mode						
t_{OV}	Output valid time HS	fpp =48 MHz	-	4.5	7	ns
t_{OH}	Output hold time HS	fpp =48 MHz	3	-	-	
CMD, D inputs (referenced to CK) in SD default mode						
tISUD	Input setup time SD	fpp =24 MHz	1.5	-	-	ns
tIHD	Input hold time SD	fpp =24 MHz	0.5	-	-	
CMD, D outputs (referenced to CK) in SD default mode						
tOVD	Output valid default time SD	fpp =24 MHz	-	4.5	6.5	ns
tOHD	Output hold default time SD	fpp =24 MHz	3.5	-	-	

1. Guaranteed by characterization results.
2. V_{DD} = 2.7 to 3.6 V.

6.3.30 RTC characteristics

Table 109. RTC characteristics

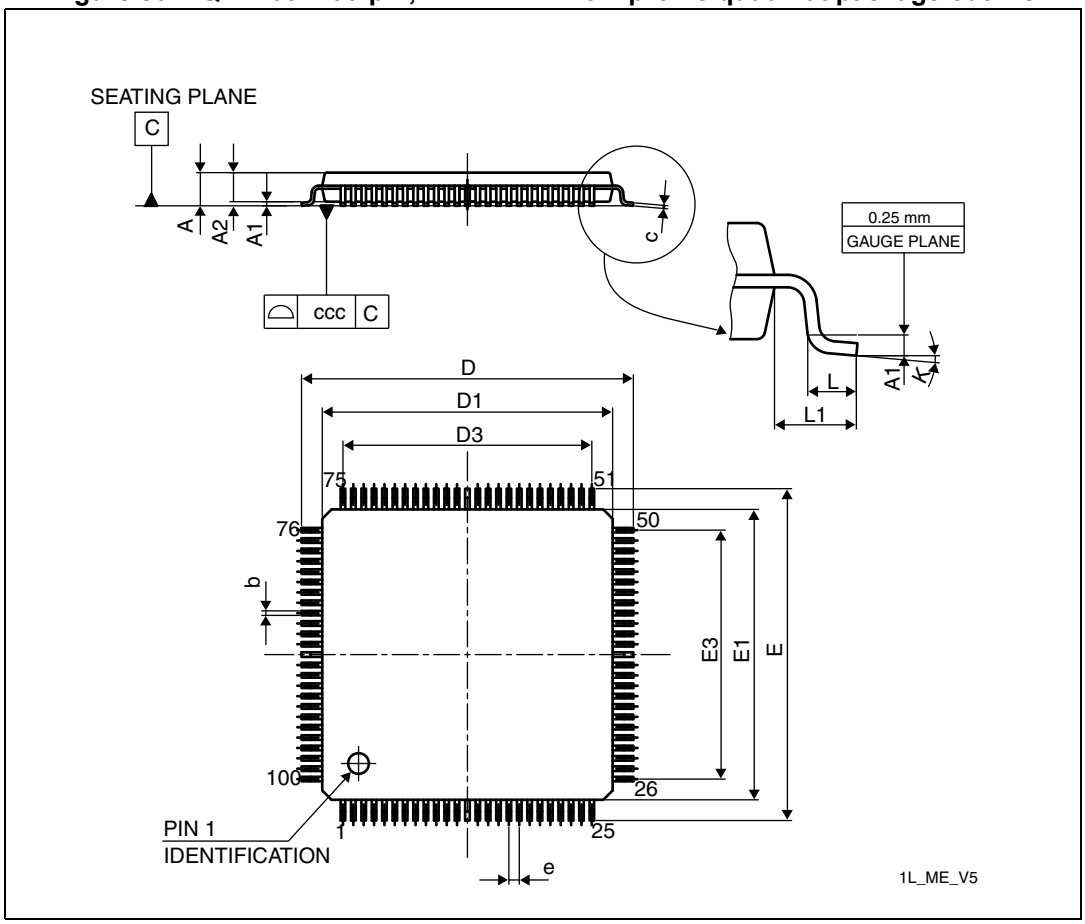
Symbol	Parameter	Conditions	Min	Max
-	f_{PCLK1}/RTCCLK frequency ratio	Any read/write operation from/to an RTC register	4	-

7 Package information

In order to meet environmental requirements, ST offers these devices in different grades of ECOPACK® packages, depending on their level of environmental compliance. ECOPACK® specifications, grade definitions and product status are available at: *www.st.com*. ECOPACK® is an ST trademark.

7.1 LQFP100 package information

Figure 80. LQFP100 -100-pin, 14 x 14 mm low-profile quad flat package outline



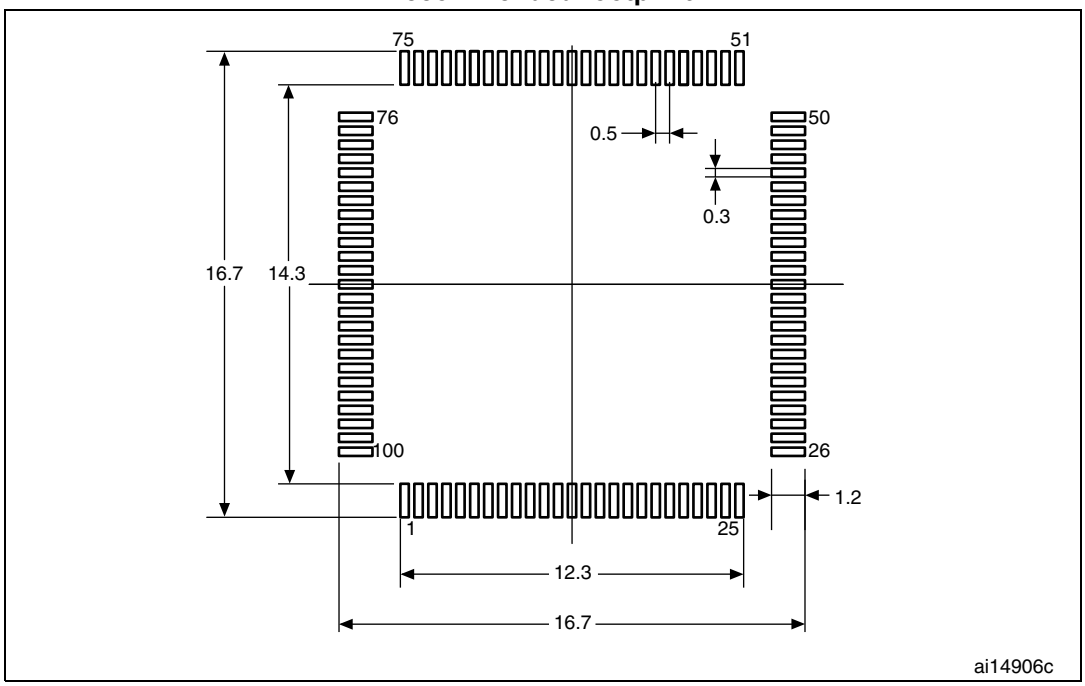
1. Drawing is not to scale.

Table 110. LQPF100 100-pin, 14 x 14 mm low-profile quad flat package mechanical data

Symbol	millimeters			inches[1]		
	Min	Typ	Max	Min	Typ	Max
A	-	-	1.600	-	-	0.0630
A1	0.050	-	0.150	0.0020	-	0.0059
A2	1.350	1.400	1.450	0.0531	0.0551	0.0571
b	0.170	0.220	0.270	0.0067	0.0087	0.0106
c	0.090	-	0.200	0.0035	-	0.0079
D	15.800	16.000	16.200	0.6220	0.6299	0.6378
D1	13.800	14.000	14.200	0.5433	0.5512	0.5591
D3	-	12.000	-	-	0.4724	-
E	15.800	16.000	16.200	0.6220	0.6299	0.6378
E1	13.800	14.000	14.200	0.5433	0.5512	0.5591
E3	-	12.000	-	-	0.4724	-
e	-	0.500	-	-	0.0197	-
L	0.450	0.600	0.750	0.0177	0.0236	0.0295
L1	-	1.000	-	-	0.0394	-
k	0.0°	3.5°	7.0°	0.0°	3.5°	7.0°
ccc	-	-	0.080	-	-	0.0031

1. Values in inches are converted from mm and rounded to 4 decimal digits.

Figure 81. LQPF100 - 100-pin, 14 x 14 mm low-profile quad flat recommended footprint



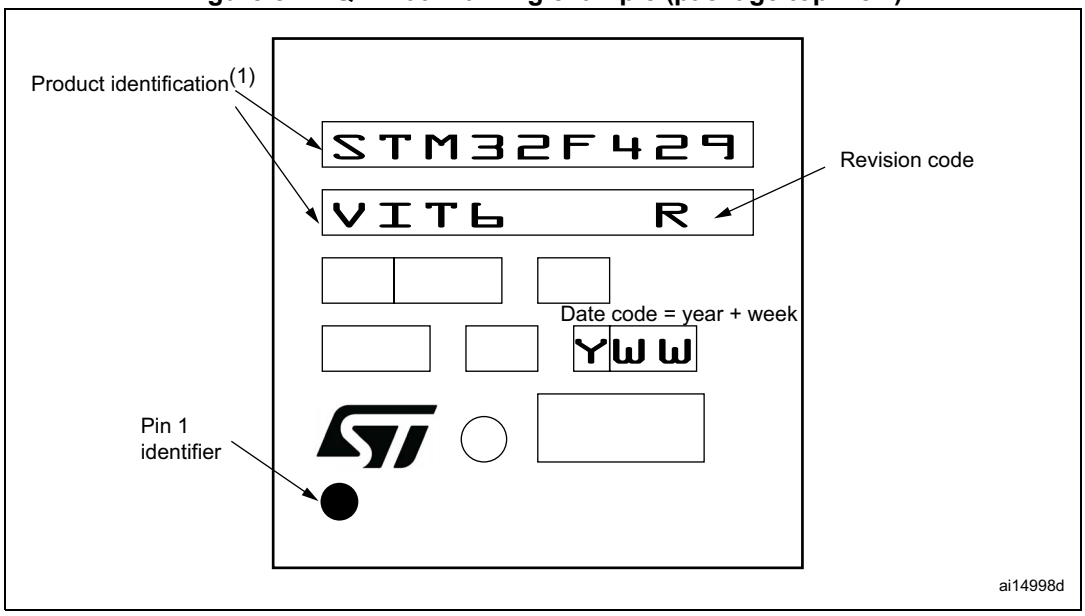
1. Dimensions are expressed in millimeters.

Device marking for LQFP100

The following figure gives an example of topside marking orientation versus pin 1 identifier location.

Other optional marking or inset/upset marks, which depend on assembly location, are not indicated below.

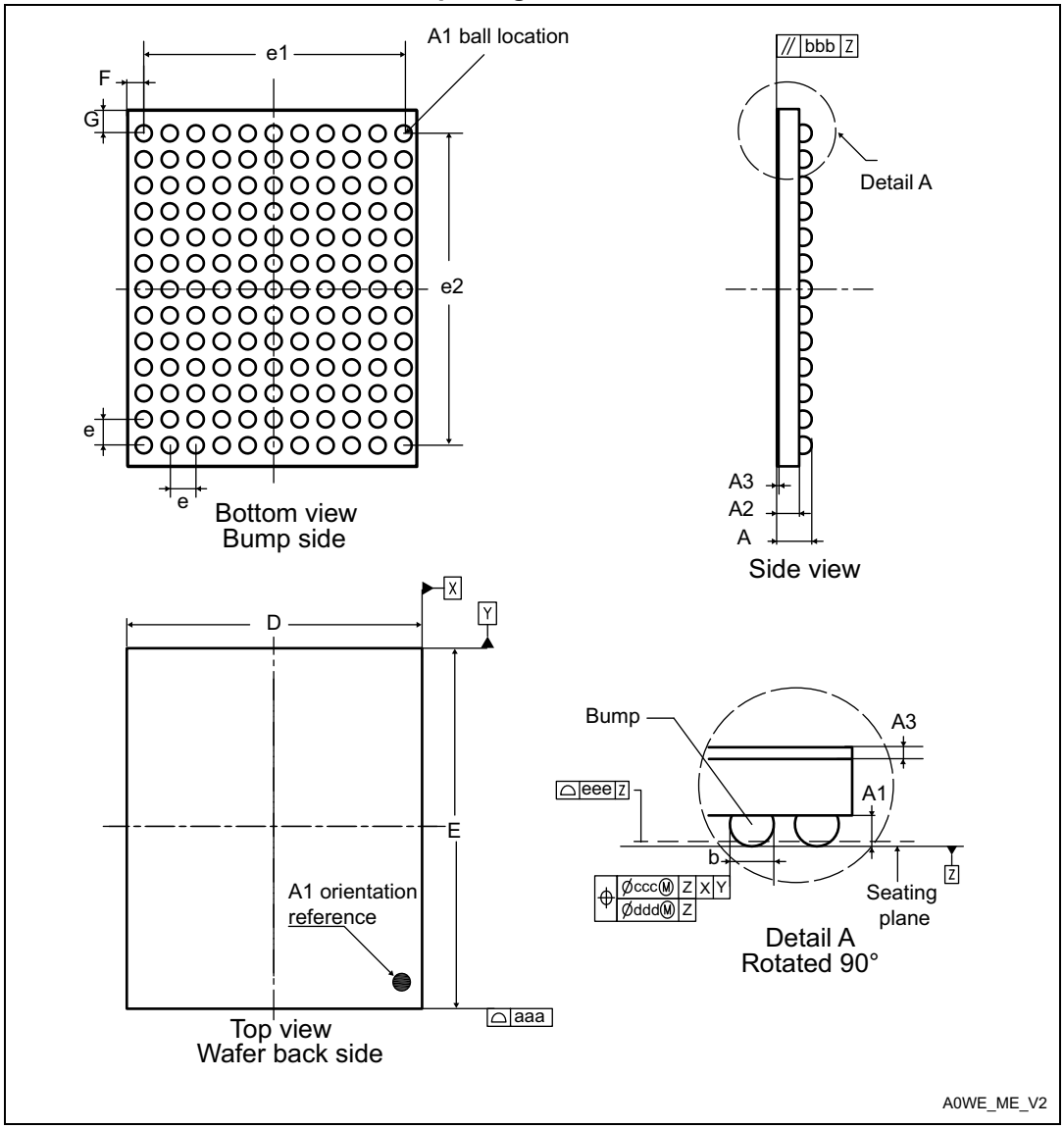
Figure 82. LQFP100 marking example (package top view)



1. Parts marked as “ES”, “E” or accompanied by an Engineering Sample notification letter, are not yet qualified and therefore not yet ready to be used in production and any consequences deriving from such usage will not be at ST charge. In no event, ST will be liable for any customer usage of these engineering samples in production. ST Quality has to be contacted prior to any decision to use these Engineering Samples to run qualification activity.

7.2 WLCSP143 package information

Figure 83. WLCSP143 - 143-ball, 4.521x 5.547 mm, 0.4 mm pitch wafer level chip scale package outline



1. Drawing is not to scale.

Table 111. WLCSP143 - 143-ball, 4.521x 5.547 mm, 0.4 mm pitch wafer level chip scale package mechanical data

Symbol	millimeters			inches[1]		
	Min	Typ	Max	Min	Typ	Max
A	0.525	0.555	0.585	0.0207	0.0219	0.0230
A1	0.155	0.175	0.195	-	0.0069	-
A2	-	0.380	-	-	0.0150	-
A3[2]	-	0.025	-	-	0.0010	-
b[3]	0.220	0.250	0.280	0.0087	0.0098	0.0110
D	4.486	4.521	4.556	0.1766	0.1780	0.1794
E	5.512	5.547	5.582	0.2170	0.2184	0.2198
e	-	0.400	-	-	0.0157	-
e1	-	4.000	-	-	0.1575	-
e2	-	4.800	-	-	0.1890	-
F	-	0.2605	-	-	0.0103	-
G	-	0.3735	-	-	0.0147	-
aaa	-	-	0.100	-	-	0.0039
bbb	-	-	0.100	-	-	0.0039
ccc	-	-	0.100	-	-	0.0039
ddd	-	-	0.050	-	-	0.0020
eee	-	-	0.050	-	-	0.0020

1. Values in inches are converted from mm and rounded to 4 decimal digits.
2. Back side coating.
3. Dimension is measured at the maximum bump diameter parallel to primary datum Z.

Figure 84. WLCSP143 - 143-ball, 4.521x 5.547 mm, 0.4 mm pitch wafer level chip scale recommended footprint

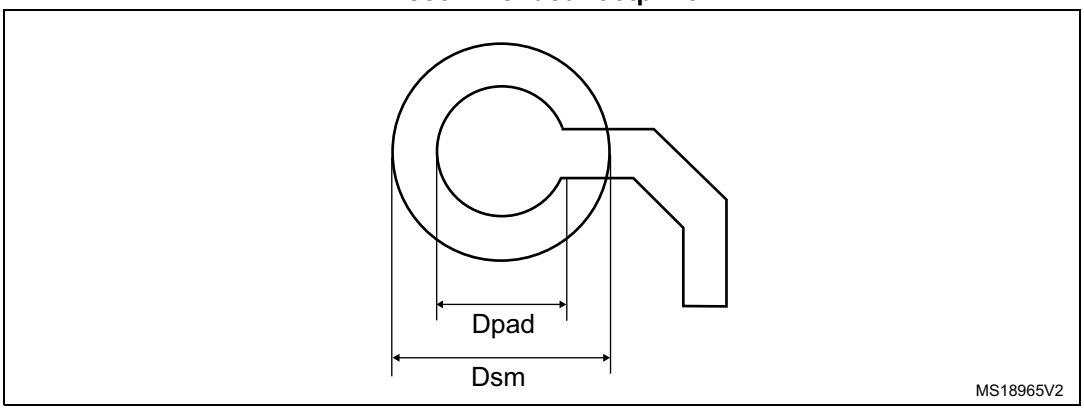


Table 112. WLCSP143 recommended PCB design rules (0.4 mm pitch)

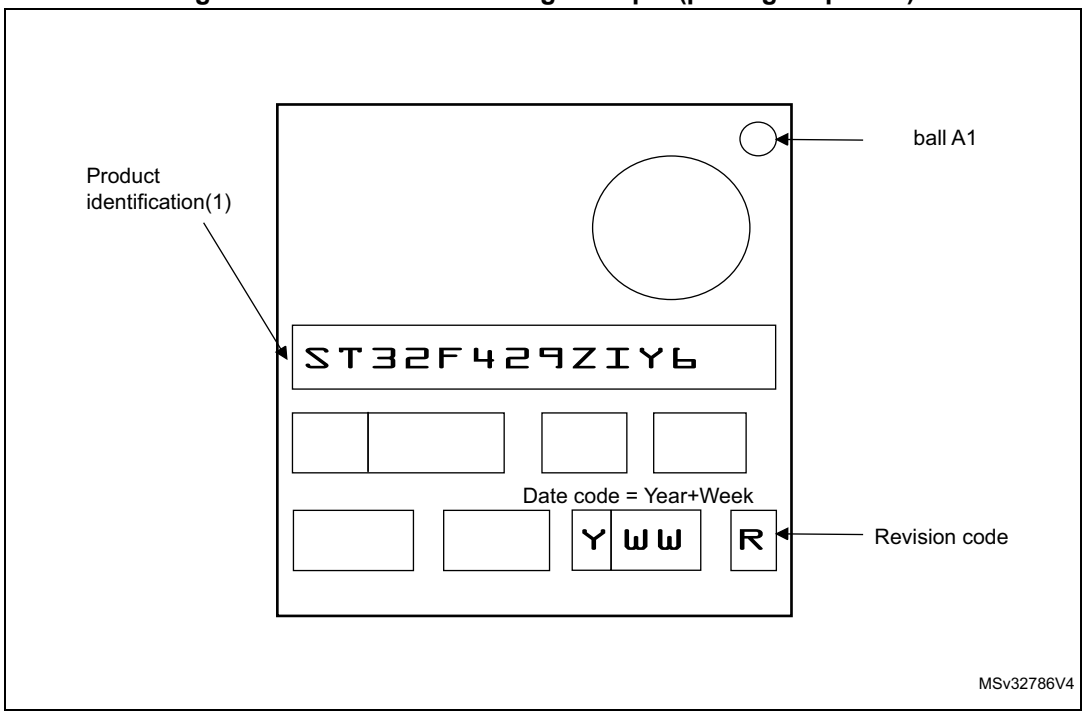
Dimension	Recommended values
Pitch	0.4
Dpad	260 µm max. (circular)
	220 µm recommended
Dsm	300 µm min. (for 260 µm diameter pad)
PCB pad design	Non-solder mask defined via underbump allowed.

Device marking for WLCSP143

The following figure gives an example of topside marking orientation versus ball A 1 identifier location.

Other optional marking or inset/upset marks, which depend on assembly location, are not indicated below.

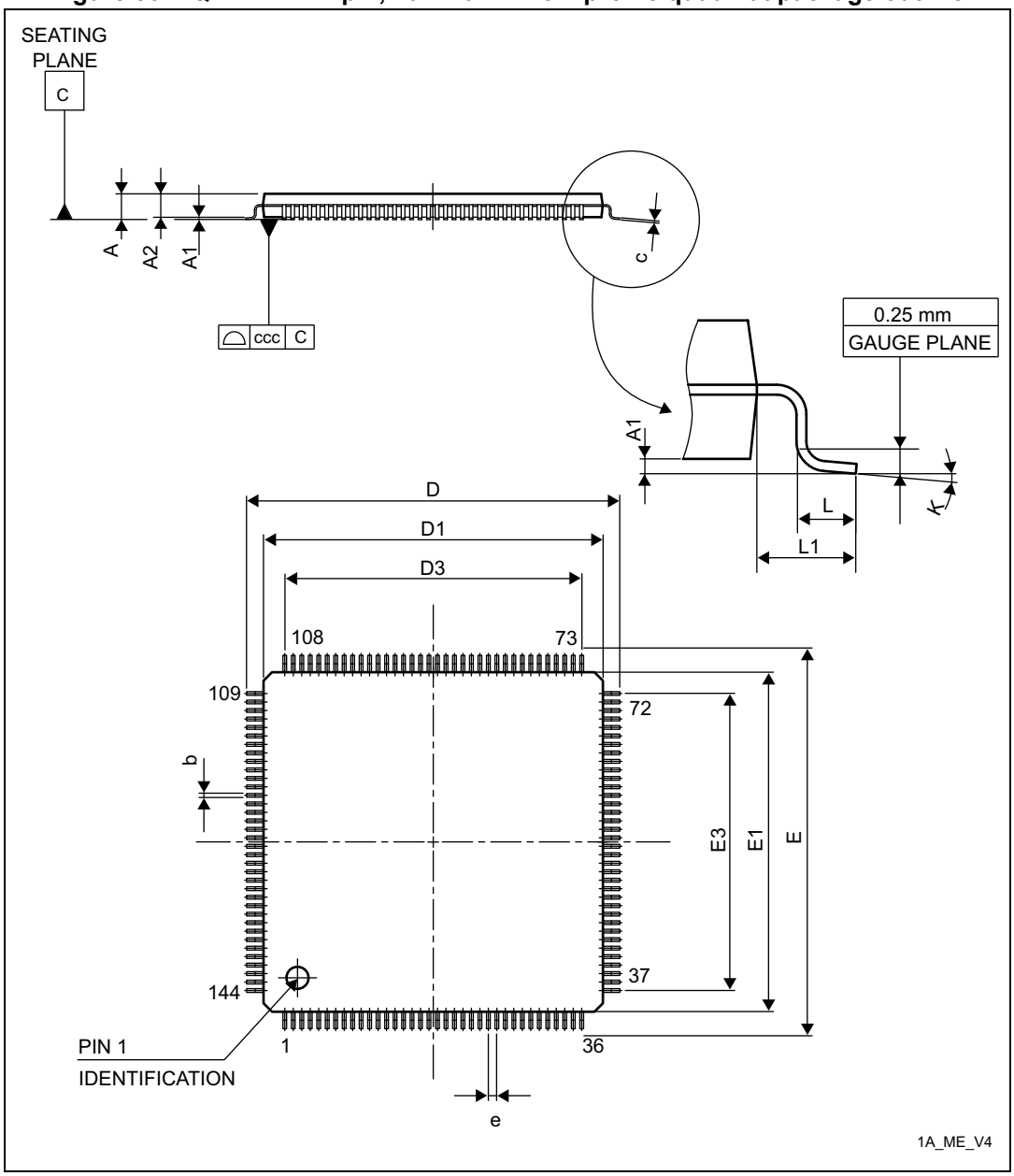
Figure 85. WLCSP143 marking example (package top view)



1. Parts marked as “ES”, “E” or accompanied by an Engineering Sample notification letter, are not yet qualified and therefore not yet ready to be used in production and any consequences deriving from such usage will not be at ST charge. In no event, ST will be liable for any customer usage of these engineering samples in production. ST Quality has to be contacted prior to any decision to use these Engineering Samples to run qualification activity.

7.3 LQFP144 package information

Figure 86. LQFP144-144-pin, 20 x 20 mm low-profile quad flat package outline



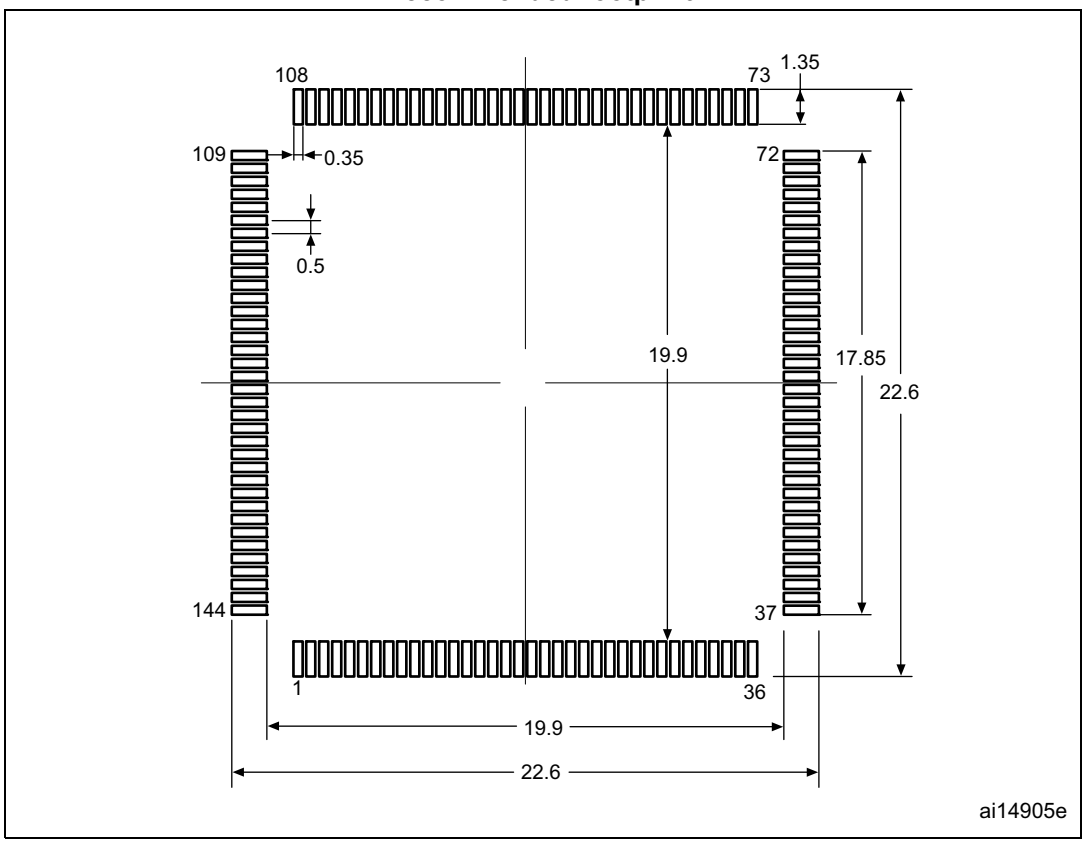
1. Drawing is not to scale.

Table 113. LQFP144 - 144-pin, 20 x 20 mm low-profile quad flat package mechanical data

Symbol	millimeters			inches[1]		
	Min	Typ	Max	Min	Typ	Max
A	-	-	1.600	-	-	0.0630
A1	0.050	-	0.150	0.0020	-	0.0059
A2	1.350	1.400	1.450	0.0531	0.0551	0.0571
b	0.170	0.220	0.270	0.0067	0.0087	0.0106
c	0.090	-	0.200	0.0035	-	0.0079
D	21.800	22.000	22.200	0.8583	0.8661	0.874
D1	19.800	20.000	20.200	0.7795	0.7874	0.7953
D3	-	17.500	-	-	0.689	-
E	21.800	22.000	22.200	0.8583	0.8661	0.8740
E1	19.800	20.000	20.200	0.7795	0.7874	0.7953
E3	-	17.500	-	-	0.6890	-
e	-	0.500	-	-	0.0197	-
L	0.450	0.600	0.750	0.0177	0.0236	0.0295
L1	-	1.000	-	-	0.0394	-
k	0°	3.5°	7°	0°	3.5°	7°
ccc	-	-	0.080	-	-	0.0031

1. Values in inches are converted from mm and rounded to 4 decimal digits.

Figure 87. LQPF144- 144-pin,20 x 20 mm low-profile quad flat package recommended footprint



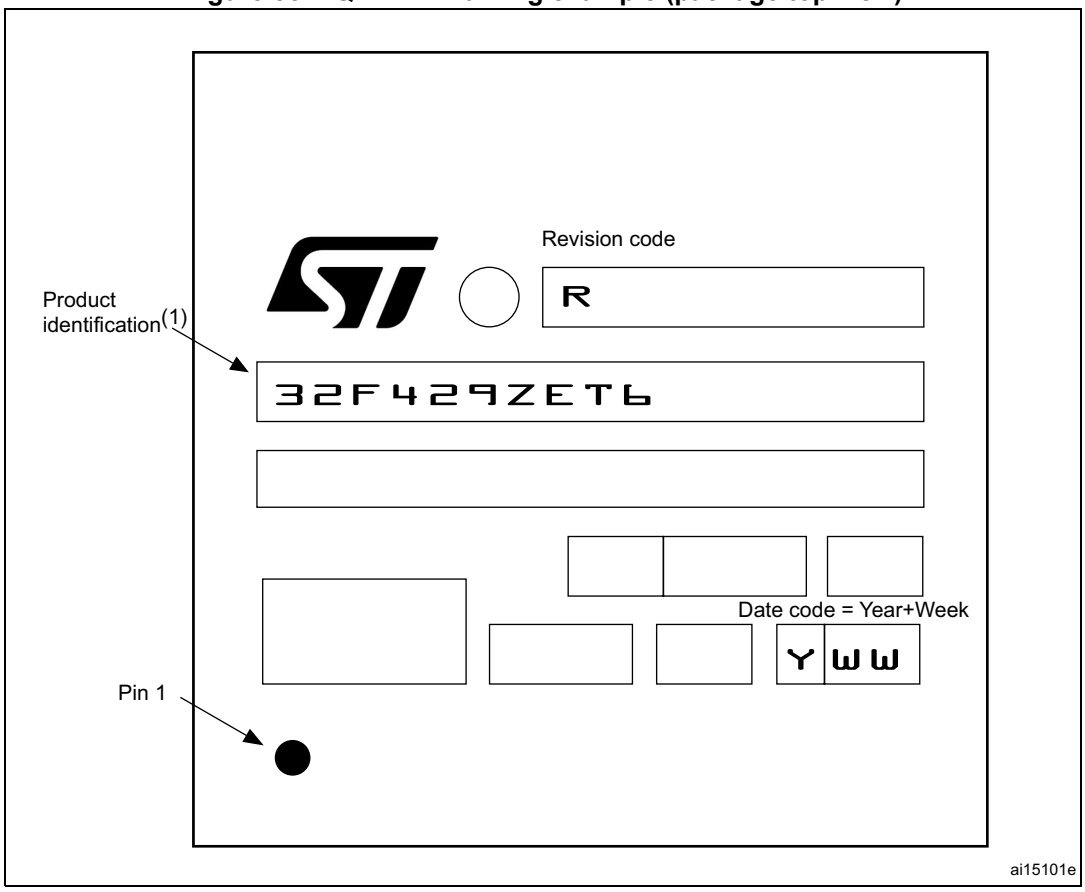
1. Dimensions are expressed in millimeters.

Device marking for LQFP144

The following figure gives an example of topside marking orientation versus pin 1 identifier location.

Other optional marking or inset/upset marks, which depend on assembly location, are not indicated below.

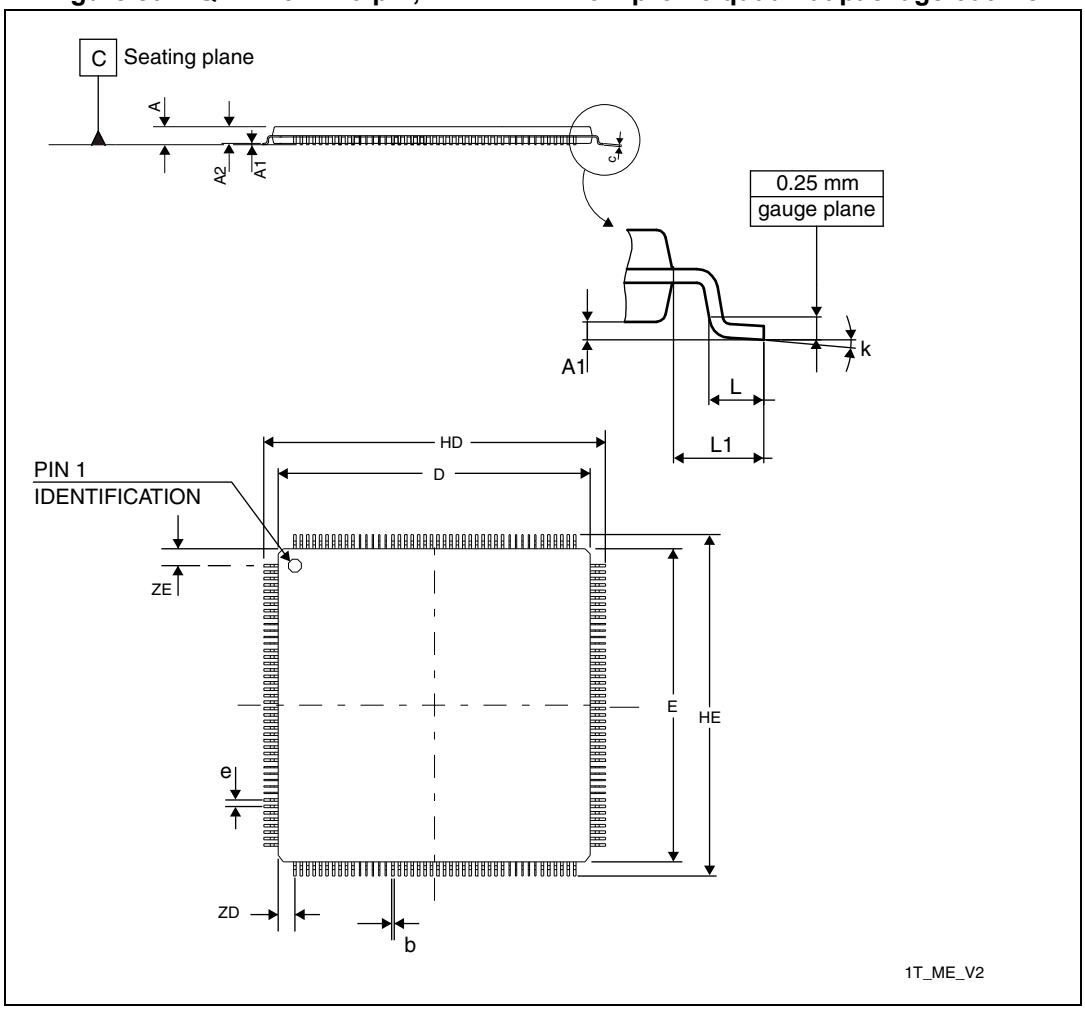
Figure 88. LQFP144 marking example (package top view)



1. Parts marked as “ES”, “E” or accompanied by an Engineering Sample notification letter, are not yet qualified and therefore not yet ready to be used in production and any consequences deriving from such usage will not be at ST charge. In no event, ST will be liable for any customer usage of these engineering samples in production. ST Quality has to be contacted prior to any decision to use these Engineering Samples to run qualification activity.

7.4 LQFP176 package information

Figure 89. LQFP176 - 176-pin, 24 x 24 mm low-profile quad flat package outline



1. Drawing is not to scale.

Table 114. LQFP176 - 176-pin, 24 x 24 mm low-profile quad flat package mechanical data

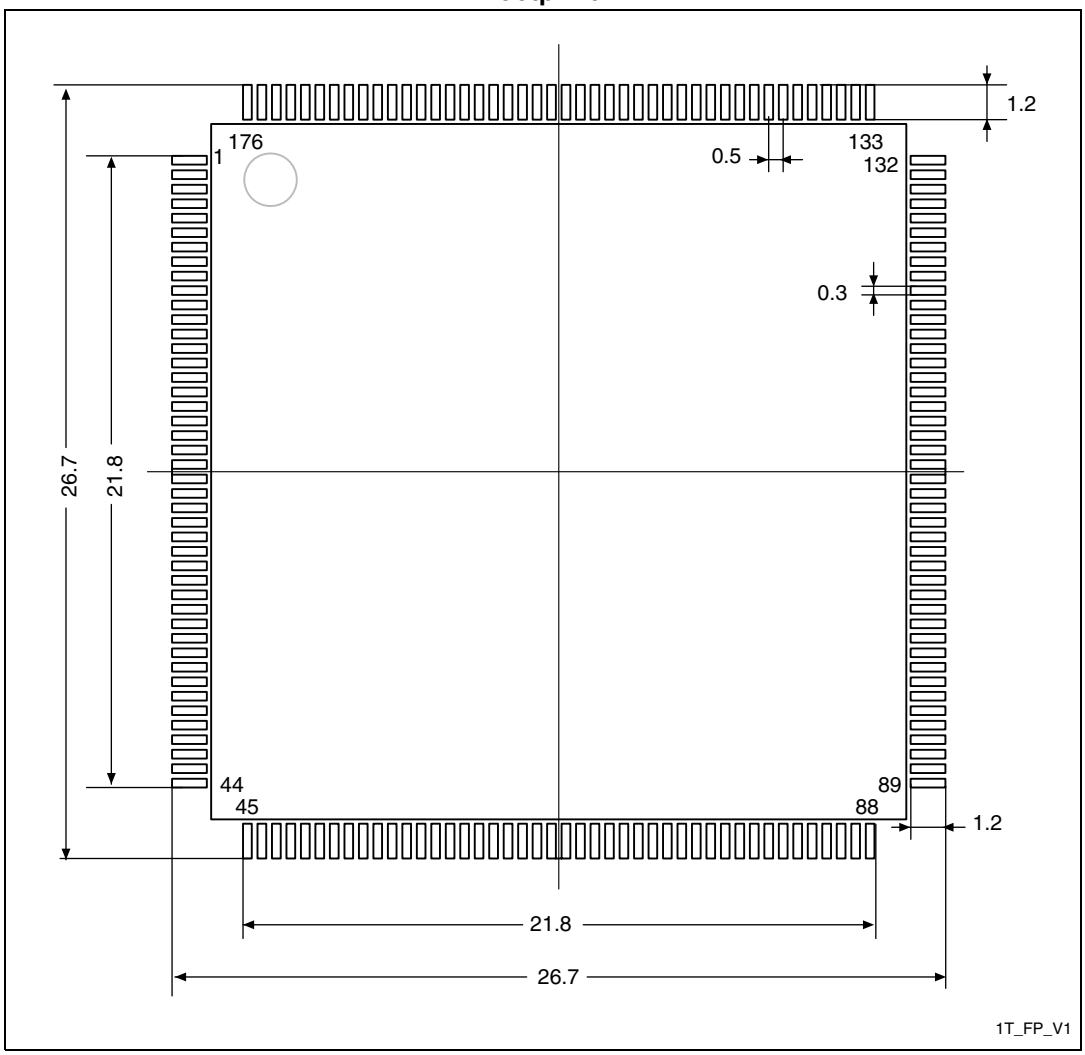
Symbol	millimeters			inches(1)		
	Min	Typ	Max	Min	Typ	Max
A	-	-	1.600	-	-	0.0630
A1	0.050	-	0.150	0.0020	-	0.0059
A2	1.350	-	1.450	0.0531	-	0.0571
b	0.170	-	0.270	0.0067	-	0.0106
c	0.090	-	0.200	0.0035	-	0.0079
D	23.900	-	24.100	0.9409	-	0.9488
HD	25.900	-	26.100	1.0197	-	1.0276

Table 114. LQFP176 - 176-pin, 24 x 24 mm low-profile quad flat package mechanical data (continued)

Symbol	millimeters			inches[1]		
	Min	Typ	Max	Min	Typ	Max
ZD	-	1.250	-	-	0.0492	-
E	23.900	-	24.100	0.9409	-	0.9488
HE	25.900	-	26.100	1.0197	-	1.0276
ZE	-	1.250	-	-	0.0492	-
e	-	0.500	-	-	0.0197	-
L[2]	0.450	-	0.750	0.0177	-	0.0295
L1	-	1.000	-	-	0.0394	-
k	0°	-	7°	0°	-	7°
ccc	-	-	0.080	-	-	0.0031

1. Values in inches are converted from mm and rounded to 4 decimal digits.
2. L dimension is measured at gauge plane at 0.25 mm above the seating plane.

Figure 90. LQFP176 - 176-pin, 24 x 24 mm low profile quad flat recommended footprint



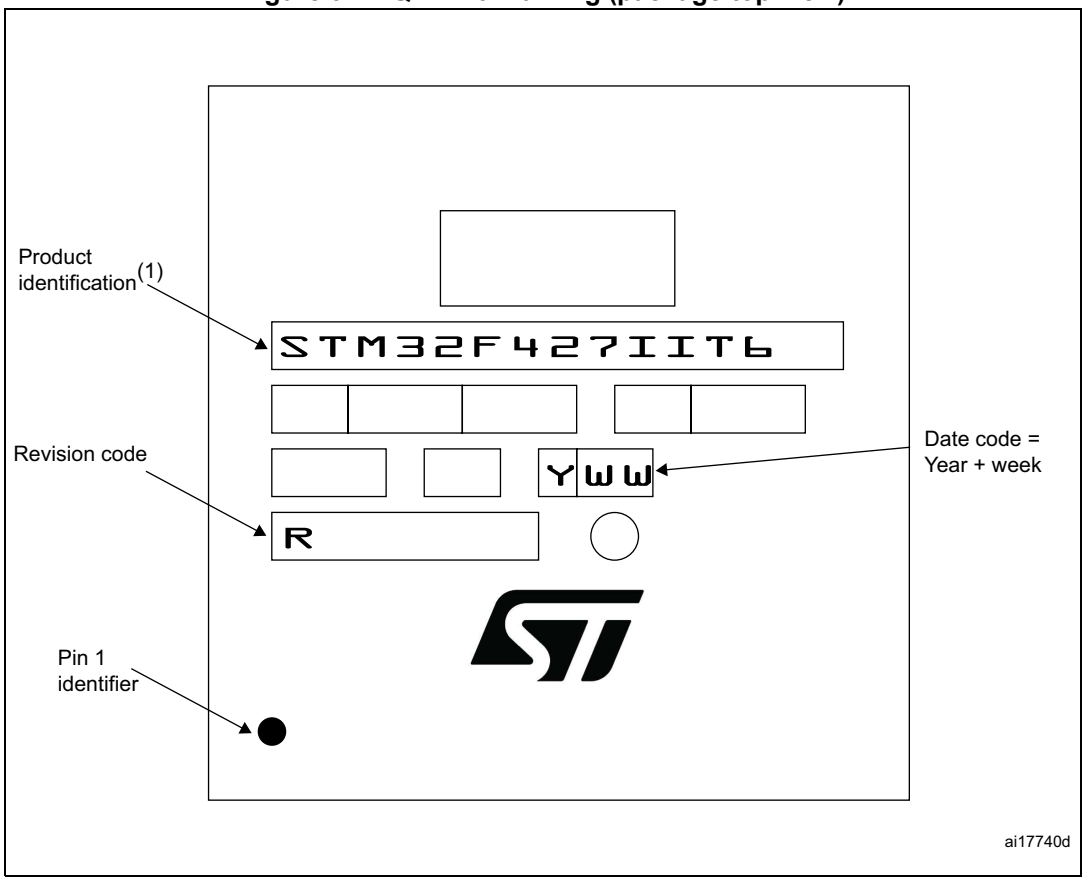
1. Dimensions are expressed in millimeters.

Device marking for LQFP176

The following figure gives an example of topside marking orientation versus pin 1 identifier location.

Other optional marking or inset/upset marks, which depend on assembly location, are not indicated below.

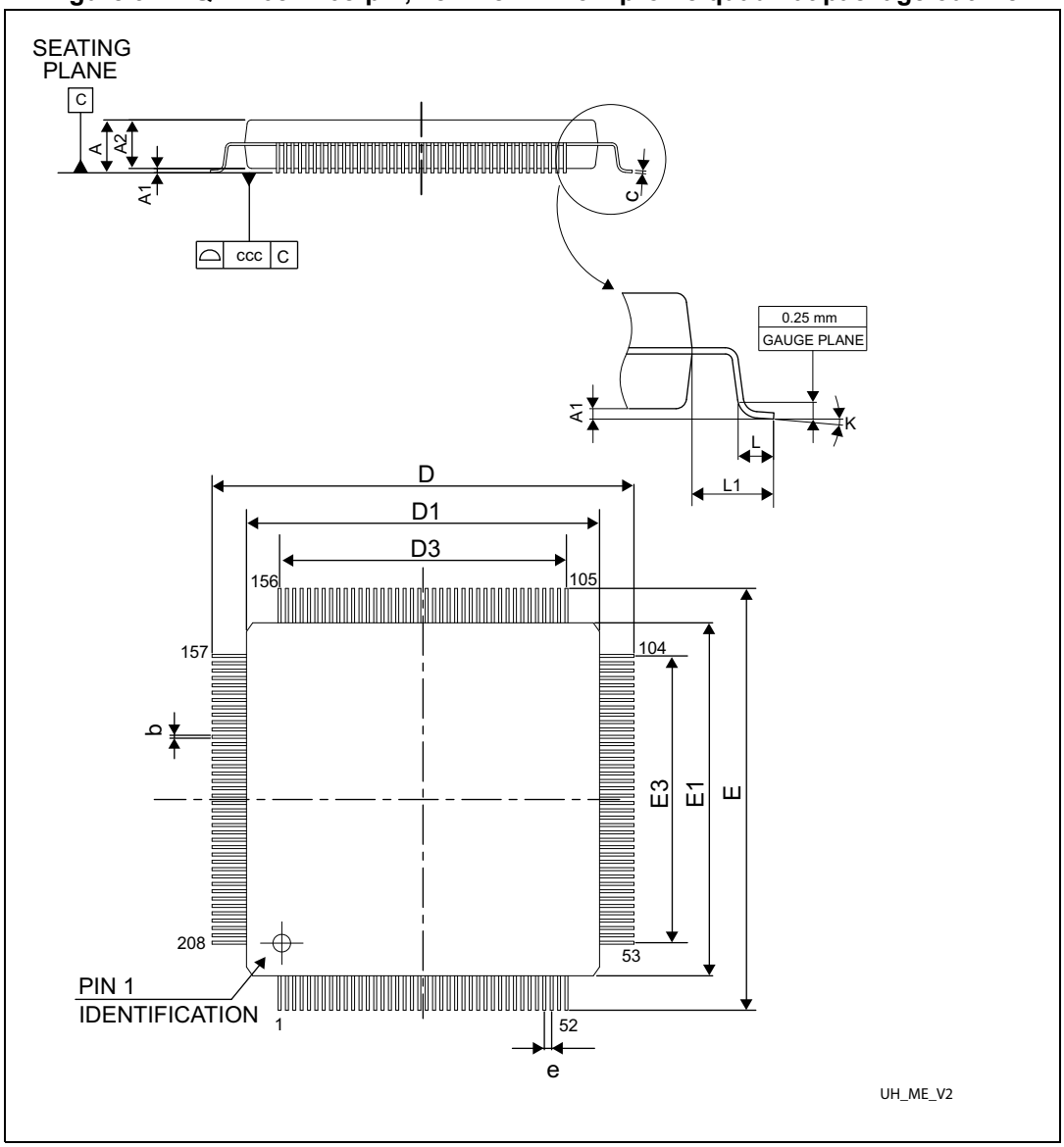
Figure 91. LQFP176 marking (package top view)



1. Parts marked as “ES”, “E” or accompanied by an Engineering Sample notification letter, are not yet qualified and therefore not yet ready to be used in production and any consequences deriving from such usage will not be at ST charge. In no event, ST will be liable for any customer usage of these engineering samples in production. ST Quality has to be contacted prior to any decision to use these Engineering Samples to run qualification activity.

7.5 LQFP208 package information

Figure 92. LQFP208 - 208-pin, 28 x 28 mm low-profile quad flat package outline



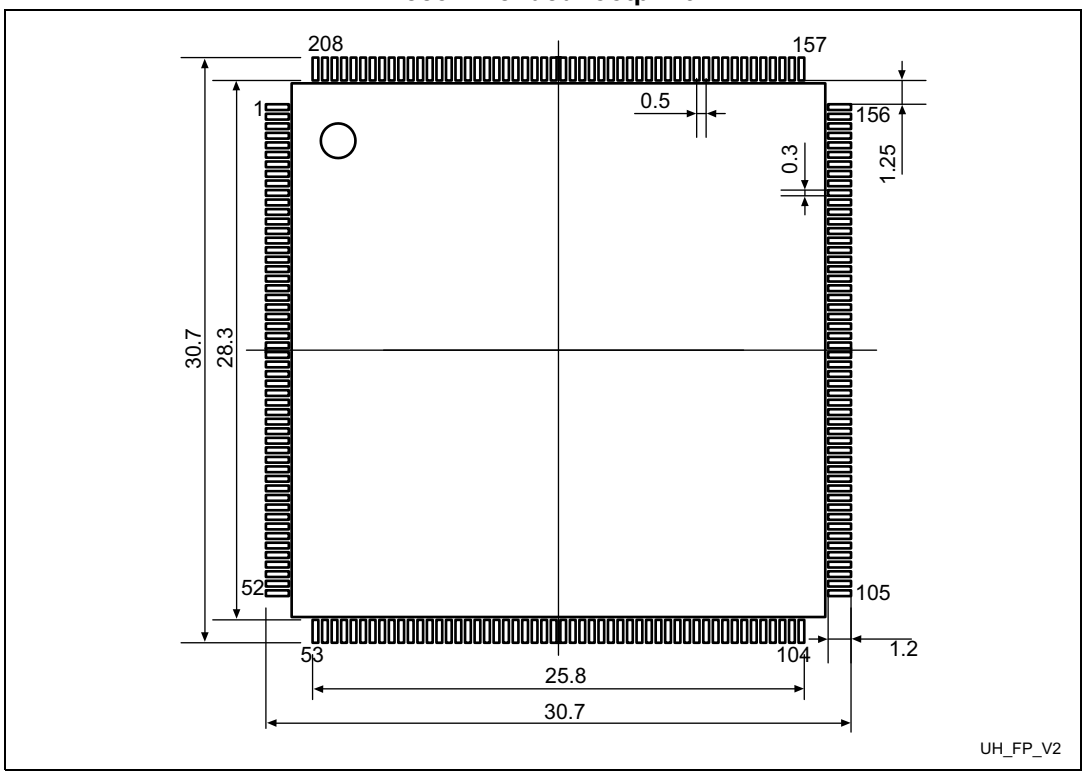
1. Drawing is not to scale.

Table 115. LQFP208 - 208-pin, 28 x 28 mm low-profile quad flat package mechanical data

Symbol	millimeters			inches[1]		
	Min	Typ	Max	Min	Typ	Max
A	-	-	1.600	--	-	0.0630
A1	0.050	-	0.150	0.0020	-	0.0059
A2	1.350	1.400	1.450	0.0531	0.0551	0.0571
b	0.170	0.220	0.270	0.0067	0.0087	0.0106
c	0.090	-	0.200	0.0035	-	0.0079
D	29.800	30.000	30.200	1.1732	1.1811	1.1890
D1	27.800	28.000	28.200	1.0945	1.1024	1.1102
D3	-	25.500	-	-	1.0039	-
E	29.800	30.000	30.200	1.1732	1.1811	1.1890
E1	27.800	28.000	28.200	1.0945	1.1024	1.1102
E3	-	25.500	-	-	1.0039	-
e	-	0.500	-	-	0.0197	-
L	0.450	0.600	0.750	0.0177	0.0236	0.0295
L1	-	1.000	-	-	0.0394	-
k	0°	3.5°	7.0°	0°	3.5°	7.0°
ccc	-	-	0.080	-	-	0.0031

1. Values in inches are converted from mm and rounded to 4 decimal digits.

Figure 93. LQFP208 - 208-pin, 28 x 28 mm low-profile quad flat package recommended footprint



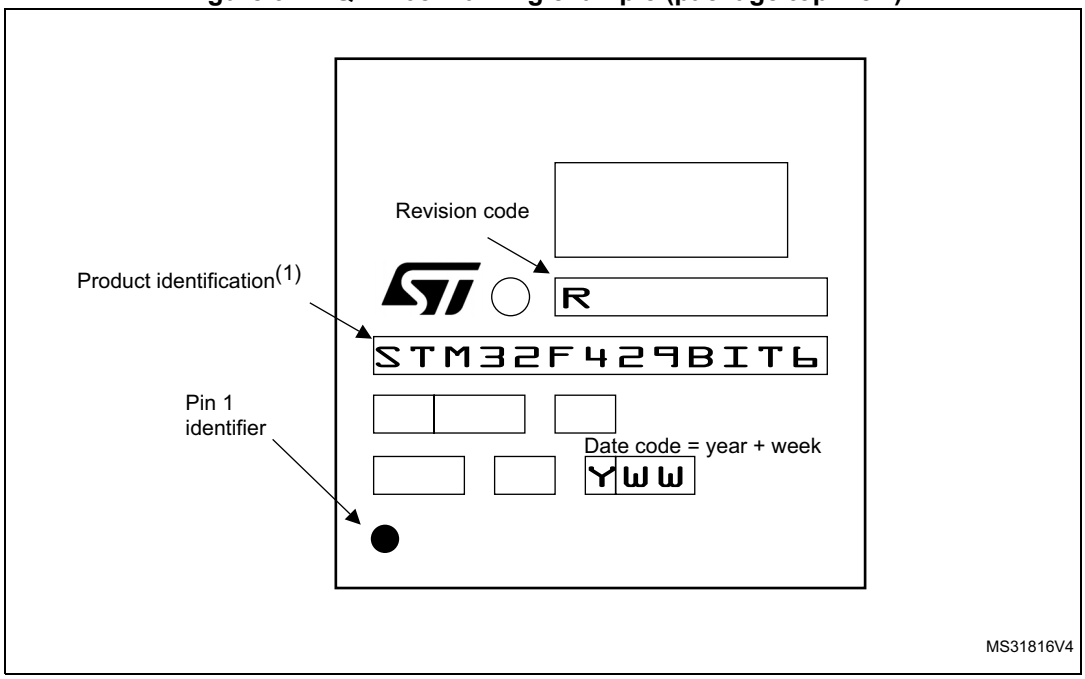
1. Dimensions are expressed in millimeters.

Device marking for LQFP208

The following figure gives an example of topside marking orientation versus pin 1 identifier location.

Other optional marking or inset/upset marks, which depend on assembly location, are not indicated below.

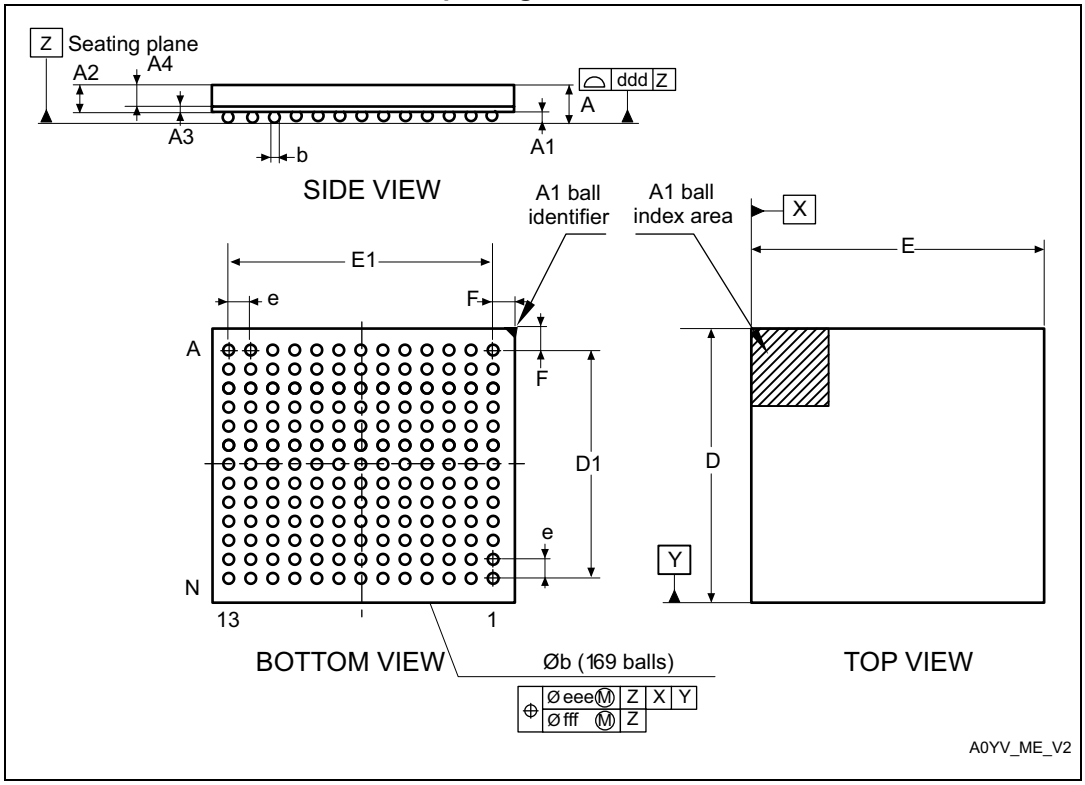
Figure 94. LQFP208 marking example (package top view)



1. Parts marked as “ES”, “E” or accompanied by an Engineering Sample notification letter, are not yet qualified and therefore not yet ready to be used in production and any consequences deriving from such usage will not be at ST charge. In no event, ST will be liable for any customer usage of these engineering samples in production. ST Quality has to be contacted prior to any decision to use these Engineering Samples to run qualification activity.

7.6 UFBGA169 package information

Figure 95. UFBGA169 - 169-ball 7 x 7 mm 0.50 mm pitch, ultra fine pitch ball grid array package outline



1. Drawing is not to scale.

Table 116. UFBGA169 - 169-ball 7 x 7 mm 0.50 mm pitch, ultra fine pitch ball grid array package mechanical data

Symbol	millimeters			inches(1)		
	Min	Typ	Max	Min	Typ	Max
A	0.460	0.530	0.600	0.0181	0.0209	0.0236
A1	0.050	0.080	0.110	0.0020	0.0031	0.0043
A2	0.400	0.450	0.500	0.0157	0.0177	0.0197
A3	-	0.130	-	-	0.0051	-
A4	0.270	0.320	0.370	0.0106	0.0126	0.0146
b	0.230	0.280	0.330	0.0091	0.0110	0.0130
D	6.950	7.000	7.050	0.2736	0.2756	0.2776
D1	5.950	6.000	6.050	0.2343	0.2362	0.2382
E	6.950	7.000	7.050	0.2736	0.2756	0.2776
E1	5.950	6.000	6.050	0.2343	0.2362	0.2382
e	-	0.500	-	-	0.0197	-

Table 116. UFBGA169 - 169-ball 7 x 7 mm 0.50 mm pitch, ultra fine pitch ball grid array package mechanical data (continued)

Symbol	millimeters			inches[1]		
	Min	Typ	Max	Min	Typ	Max
F	0.450	0.500	0.550	0.0177	0.0197	0.0217
ddd	-	-	0.100	-	-	0.0039
eee	-	-	0.150	-	-	0.0059
fff	-	-	0.050	-	-	0.0020

1. Values in inches are converted from mm and rounded to 4 decimal digits.

Figure 96. UFBGA169 - 169-ball, 7 x 7 mm, 0.50 mm pitch, ultra fine pitch ball grid array recommended footprint

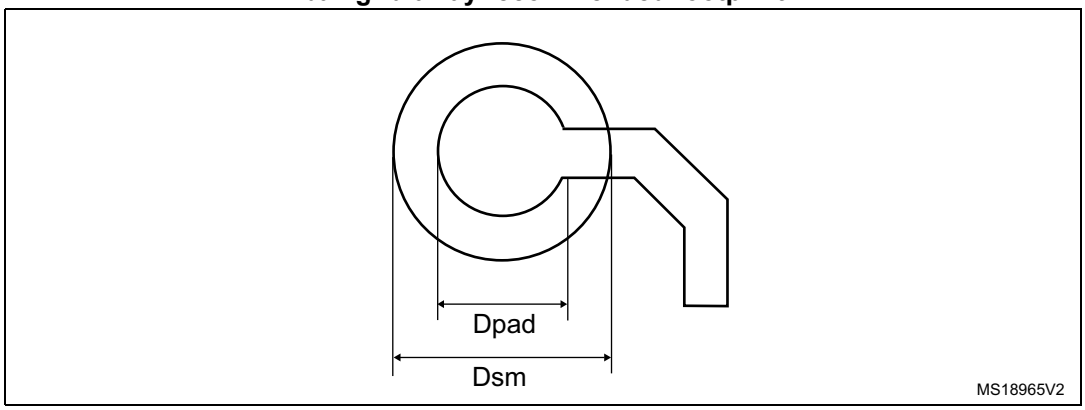


Table 117. UFBGA169 recommended PCB design rules (0.5 mm pitch BGA)

Dimension	Recommended values
Pitch	0.5
Dpad	0.27 mm
Dsm	0.35 mm typ. (depends on the soldermask registration tolerance)
Solder paste	0.27 mm aperture diameter.

Note: Non-solder mask defined (NSMD) pads are recommended.

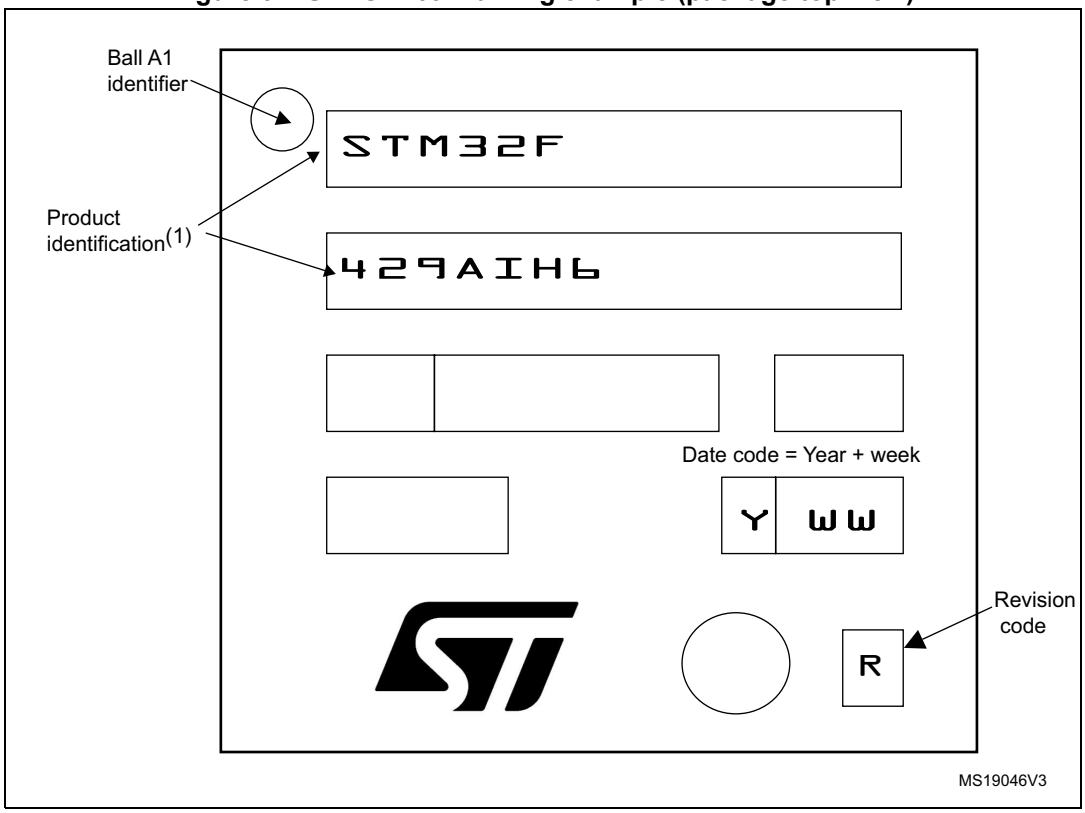
4 to 6 mils solder paste screen printing process.

Device marking for UFBGA169

The following figure gives an example of topside marking orientation versus ball A1 identifier location.

Other optional marking or inset/upset marks, which depend on assembly location, are not indicated below.

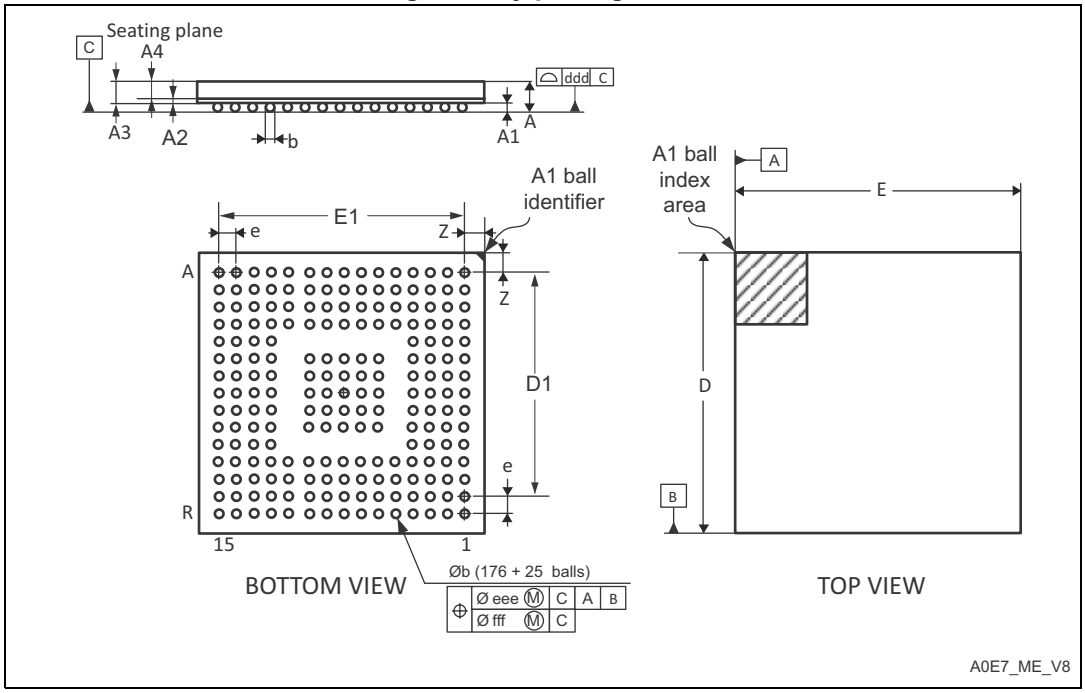
Figure 97. UFBGA169 marking example (package top view)



1. Parts marked as “ES”, “E” or accompanied by an Engineering Sample notification letter, are not yet qualified and therefore not yet ready to be used in production and any consequences deriving from such usage will not be at ST charge. In no event, ST will be liable for any customer usage of these engineering samples in production. ST Quality has to be contacted prior to any decision to use these Engineering Samples to run qualification activity.

7.7 UFBGA176+25 package information

Figure 98. UFBGA176+25 - ball 10 x 10 mm, 0.65 mm pitch ultra thin fine pitch ball grid array package outline



1. Drawing is not to scale.

Table 118. UFBGA176+25 - ball, 10 x 10 mm, 0.65 mm pitch, ultra fine pitch ball grid array package mechanical data

Symbol	millimeters			inches(1)		
	Min.	Typ.	Max.	Min.	Typ.	Max.
A	-	-	0.600	-	-	0.0236
A1	-	-	0.110	-	-	0.0043
A2	-	0.130	-	-	0.0051	-
A3	-	0.450	-	-	0.0177	-
A4	-	0.320	-	-	0.0126	-
b	0.240	0.290	0.340	0.0094	0.0114	0.0134
D	9.850	10.000	10.150	0.3878	0.3937	0.3996
D1	-	9.100	-	-	0.3583	-
E	9.850	10.000	10.150	0.3878	0.3937	0.3996
E1	-	9.100	-	-	0.3583	-
e	-	0.650	-	-	0.0256	-
Z	-	0.450	-	-	0.0177	-
ddd	-	-	0.080	-	-	0.0031

Table 118. UFBGA176+25 - ball, 10 x 10 mm, 0.65 mm pitch, ultra fine pitch ball grid array package mechanical data (continued)

Symbol	millimeters			inches[1]		
	Min.	Typ.	Max.	Min.	Typ.	Max.
eee	-	-	0.150	-	-	0.0059
fff	-	-	0.050	-	-	0.0020

1. Values in inches are converted from mm and rounded to 4 decimal digits.

Figure 99. UFBGA176+25-ball, 10 x 10 mm, 0.65 mm pitch, ultra fine pitch ball grid array package recommended footprint

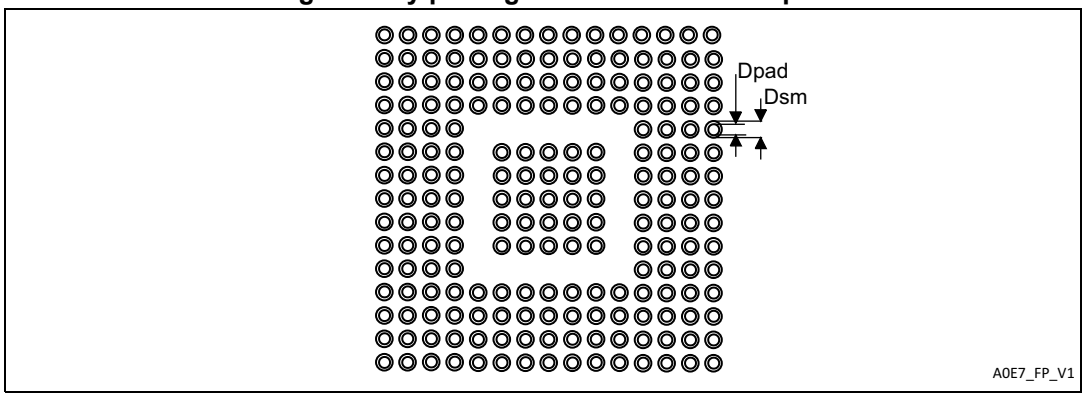


Table 119. UFBGA176+25 recommended PCB design rules (0.65 mm pitch BGA)

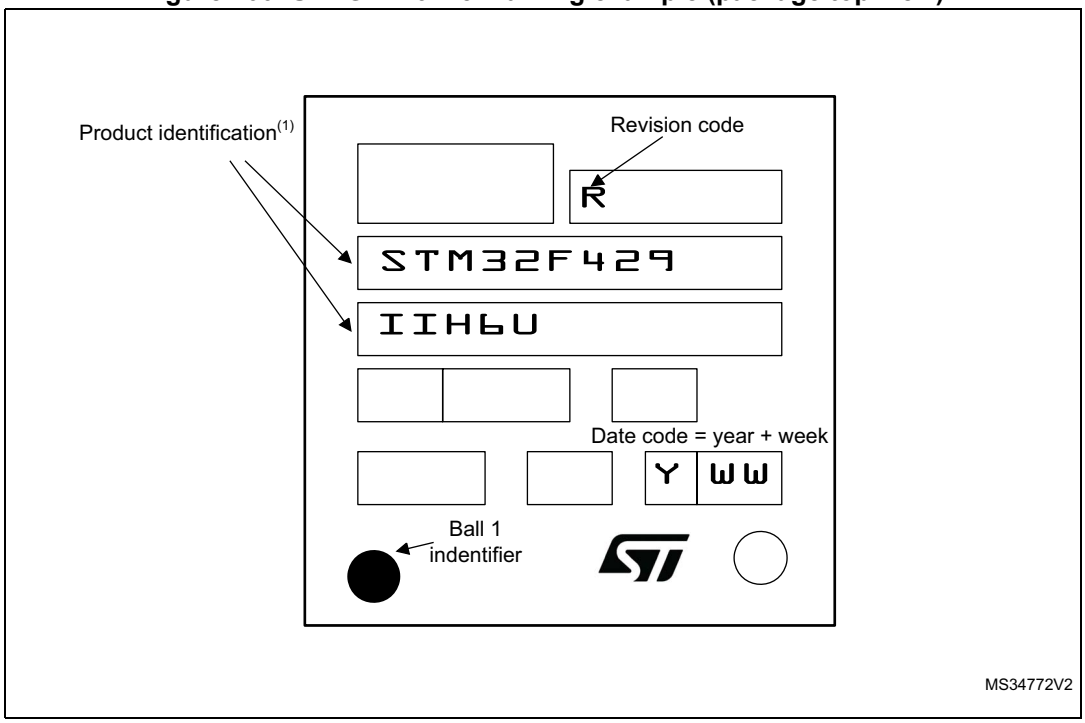
Dimension	Recommended values
Pitch	0.65 mm
Dpad	0.300 mm
Dsm	0.400 mm typ. (depends on the soldermask registration tolerance)
Stencil opening	0.300 mm
Stencil thickness	Between 0.100 mm and 0.125 mm
Pad trace width	0.100 mm

Device marking for UFBGA176+25

The following figure gives an example of topside marking orientation versus ball A1 identifier location.

Other optional marking or inset/upset marks, which depend on assembly location, are not indicated below.

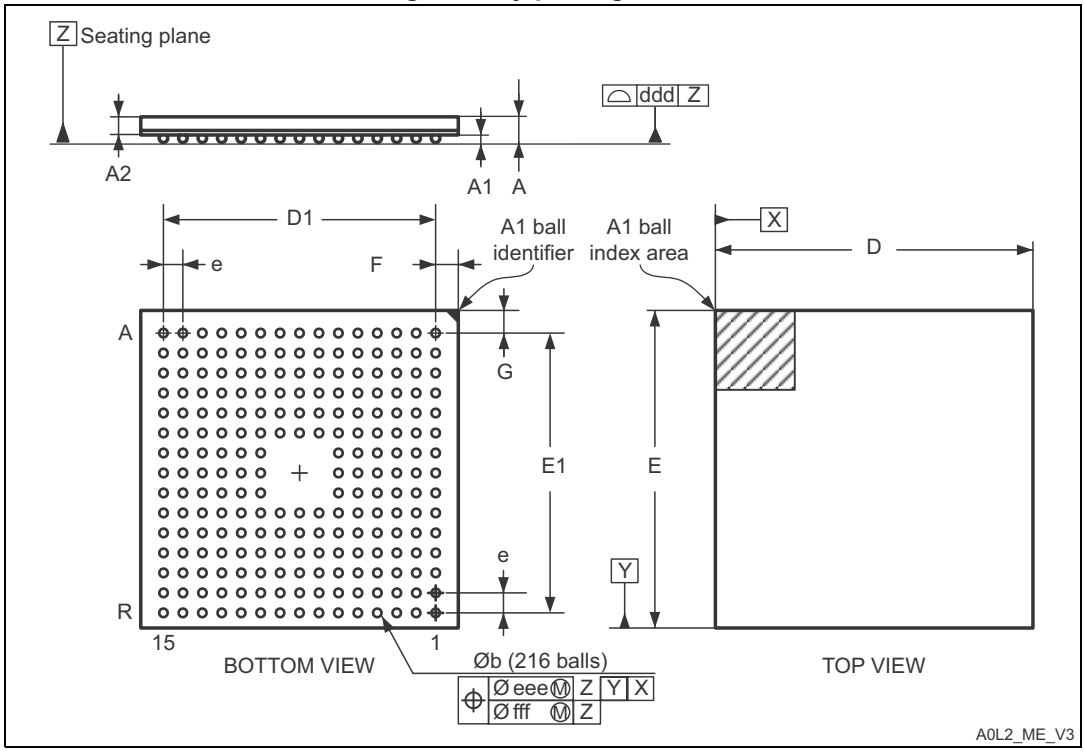
Figure 100. UFBGA176+25 marking example (package top view)



1. Parts marked as “ES”, “E” or accompanied by an Engineering Sample notification letter, are not yet qualified and therefore not yet ready to be used in production and any consequences deriving from such usage will not be at ST charge. In no event, ST will be liable for any customer usage of these engineering samples in production. ST Quality has to be contacted prior to any decision to use these Engineering Samples to run qualification activity.

7.8 TFBGA216 package information

Figure 101. TFBGA216 - 216 ball 13 × 13 mm 0.8 mm pitch thin fine pitch ball grid array package outline



1. Drawing is not to scale.

Table 120. TFBGA216 - 216 ball 13 × 13 mm 0.8 mm pitch thin fine pitch ball grid array package mechanical data

Symbol	millimeters			inches[1]		
	Min	Typ	Max	Min	Typ	Max
A	-	-	1.100	-	-	0.0433
A1	0.150	-	-	0.0059	-	-
A2	-	0.760	-	-	0.0299	-
b	0.350	0.400	0.450	0.0138	0.0157	0.0177
D	12.850	13.000	13.150	0.5118	0.5118	0.5177
D1	-	11.200	-	-	0.4409	-
E	12.850	13.000	13.150	0.5118	0.5118	0.5177
E1	-	11.200	-	-	0.4409	-
e	-	0.800	-	-	0.0315	-
F	-	0.900	-	-	0.0354	-
ddd	-	-	0.100	-	-	0.0039

Table 120. TFBGA216 - 216 ball 13 × 13 mm 0.8 mm pitch thin fine pitch ball grid array package mechanical data (continued)

Symbol	millimeters			inches[1]		
	Min	Typ	Max	Min	Typ	Max
eee	-	-	0.150	-	-	0.0059
fff	-	-	0.080	-	-	0.0031

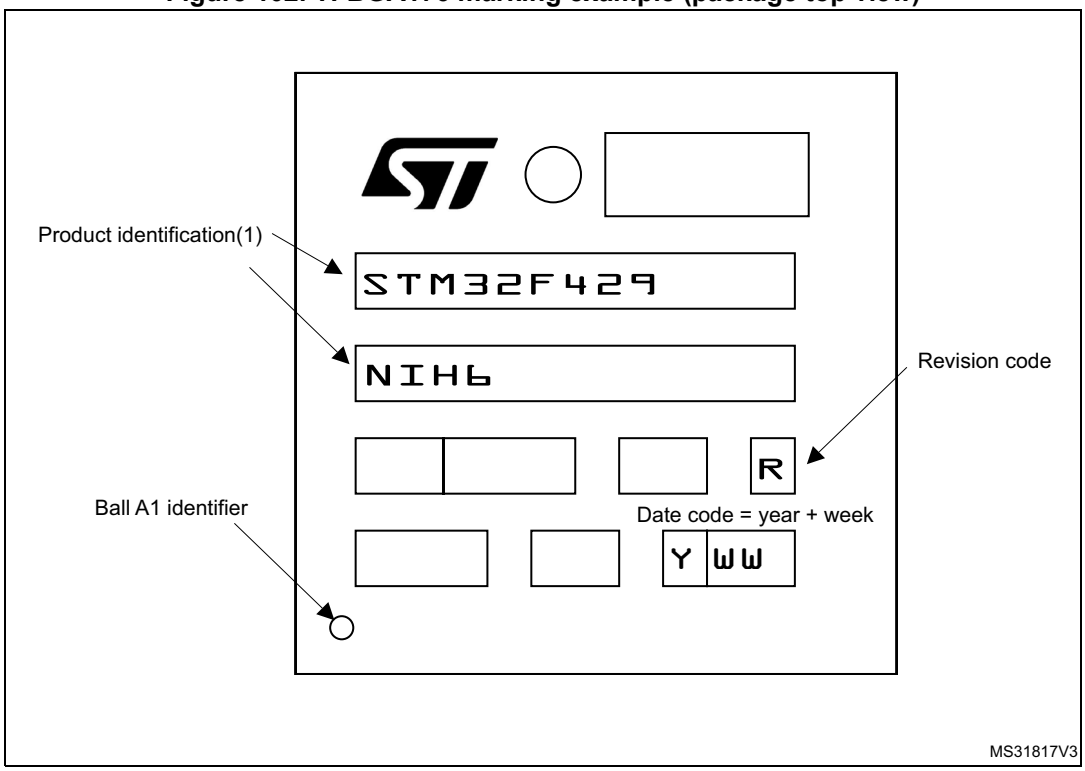
1. Values in inches are converted from mm and rounded to 4 decimal digits.

Device marking for TFBGA176

The following figure gives an example of topside marking orientation versus ball A1 identifier location.

Other optional marking or inset/upset marks, which depend on assembly location, are not indicated below.

Figure 102. TFBGA176 marking example (package top view)



1. Parts marked as “ES”, “E” or accompanied by an Engineering Sample notification letter, are not yet qualified and therefore not yet ready to be used in production and any consequences deriving from such usage will not be at ST charge. In no event, ST will be liable for any customer usage of these engineering samples in production. ST Quality has to be contacted prior to any decision to use these Engineering Samples to run qualification activity.

7.9 Thermal characteristics

The maximum chip-junction temperature, T_J max, in degrees Celsius, may be calculated using the following equation:

T_J max = T_A max + (P_D max x Θ_{JA})

Where:

- T_A max is the maximum ambient temperature in °C,
- Θ_{JA} is the package junction-to-ambient thermal resistance, in °C/W,
- P_D max is the sum of P_{INT} max and $P_{I/O}$ max (P_D max = P_{INT} max + $P_{I/O}$max),
- P_{INT} max is the product of I_{DD} and V_{DD}, expressed in Watts. This is the maximum chip internal power.

$P_{I/O}$ max represents the maximum power dissipation on output pins where:

$P_{I/O}$ max = Σ (V_{OL} × I_{OL}) + Σ((V_{DD} – V_{OH}) × I_{OH}),

taking into account the actual V_{OL} / I_{OL} and V_{OH} / I_{OH} of the I/Os at low and high level in the application.

Table 121. Package thermal characteristics

Symbol	Parameter	Value	Unit
Θ_{JA}	**Thermal resistance junction-ambient** LQFP100 - 14 × 14 mm / 0.5 mm pitch	43	°C/W
	Thermal resistance junction-ambient WLCSP143	31.2	
	Thermal resistance junction-ambient LQFP144 - 20 × 20 mm / 0.5 mm pitch	40	
	Thermal resistance junction-ambient LQFP176 - 24 × 24 mm / 0.5 mm pitch	38	
	Thermal resistance junction-ambient LQFP208 - 28 × 28 mm / 0.5 mm pitch	19	
	Thermal resistance junction-ambient UFBGA169 - 7 × 7mm / 0.5 mm pitch	52	
	Thermal resistance junction-ambient UFBGA176 - 10× 10 mm / 0.5 mm pitch	39	
	Thermal resistance junction-ambient TFBGA216 - 13 × 13 mm / 0.8 mm pitch	29	

Reference document

JESD51-2 Integrated Circuits Thermal Test Method Environment Conditions - Natural Convection (Still Air). Available from www.jedec.org.

8 Part numbering

Table 122. Ordering information scheme

Example: STM32 F 429 V I T 6 xxx

Device family

STM32 = Arm-based 32-bit microcontroller

Product type

F = general-purpose

Device subfamily

427= STM32F427xx, USB OTG FS/HS, camera interface, Ethernet

429= STM32F429xx, USB OTG FS/HS, camera interface, Ethernet, LCD-TFT

Pin count

V = 100 pins

Z = 143 and 144 pins

A = 169 pins

I = 176 pins

B = 208 pins

N = 216 pins

Flash memory size

E = 512 Kbytes of Flash memory

G = 1024 Kbytes of Flash memory

I = 2048 Kbytes of Flash memory

Package

T = LQFP

H = BGA

Y = WLCSP

Temperature range

6 = Industrial temperature range, –40 to 85 °C.

7 = Industrial temperature range, –40 to 105 °C.

Options

xxx = programmed parts

TR = tape and reel

For a list of available options (speed, package, etc.) or for further information on any aspect of this device, please contact your nearest ST sales office.

Appendix A Recommendations when using internal reset OFF

When the internal reset is OFF, the following integrated features are no longer supported:

- The integrated power-on reset (POR) / power-down reset (PDR) circuitry is disabled.
- The brownout reset (BOR) circuitry must be disabled.
- The embedded programmable voltage detector (PVD) is disabled.
- V_{BAT} functionality is no more available and VBAT pin should be connected to V_{DD}.
- The over-drive mode is not supported.

A.1 Operating conditions

Table 123. Limitations depending on the operating power supply range

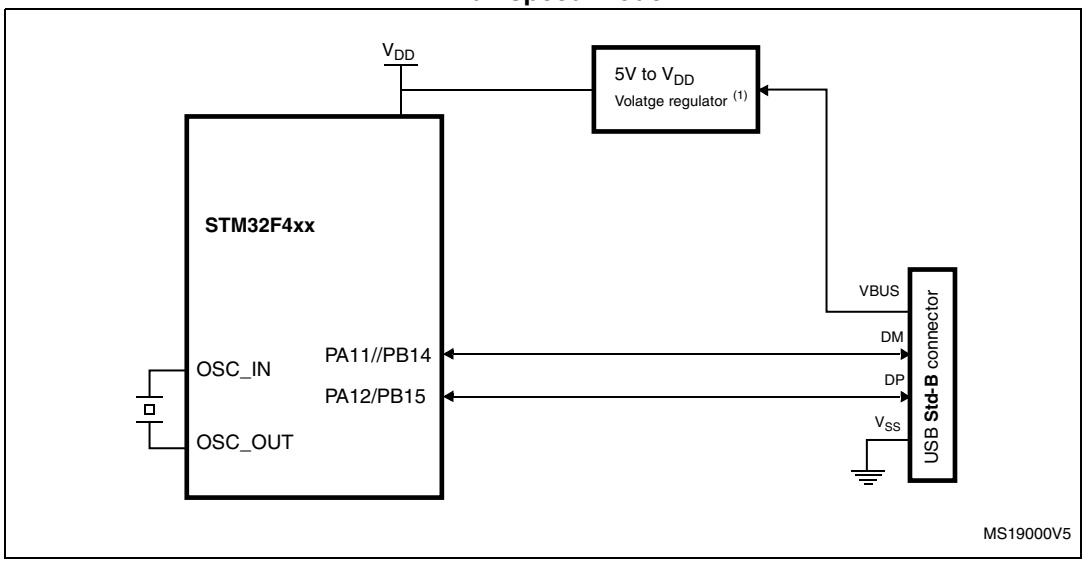
Operating power supply range	ADC operation	Maximum Flash memory access frequency with no wait states ($f_{Flashmax}$)	Maximum Flash memory access frequency with wait states (1)(2)	I/O operation	Possible Flash memory operations
V_{DD} =1.7 to 2.1 V(3)	Conversion time up to 1.2 Msps	20 MHz(4)	168 MHz with 8 wait states and over-drive OFF	– No I/O compensation	8-bit erase and program operations only

1. Applicable only when the code is executed from Flash memory. When the code is executed from RAM, no wait state is required.
2. Thanks to the ART accelerator and the 128-bit Flash memory, the number of wait states given here does not impact the execution speed from Flash memory since the ART accelerator allows to achieve a performance equivalent to 0 wait state program execution.
3. V_{DD}/V_{DDA} minimum value of 1.7 V, with the use of an external power supply supervisor (refer to *Section 3.17.1: Internal reset ON*).
4. Prefetch is not available. Refer to AN3430 application note for details on how to adjust performance and power.

Appendix B Application block diagrams

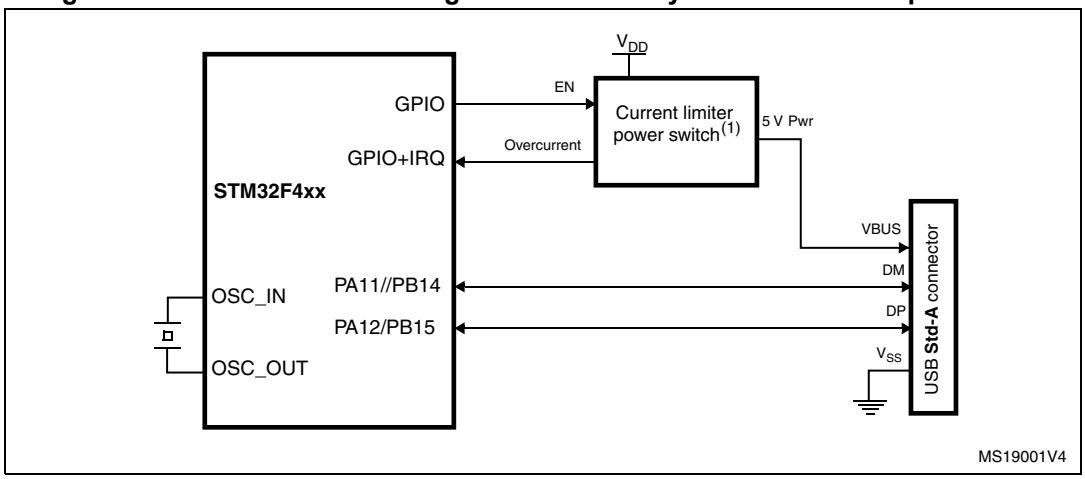
B.1 USB OTG full speed (FS) interface solutions

Figure 103. USB controller configured as peripheral-only and used in Full speed mode



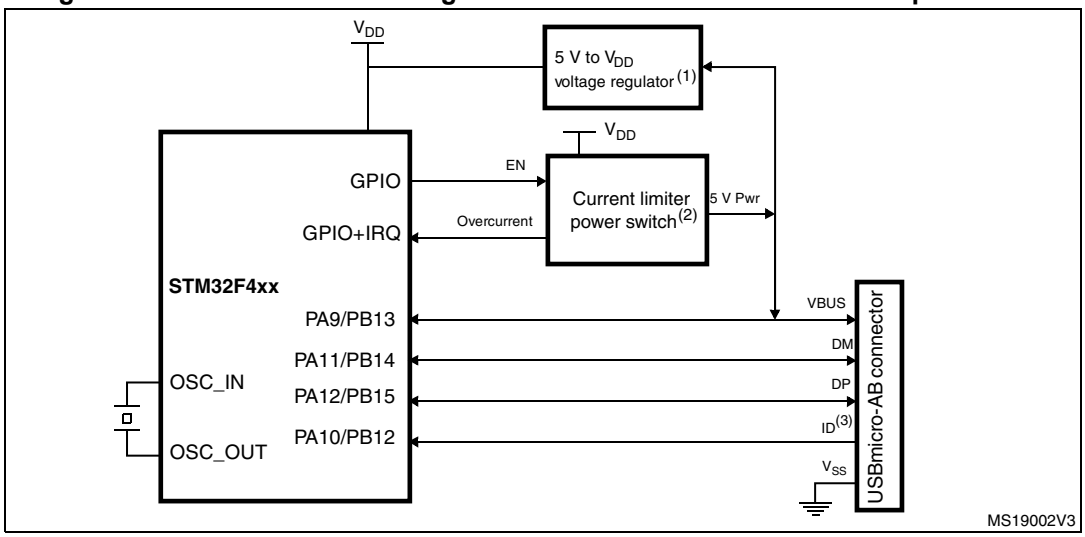
1. External voltage regulator only needed when building a V_{BUS} powered device.
2. The same application can be developed using the OTG HS in FS mode to achieve enhanced performance thanks to the large Rx/Tx FIFO and to a dedicated DMA controller.

Figure 104. USB controller configured as host-only and used in full speed mode



1. The current limiter is required only if the application has to support a V_{BUS} powered device. A basic power switch can be used if 5 V are available on the application board.
2. The same application can be developed using the OTG HS in FS mode to achieve enhanced performance thanks to the large Rx/Tx FIFO and to a dedicated DMA controller.

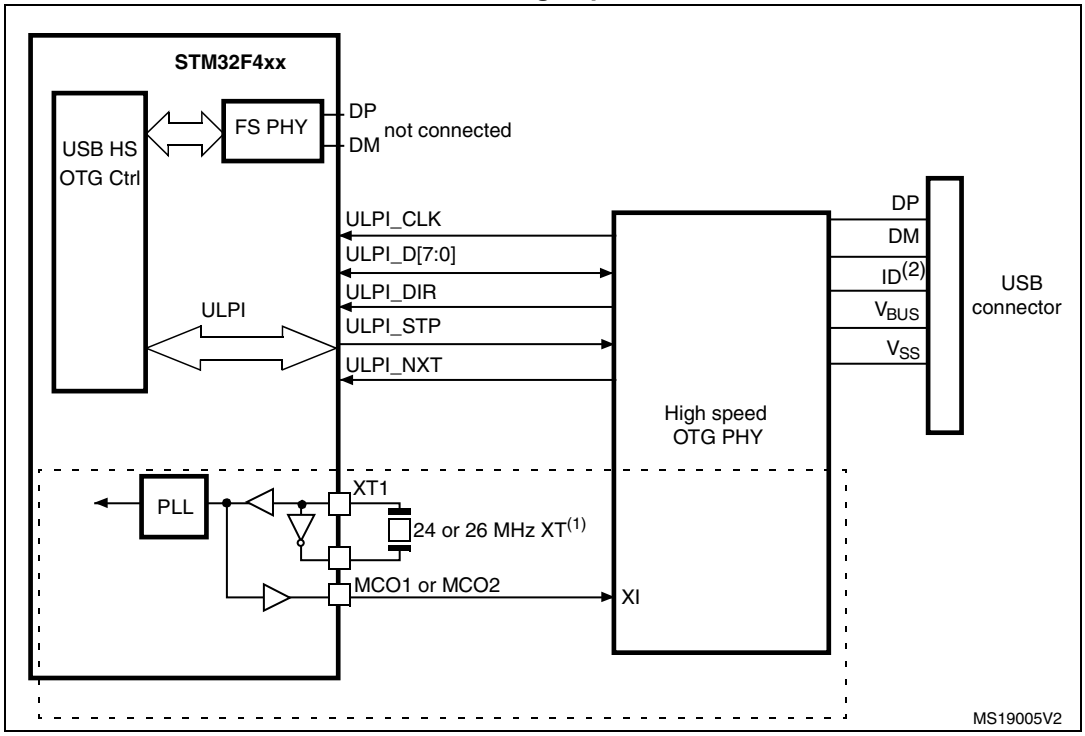
Figure 105. USB controller configured in dual mode and used in full speed mode



1. External voltage regulator only needed when building a V_{BUS} powered device.
2. The current limiter is required only if the application has to support a V_{BUS} powered device. A basic power switch can be used if 5 V are available on the application board.
3. The ID pin is required in dual role only.
4. The same application can be developed using the OTG HS in FS mode to achieve enhanced performance thanks to the large Rx/Tx FIFO and to a dedicated DMA controller.

B.2 USB OTG high speed (HS) interface solutions

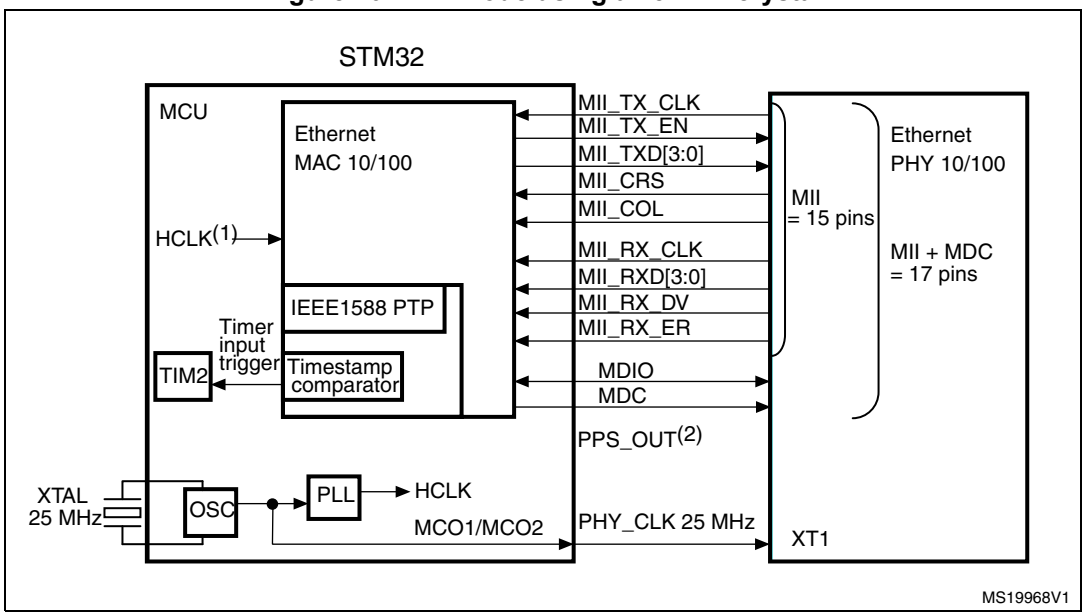
Figure 106. USB controller configured as peripheral, host, or dual-mode and used in high speed mode



1. It is possible to use MCO1 or MCO2 to save a crystal. It is however not mandatory to clock the STM32F42x with a 24 or 26 MHz crystal when using USB HS. The above figure only shows an example of a possible connection.
2. The ID pin is required in dual role only.

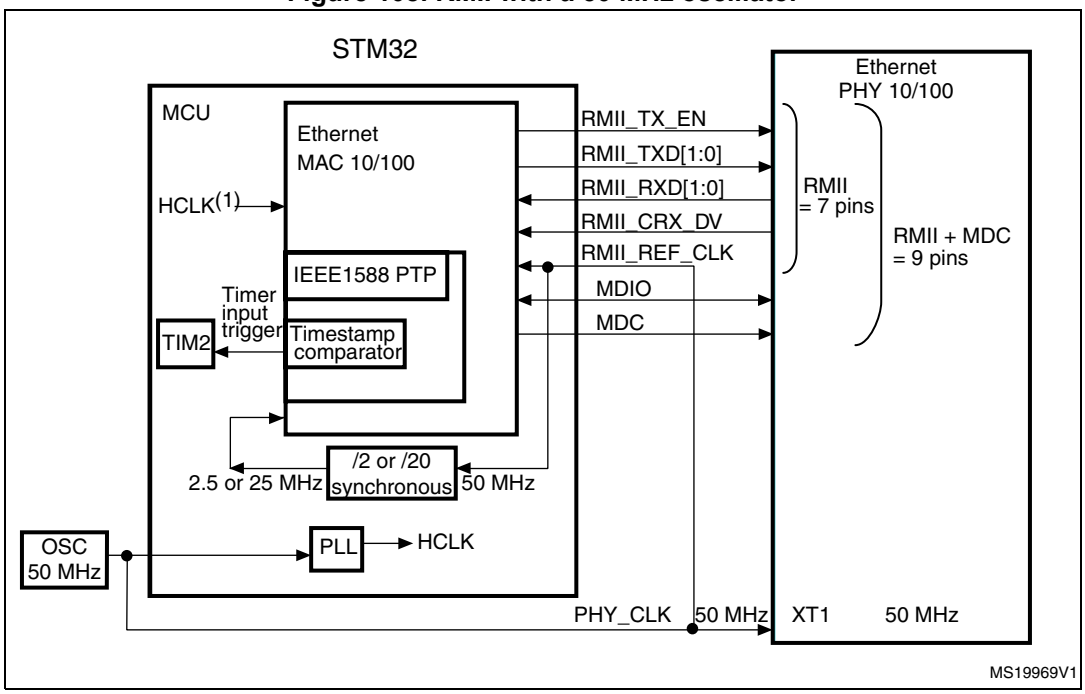
B.3 Ethernet interface solutions

Figure 107. MII mode using a 25 MHz crystal



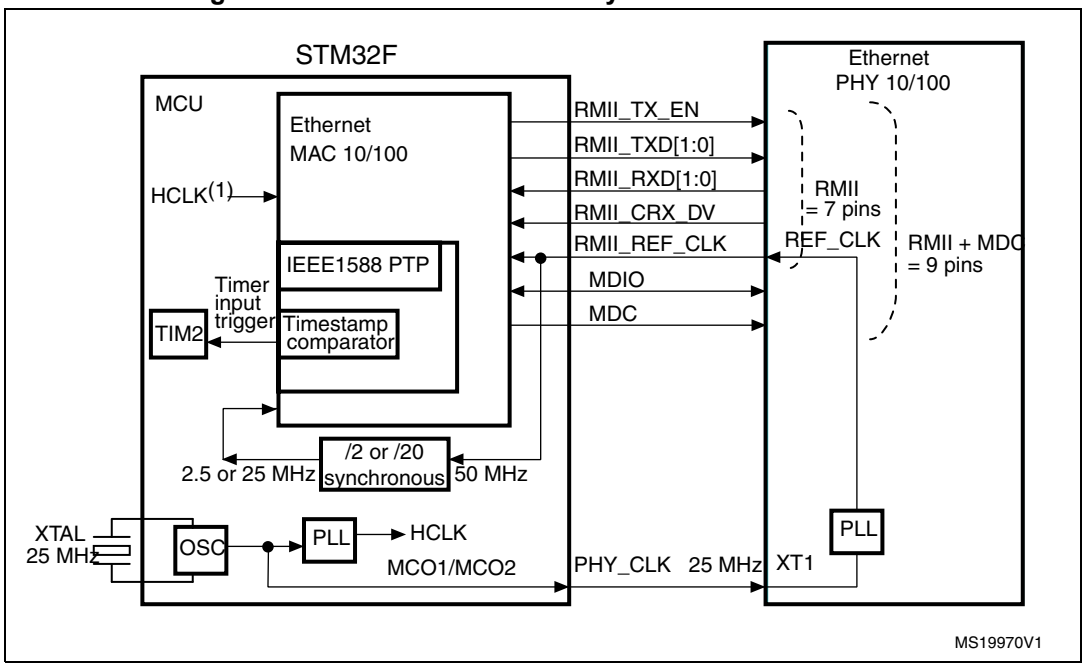
1. f_{HCLK} must be greater than 25 MHz.
2. Pulse per second when using IEEE1588 PTP optional signal.

Figure 108. RMII with a 50 MHz oscillator



1. f_{HCLK} must be greater than 25 MHz.

Figure 109. RMII with a 25 MHz crystal and PHY with PLL



1. f_{HCLK} must be greater than 25 MHz.
2. The 25 MHz (PHY_CLK) must be derived directly from the HSE oscillator, before the PLL block.

9 Revision history

Table 124. Document revision history

Date	Revision	Changes
19-Mar-2013	1	Initial release.
10-Sep-2013	2	Added STM32F429xx part numbers and related informations. **STM32F427xx part numbers:** Replaced FSMC by FMC added Chrom-ART Accelerator and SAI interface. Increased core, timer, GPIOs, SPI maximum frequencies Updated *Figure 8*.Updated *Figure 9*. Removed note in *Section ·: Standby mode*. Updated *Figure 18*. Updated *Table 10: STM32F427xx and STM32F429xx pin and ball definitions* and *Table 12: STM32F427xx and STM32F429xx alternate function mapping*.. Modified *Figure 19: Memory map*. Updated *Table 17: General operating conditions*, *Table 18: Limitations depending on the operating power supply range*. Removed note 1 in *Table 22: reset and power control block characteristics*. Added *Table 23: Over-drive switching characteristics*. Updated *Section : Typical and maximum current consumption*, *Table 34: Switching output I/O current consumption*, *Table 35: Peripheral current consumption* and *Section : On-chip peripheral current consumption*. Updated *Table 36: Low-power mode wakeup timings*. Modified *Section : High-speed external user clock generated from an external source*, *Section : Low-speed external user clock generated from an external source*, and *Section 6.3.10: Internal clock source characteristics*. Updated *Table 43: Main PLL characteristics* and *Table 45: PLLISAI (audio and LCD-TFT PLL) characteristics*. Updated *Table 52: EMI characteristics*. Updated *Table 57: Output voltage characteristics* and *Table 58: I/O AC characteristics*. Updated *Table 60: TIMx characteristics*, *Table 61: I^2C characteristics*, *Table 62: SPI dynamic characteristics*, *Section : SAI characteristics*. Updated *Table 102: SDRAM read timings* and *Table 104: SDRAM write timings*.

Table 124. Document revision history

Date	Revision	Changes
24-Jan-2014	3	Added STM32F429xE part numbers featuring 512 Mbytes of Flash memory and UFBGA169 package. Added LPSDR SDRAM. Changed INTN into INTR in *Figure 4: STM32F427xx and STM32F429xx block diagram*. Added note 4 in *Table 2: STM32F427xx and STM32F429xx features and peripheral counts*. Updated *Section 3.15: Boot modes*. Updated for PA4 and PA5 in *Table 10: STM32F427xx and STM32F429xx pin and ball definitions*. Added V_{IN} for BOOT0 pins in *Table 14: Voltage characteristics*. Updated Note *6.*, added Note *1.*,and updated maximum V_{IN} for B pins in *Table 17: General operating conditions*. Updated maximum Flash memory access frequency with wait states for V_{DD} =1.8 to 2.1 V in *Table 18: Limitations depending on the operating power supply range*. Updated *Table 24: Typical and maximum current consumption in Run mode, code with data processing running from Flash memory (ART accelerator enabled except prefetch) or RAM* and *Table 25: Typical and maximum current consumption in Run mode, code with data processing running from Flash memory (ART accelerator disabled)*. Updated *Table 30: Typical current consumption in Run mode, code with data processing running from Flash memory or RAM, regulator ON (ART accelerator enabled except prefetch), VDD=1.7 V*, *Table 31: Typical current consumption in Run mode, code with data processing running from Flash memory, regulator OFF (ART accelerator enabled except prefetch)*, and *Table 32: Typical current consumption in Sleep mode, regulator ON, VDD=1.7 V*. Updated *Table 57: Output voltage characteristics*. Updated *Table 58: I/O AC characteristics*. Added *Figure 35*. Updated $t_{h(SDA)}$, $t_{r(SDA)}$ and $t_{r(SCL)}$ and added t_{SP} in *Table 61: I^2C characteristics*. Updated f_{SCK} in *Table 62: SPI dynamic characteristics*. Updated *Table 70: Dynamic characteristics: USB ULPI*. Updated *Section 6.3.26: FMC characteristics* conditions. Updated *Figure 73: SDRAM read access waveforms (CL = 1)* and *Figure 74: SDRAM write access waveforms*. Added *Table 103: LPSDR SDRAM read timings* and *Table 105: LPSDR SDRAM write timings*. Updated *Table 102: SDRAM read timings* and *Table 104: SDRAM write timings* and added note *2.Table 108: Dynamic characteristics: SD / MMC characteristics*.

Table 124. Document revision history

Date	Revision	Changes
24-Apr-2014	4	In the whole document, minimum supply voltage changed to 1.7 V when external power supply supervisor is used. Added DCMI_VSYNC alternate function on PG9 and updated note *6.* in *Table 10: STM32F427xx and STM32F429xx pin and ball definitions* and *Table 12: STM32F427xx and STM32F429xx alternate function mapping*. Added note *2.*below*Figure 16: STM32F42x UFBGA169 ballout*. Changed SVGA (800x600) into XGA1024x768) on cover page and in *Section 3.10: LCD-TFT controller (available only on STM32F429xx)*. Updated *Section 3.18.2: Regulator OFF*. Updated signal corresponding to pin L5 in *Figure 12: STM32F42x WLCSP143 ballout*. Added ACC_{HSE} in *Table 39: HSE 4-26 MHz oscillator characteristics* and ACC_{LSE} in *Table 40: LSE oscillator characteristics (fLSE = 32.768 kHz)*. Updated *Table 53: ESD absolute maximum ratings*. Updated V_{IH} in *Table 56: I/O static characteristics*. Added condition V_{DD}>1.7 V in *Table 58: I/O AC characteristics*. Updated conditions in *Table 62: SPI dynamic characteristics*. Added Z_{DRV} in *Table 67: USB OTG full speed electrical characteristics* Removed note 3 in *Table 80: Temperature sensor characteristics*. Added *Figure 82: LQFP100 marking example (package top view)*, *Figure 85: WLCSP143 marking example (package top view)*, *Figure 88: LQFP144 marking example (package top view)*, *Figure 91: LQFP176 marking (package top view)*, *Figure 94: LQFP208 marking example (package top view)*, *Figure 97: UFBGA169 marking example (package top view)* and *Figure 100: UFBGA176+25 marking example (package top view)*. Added *Appendix A: Recommendations when using internal reset OFF*. Removed Internal reset OFF hardware connection appendix.

Table 124. Document revision history

Date	Revision	Changes
19-Feb-2015	5	Update SPI/IS2 in *Table 2: STM32F427xx and STM32F429xx features and peripheral counts*. Updated LQFP208 in *Table 4: Regulator ON/OFF and internal reset ON/OFF availability*. Updated *Figure 19: Memory map*. Changed PLS[2:0]=101 (falling edge) maximum value in *Table 22: reset and power control block characteristics*. Updated current consumption with all peripherals disabled in *Table 24: Typical and maximum current consumption in Run mode, code with data processing running from Flash memory (ART accelerator enabled except prefetch) or RAM*. Updated note *1.* in *Table 28: Typical and maximum current consumptions in Standby mode*. Updated t_{WUSTOP} in *Table 36: Low-power mode wakeup timings*. Updated ESD standards and *Table 53: ESD absolute maximum ratings*. Updated *Table 56: I/O static characteristics*. *Section : I2C interface characteristics*: updated section introduction, removed Table *I2C characteristics*, Figure *I2C bus AC waveforms and measurement circuit* and Table *SCL frequency*; added *Table 61: I2C analog filter characteristics*. Updated measurement conditions in *Table 62: SPI dynamic characteristics*. Updated *Figure 51: Typical connection diagram using the ADC*. Updated *Section : Device marking for LQFP100*. Updated *Figure 83: WLCSP143 - 143-ball, 4.521x 5.547 mm, 0.4 mm pitch wafer level chip scale package outline* and *Table 111: WLCSP143 - 143-ball, 4.521x 5.547 mm, 0.4 mm pitch wafer level chip scale package mechanical data*; added *Figure 84: WLCSP143 - 143-ball, 4.521x 5.547 mm, 0.4 mm pitch wafer level chip scale recommended footprint* and *Table 112: WLCSP143 recommended PCB design rules (0.4 mm pitch)*. Updated *Figure 85: WLCSP143 marking example (package top view)* and related note. Updated *Section : Device marking for WLCSP143*. Updated *Section : Device marking for LQFP144*. Updated *Section : Device marking for LQFP176*. Updated *Figure 92: LQFP208 - 208-pin, 28 x 28 mm low-profile quad flat package outline*; Updated *Section : Device marking for LQFP208*. Modified UFBGA169 pitch, updated *Figure 95: UFBGA169 - 169-ball 7 x 7 mm 0.50 mm pitch, ultra fine pitch ball grid array package outline* and *Table 116: UFBGA169 - 169-ball 7 x 7 mm 0.50 mm pitch, ultra fine pitch ball grid array package mechanical data*; updated *Section : Device marking for LQFP208*. updated *Section : Device marking for UFBGA169*, *Section : Device marking for UFBGA176+25* and *Section : Device marking for TFBGA176*. Updated Z pin count in *Table 122: Ordering information scheme*.

Table 124. Document revision history

Date	Revision	Changes
17-Sep-2015	6	Updated notes related to the minimum and maximum values guaranteed by design, characterization or test in production. Updated $I_{DD_STOP_UDM}$ in *Table 27: Typical and maximum current consumptions in Stop mode*. Removed note related to tests in production in *Table 24: Typical and maximum current consumption in Run mode, code with data processing running from Flash memory (ART accelerator enabled except prefetch) or RAM* and *Table 26: Typical and maximum current consumption in Sleep mode*. Updated *Table 41: HSI oscillator characteristics*. *Figure 31* renamed *ACCHSI accuracy versus temperature* and updated. Updated *Figure 38: SPI timing diagram - slave mode and CPHA = 0*. Updated *Section : Ethernet characteristics*. Updated *Table 43: Main PLL characteristics*, *Table 44: PLLI2S (audio PLL) characteristics* and *Table 45: PLLISAI (audio and LCD-TFT PLL) characteristics*. Removed note 1 in *Table 75: ADC static accuracy at fADC = 18 MHz*, *Table 76: ADC static accuracy at fADC = 30 MHz* and *Table 77: ADC static accuracy at fADC = 36 MHz*. Updated $t_{d(SDCLKL_Data)}$ and $t_{h(SDCLKL_Data)}$ in *Table 104: SDRAM write timings*. Added *Figure 96: UFBGA169 - 169-ball, 7 x 7 mm, 0.50 mm pitch, ultra fine pitch ball grid array recommended footprint* and *Table 117: UFBGA169 recommended PCB design rules (0.5 mm pitch BGA)*. Added *Figure 99: UFBGA176+25-ball, 10 x 10 mm, 0.65 mm pitch, ultra fine pitch ball grid array package recommended footprint* and *Table 119: UFBGA176+25 recommended PCB design rules (0.65 mm pitch BGA)*.
30-Nov-2015	7	Updated $\lvert V_{SSX} - V_{SS} \rvert$ in *Table 14: Voltage characteristics* to add V_{REF-}. Updated $t_{d(TXEN)}$ and $t_{d(TXD)}$ minimum value in *Table 72: Dynamics characteristics: Ethernet MAC signals for RMII* and *Table 73: Dynamics characteristics: Ethernet MAC signals for MII*. Added V_{REF-} in *Table 74: ADC characteristics*. Added A1 minimum and maximum values in *Table 111: WLCSP143 - 143-ball, 4.521x 5.547 mm, 0.4 mm pitch wafer level chip scale package mechanical data*. Updated *Figure 86: LQFP144-144-pin, 20 x 20 mm low-profile quad flat package outline*. Updated *Figure 98: UFBGA176+25 - ball 10 x 10 mm, 0.65 mm pitch ultra thin fine pitch ball grid array package outline* and *Table 118: UFBGA176+25 - ball, 10 x 10 mm, 0.65 mm pitch, ultra fine pitch ball grid array package mechanical data*. Updated *Figure 101: TFBGA216 - 216 ball 13 × 13 mm 0.8 mm pitch thin fine pitch ball grid array package outline* and *Table 120: TFBGA216 - 216 ball 13 × 13 mm 0.8 mm pitch thin fine pitch ball grid array package mechanical data*.
21-Jan-2016	8	Updated *Figure 22: Power supply scheme*. Added $t_{d(TXD)}$ values corresponding to 1.71 V < V_{DD} < 3.6 V in *Table 72: Dynamics characteristics: Ethernet MAC signals for RMII*.

Table 124. Document revision history

Date	Revision	Changes
18-Jul-2016	9	Updated *Figure 1: Compatible board design STM32F10xx/STM32F2xx/STM32F4xx for LQFP100 package*. Added mission profile compliance with JEDEC JESD47 in *Section 6.2: Absolute maximum ratings*. Changed *Figure 31 HSI deviation versus temperature to ACCHSI versus temperature.* Updated R_{LOAD} in *Table 85: DAC characteristics*. Added note *2.* related to the position of the 0.1 µF capacitor below *Figure 37: Recommended NRST pin protection*. Updated *Figure 40: SPI timing diagram - master mode*. Added reference to optional marking or inset/upset marks in all package device marking sections. Updated *Figure 85: WLCSP143 marking example (package top view)*, *Figure 88: LQFP144 marking example (package top view)*, *Figure 91: LQFP176 marking (package top view)*, *Figure 94: LQFP208 marking example (package top view)*. Updated *Figure 98: UFBGA176+25 - ball 10 x 10 mm, 0.65 mm pitch ultra thin fine pitch ball grid array package outline* and *Table 118: UFBGA176+25 - ball, 10 x 10 mm, 0.65 mm pitch, ultra fine pitch ball grid array package mechanical data*.
19-Jan-2018	10	Updated Arm wordmark and added Arm logo in *Section 2: Description*. Updated LDC-TFT feature on cover page. Updated *Table 24: Typical and maximum current consumption in Run mode, code with data processing running from Flash memory (ART accelerator enabled except prefetch) or RAM* and *Table 26: Typical and maximum current consumption in Sleep mode*. R_{ADC} minimum value added in *Table 74: ADC characteristics*. LTDC clock output frequency changed to 83 MHz in *Table 107: LTDC characteristics*.

IMPORTANT NOTICE – PLEASE READ CAREFULLY

STMicroelectronics NV and its subsidiaries (“ST”) reserve the right to make changes, corrections, enhancements, modifications, and improvements to ST products and/or to this document at any time without notice. Purchasers should obtain the latest relevant information on ST products before placing orders. ST products are sold pursuant to ST’s terms and conditions of sale in place at the time of order acknowledgement.

Purchasers are solely responsible for the choice, selection, and use of ST products and ST assumes no liability for application assistance or the design of Purchasers’ products.

No license, express or implied, to any intellectual property right is granted by ST herein.

Resale of ST products with provisions different from the information set forth herein shall void any warranty granted by ST for such product.

ST and the ST logo are trademarks of ST. All other product or service names are the property of their respective owners.

Information in this document supersedes and replaces information previously supplied in any prior versions of this document.

© 2018 STMicroelectronics – All rights reserved

Advanced Monolithic Systems

AMS1117

800mA LOW DROPOUT VOLTAGE REGULATOR

FEATURES

- **Three Terminal Adjustable or Fixed Voltages* 1.5V, 1.8V, 2.5V, 2.85V, 3.3V and 5.0V**
- **Output Current of 800mA**
- **Operates Down to 1V Dropout**
- **Line Regulation: 0.2% Max.**
- **Load Regulation: 0.4% Max.**
- **SOT-223 and TO-252 package available**

APPLICATIONS

- **High Efficiency Linear Regulators**
- **Post Regulators for Switching Supplies**
- **5V to 3.3V Linear Regulator**
- **Battery Chargers**
- **Active SCSI Terminators**
- **Power Management for Notebook**
- **Battery Powered Instrumentation**

GENERAL DESCRIPTION

The AMS1117 series of adjustable and fixed voltage regulators are designed to provide 800mA output current and to operate down to 1V input-to-output differential. The dropout voltage of the device is guaranteed maximum 1.3V at maximum output current, decreasing at lower load currents.

On-chip trimming adjusts the reference voltage to 1%. Current limit is also trimmed, minimizing the stress under overload conditions on both the regulator and power source circuitry.

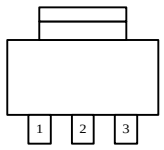
The AMS1117 devices are pin compatible with other three-terminal SCSI regulators and are offered in the low profile surface mount SOT-223 package and in the TO-252 (DPAK) plastic package.

ORDERING INFORMATION:

PACKAGE TYPE		OPERATING JUNCTION
TO-252	SOT-223	TEMPERATURE RANGE
AMS1117CD	AMS1117	0 to 125° C
AMS1117CD-1.5	AMS1117-1.5	0 to 125° C
AMS1117CD-1.8	AMS1117-1.8	0 to 125° C
AMS1117CD-2.5	AMS1117-2.5	0 to 125° C
AMS1117CD-2.85	AMS1117-2.85	0 to 125° C
AMS1117CD-3.3	AMS1117-3.3	0 to 125° C
AMS1117CD-5.0	AMS1117-5.0	0 to 125° C

*For additional available fixed voltages contact factory.

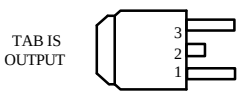
SOT-223 Top View



PIN CONNECTIONS

FIXED VERSION	ADJUSTABLE VERSION
1- Ground	1- Adjust
2- V_{OUT}	2- V_{OUT}
3- V_{IN}	3- V_{IN}

TO-252 FRONT VIEW



Advanced Monolithic Systems, Inc. 6680B Sierra Lane, Dublin, CA 94568 Phone (925) 556-9090 Fax (925) 556-9140

ABSOLUTE MAXIMUM RATINGS (Note 1)

Power Dissipation	Internally limited
Input Voltage	15V
Operating Junction Temperature	
Control Section	0°C to 125°C
Power Transistor	0°C to 150°C
Storage temperature	- 65°C to +150°C

Soldering information

Lead Temperature (10 sec) 300°C

Thermal Resistance

TO-252 package φ_{JA}= 80°C/W

SOT-223 package φ_{JA}= 90°C/W*

* With package soldering to copper area over backside ground plane or internal power plane φ_{JA} can vary from 46°C/W to >90°C/W depending on mounting technique and the size of the copper area.

ELECTRICAL CHARACTERISTICS

Electrical Characteristics at I_{OUT} = 0 mA, and T_J = +25°C unless otherwise specified.

Parameter	Device	Conditions	Min	Typ	Max	Units
Reference Voltage (Note 2)	AMS1117	I_{OUT} = 10 mA 10mA ≤ I_{OUT} ≤ 800mA, 1.5V≤ (V_{IN} - V_{OUT}) ≤ 12V	1.238 **1.225**	1.250 **1.250**	1.262 **1.270**	V V
Output Voltage (Note 2)	AMS1117-1.5	0 ≤ I_{OUT} ≤ 800mA , 3.0V≤ V_{IN} ≤ 12V	1.485 **1.476**	1.500 **1.500**	1.515 **1.524**	V V
	AMS1117-1.8	0 ≤ I_{OUT} ≤ 800mA , 3.3V≤ V_{IN} ≤ 12V	1.782 **1.773**	1.800 **1.800**	1.818 **1.827**	V V
	AMS1117-2.5	0 ≤ I_{OUT} ≤ 800mA , 4.0V≤ V_{IN} ≤ 12V	2.475 **2.460**	2.500 **2.500**	2.525 **2.560**	V V
	AMS1117-2.85	0 ≤ I_{OUT} ≤ 800mA , 4.35V≤ V_{IN} ≤ 12V	2.82 **2.79**	2.850 **2.850**	2.88 **2.91**	V V
	AMS1117-3.3	0 ≤ I_{OUT} ≤ 800mA , 4.75V ≤ V_{IN} ≤ 12V	3.267 **3.235**	3.300 **3.300**	3.333 **3.365**	V V
	AMS1117-5.0	0 ≤ I_{OUT} ≤ 800mA , 6.5V ≤ V_{IN} ≤ 12V	4.950 **4.900**	5.000 **5.000**	5.050 **5.100**	V V
Line Regulation	AMS1117	I_{LOAD} = 10 mA , 1.5V≤ (V_{IN} - V_{OUT}) ≤ 12V		0.015 **0.035**	0.2 **0.2**	% %
	AMS1117-1.5	3.0V≤ V_{IN} ≤ 12V		0.3 **0.6**	5 **6**	mV mV
	AMS1117-1.8	3.3V≤ V_{IN} ≤ 12V		0.3 **0.6**	5 **6**	mV mV
	AMS1117-2.5	4.0V≤ V_{IN} ≤ 12V		0.3 **0.6**	6 **6**	mV mV
	AMS1117-2.85	4.35V≤ V_{IN} ≤ 12V		0.3 **0.6**	6 **6**	mV mV
	AMS1117-3.3	4.75V≤ V_{IN} ≤ 12V		0.5 **1.0**	10 **10**	mV mV
	AMS1117-5.0	6.5V≤ V_{IN} ≤ 12V		0.5 **1.0**	10 **10**	mV mV
Load Regulation (Notes 2, 3)	AMS1117	(V_{IN} - V_{OUT}) =3V, 10mA ≤ I_{OUT} ≤ 800mA		0.1 **0.2**	0.3 **0.4**	% %
	AMS1117-1.5	V_{IN} = 5V, 0 ≤ I_{OUT} ≤ 800mA		3 **6**	10 **20**	mV mV
	AMS1117-1.8	V_{IN} = 5V, 0 ≤ I_{OUT} ≤ 800mA		3 **6**	10 **20**	mV mV
	AMS1117-2.5	V_{IN} = 5V, 0 ≤ I_{OUT} ≤ 800mA		3 **6**	12 **20**	mV mV

ELECTRICAL CHARACTERISTICS

Electrical Characteristics at I_{OUT} = 0 mA, and T_J = +25°C unless otherwise specified.

Parameter	Device	Conditions	Min	Typ	Max	Units
Load Regulation (Notes 2, 3)	AMS1117-2.85	V_{IN} = 5V, 0 ≤ I_{OUT} ≤ 800mA		3 **6**	12 **20**	mV mV
	AMS1117-3.3	V_{IN} = 5V, 0 ≤ I_{OUT} ≤ 800mA		3 **7**	15 **25**	mV mV
	AMS1117-5.0	V_{IN} = 8V, 0 ≤ I_{OUT} ≤ 800mA		5 **10**	20 **35**	mV mV
Dropout Voltage (V_{IN} - V_{OUT})	AMS1117-1.5/-1.8/-2.5/-2.85/-3.3/-5.0	ΔV_{OUT}, ΔV_{REF} = 1%, I_{OUT} = 800mA (Note 4)		**1.1**	**1.3**	V
Current Limit	AMS1117-1.5/-1.8/-2.5/-2.85/-3.3/-5.0	(V_{IN} - V_{OUT}) = 5V	**900**	**1,100**	**1,500**	mA
Minimum Load Current	AMS1117	(V_{IN} - V_{OUT}) = 12V (Note 5)		**5**	**10**	mA
Quiescent Current	AMS1117-1.5/-1.8/-2.5/-2.85/-3.3/-5.0	V_{IN} ≤ 12V		**5**	**10**	mA
Ripple Rejection	AMS1117	f =120Hz , C_{OUT} = 22µF Tantalum, I_{OUT} = 800mA, (V_{IN}-V_{OUT}) = 3V, C_{ADJ} =10µF	**60**	**75**		dB
	AMS1117-1.5/-1.8/-2.5/-2.85	f =120Hz , C_{OUT} = 22µF Tantalum, I_{OUT} = 800mA, V_{IN} = 6V	**60**	**72**		dB
	AMS1117-3.3	f =120Hz , C_{OUT} = 22µF Tantalum, I_{OUT} = 800mA V_{IN} = 6.3V	**60**	**72**		dB
	AMS1117-5.0	f =120Hz , C_{OUT} = 22µF Tantalum, I_{OUT} = 800mA V_{IN} = 8V	**60**	**68**		dB
Thermal Regulation	AMS1117	T_A = 25°C, 30ms pulse		0.008	0.04	%W
Adjust Pin Current	AMS1117	10mA ≤ I_{OUT} ≤ 800mA , 1.5V≤ (V_{IN} - V_{OUT}) ≤ 12V		55	 **120**	µA µA
Adjust Pin Current Change	AMS1117	10mA ≤ I_{OUT} ≤ 800mA , 1.5V≤ (V_{IN} - V_{OUT}) ≤ 12V		**0.2**	**5**	µA
Temperature Stability				**0.5**		%
Long Term Stability		T_A =125°C, 1000Hrs		0.3	1	%
RMS Output Noise (% of V_{OUT})		T_A = 25°C , 10Hz ≤ f ≤ 10kHz		0.003		%
Thermal Resistance Junction-to-Case					15	°C/W

Parameters identified with **boldface type** apply over the full operating temperature range.

Note 1: Absolute Maximum Ratings indicate limits beyond which damage to the device may occur. For guaranteed specifications and test conditions, see the Electrical Characteristics. The guaranteed specifications apply only for the test conditions listed.

Note 2: Line and Load regulation are guaranteed up to the maximum power dissipation of 1.2 W. Power dissipation is determined by the input/output differential and the output current. Guaranteed maximum power dissipation will not be available over the full input/output range.

Note 3: See thermal regulation specifications for changes in output voltage due to heating effects. Line and load regulation are measured at a constant junction temperature by low duty cycle pulse testing. Load regulation is measured at the output lead ~1/8" from the package.

Note 4: Dropout voltage is specified over the full output current range of the device.

Note 5: Minimum load current is defined as the minimum output current required to maintain regulation. When 1.5V ≤ (V_{IN} - V_{OUT}) ≤ 12V the device is guaranteed to regulate if the output current is greater than 10mA.

APPLICATION HINTS

The AMS1117 series of adjustable and fixed regulators are easy to use and are protected against short circuit and thermal overloads. Thermal protection circuitry will shut-down the regulator should the junction temperature exceed 165°C at the sense point.
Pin compatible with older three terminal adjustable regulators, these devices offer the advantage of a lower dropout voltage, more precise reference tolerance and improved reference stability with temperature.

Stability

The circuit design used in the AMS1117 series requires the use of an output capacitor as part of the device frequency compensation. The addition of 22µF solid tantalum on the output will ensure stability for all operating conditions.
When the adjustment terminal is bypassed with a capacitor to improve the ripple rejection, the requirement for an output capacitor increases. The value of 22µF tantalum covers all cases of bypassing the adjustment terminal. Without bypassing the adjustment terminal smaller capacitors can be used with equally good results.
To ensure good transient response with heavy load current changes capacitor values on the order of 100µF are used in the output of many regulators. To further improve stability and transient response of these devices larger values of output capacitor can be used.

Protection Diodes

Unlike older regulators, the AMS1117 family does not need any protection diodes between the adjustment pin and the output and from the output to the input to prevent over-stressing the die. Internal resistors are limiting the internal current paths on the AMS1117 adjustment pin, therefore even with capacitors on the adjustment pin no protection diode is needed to ensure device safety under short-circuit conditions.
Diodes between the input and output are not usually needed. Microsecond surge currents of 50A to 100A can be handled by the internal diode between the input and output pins of the device. In normal operations it is difficult to get those values of surge currents even with the use of large output capacitances. If high value output capacitors are used, such as 1000µF to 5000µF and the input pin is instantaneously shorted to ground, damage can occur. A diode from output to input is recommended, when a crowbar circuit at the input of the AMS1117 is used (Figure 1).

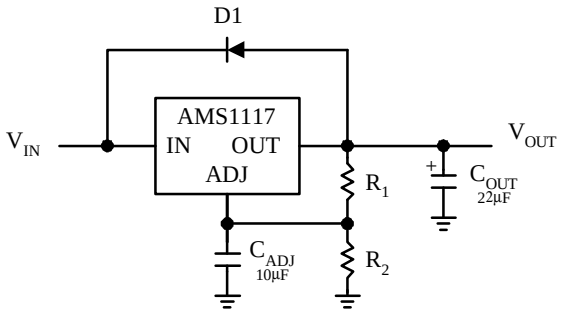
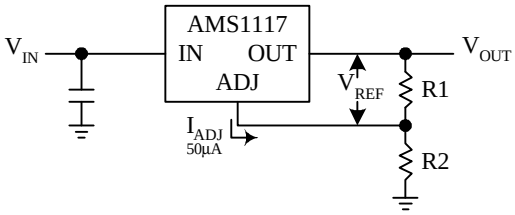


Figure 1.

Output Voltage

The AMS1117 series develops a 1.25V reference voltage between the output and the adjust terminal. Placing a resistor between these two terminals causes a constant current to flow through R1 and down through R2 to set the overall output voltage. This current is normally the specified minimum load current of 10mA. Because I_{ADJ} is very small and constant it represents a small error and it can usually be ignored.



$V_{OUT} = V_{REF} (1+ R2/R1)+I_{ADJ}R2$

Figure 2. Basic Adjustable Regulator

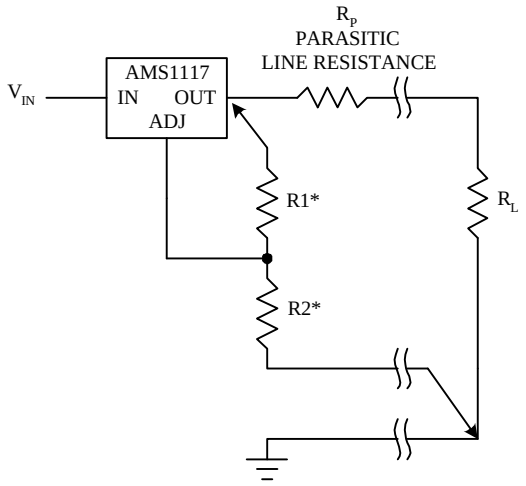
Load Regulation

True remote load sensing it is not possible to provide, because the AMS1117 is a three terminal device. The resistance of the wire connecting the regulator to the load will limit the load regulation. The data sheet specification for load regulation is measured at the bottom of the package. Negative side sensing is a true Kelvin connection, with the bottom of the output divider returned to the negative side of the load.
The best load regulation is obtained when the top of the resistor divider R1 is connected directly to the case not to the load. If R1 were connected to the load, the effective resistance between the regulator and the load would be:

$$R_P \times \left(\frac{R2+R1}{R1} \right), \quad R_P = \text{Parasitic Line Resistance}$$

APPLICATION HINTS

Connected as shown , R_P is not multiplied by the divider ratio



*CONNECT R1 TO CASE
CONNECT R2 TO LOAD

Figure 3. Connections for Best Load Regulation

In the case of fixed voltage devices the top of R1 is connected Kelvin internally, and the ground pin can be used for negative side sensing.

Thermal Considerations

The AMS1117 series have internal power and thermal limiting circuitry designed to protect the device under overload conditions. However maximum junction temperature ratings of 125°C should not be exceeded under continuous normal load conditions.
Careful consideration must be given to all sources of thermal resistance from junction to ambient. For the surface mount package SOT-223 additional heat sources mounted near the device must be considered. The heat dissipation capability of the PC board and its copper traces is used as a heat sink for the device. The thermal resistance from the junction to the tab for the AMS1117 is 15°C/W. Thermal resistance from tab to ambient can be as low as 30°C/W.

The total thermal resistance from junction to ambient can be as low as 45°C/W. This requires a reasonable sized PC board with at least on layer of copper to spread the heat across the board and couple it into the surrounding air.
Experiments have shown that the heat spreading copper layer does not need to be electrically connected to the tab of the device. The PC material can be very effective at transmitting heat between the pad area, attached to the pad of the device, and a ground plane layer either inside or on the opposite side of the board. Although the actual thermal resistance of the PC material is high, the Length/Area ratio of the thermal resistance between layers is small. The data in Table 1, was taken using 1/16" FR-4 board with 1 oz. copper foil, and it can be used as a rough guideline for estimating thermal resistance.
For each application the thermal resistance will be affected by thermal interactions with other components on the board. To determine the actual value some experimentation will be necessary.
The power dissipation of the AMS1117 is equal to:

$P_D = (V_{IN} - V_{OUT})(I_{OUT})$

Maximum junction temperature will be equal to:

$T_J = T_{A(MAX)} + P_D$(Thermal Resistance (junction-to-ambient))

Maximum junction temperature must not exceed 125°C.

Ripple Rejection

The ripple rejection values are measured with the adjustment pin bypassed. The impedance of the adjust pin capacitor at the ripple frequency should be less than the value of R1 (normally 100Ω to 200Ω) for a proper bypassing and ripple rejection approaching the values shown. The size of the required adjust pin capacitor is a function of the input ripple frequency. If R1=100Ω at 120Hz the adjust pin capacitor should be >13µF. At 10kHz only 0.16µF is needed.

The ripple rejection will be a function of output voltage, in circuits without an adjust pin bypass capacitor. The output ripple will increase directly as a ratio of the output voltage to the reference voltage (V_{OUT} / V_{REF}).

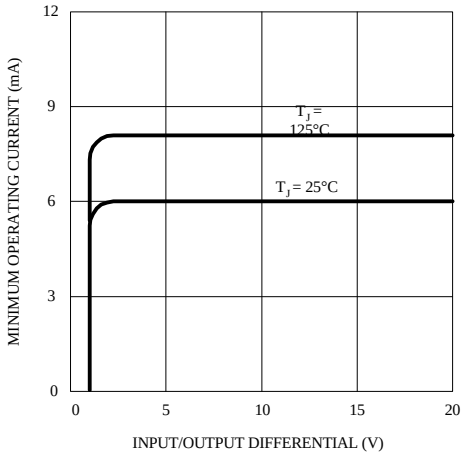
Table 1.

COPPER AREA			THERMAL RESISTANCE
TOP SIDE*	BACK SIDE	BOARD AREA	(JUNCTION-TO-AMBIENT)
2500 Sq. mm	2500 Sq. mm	2500 Sq. mm	45°C/W
1000 Sq. mm	2500 Sq. mm	2500 Sq. mm	45°C/W
225 Sq. mm	2500 Sq. mm	2500 Sq. mm	53°C/W
100 Sq. mm	2500 Sq. mm	2500 Sq. mm	59°C/W
1000 Sq. mm	1000 Sq. mm	1000 Sq. mm	52°C/W
1000 Sq. mm	0	1000 Sq. mm	55°C/W

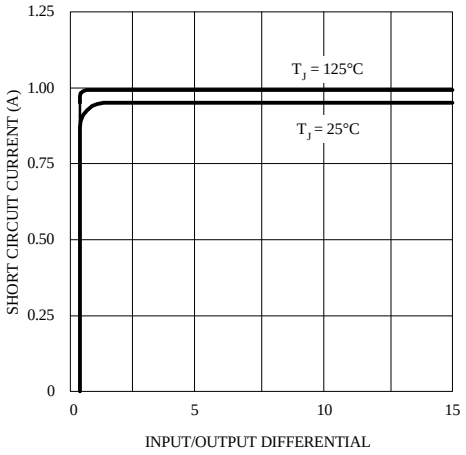
* Tab of device attached to topside copper.

TYPICAL PERFORMANCE CHARACTERISTICS

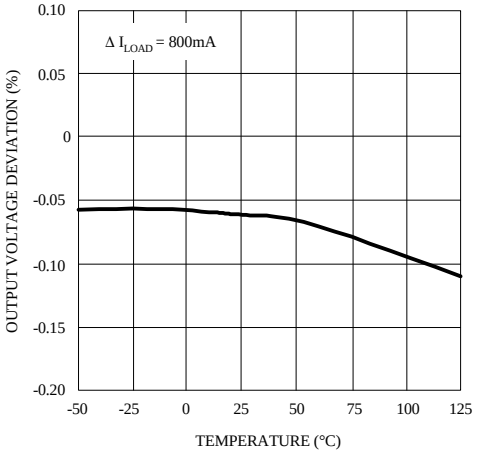
Minimum Operating Current (Adjustable Device)



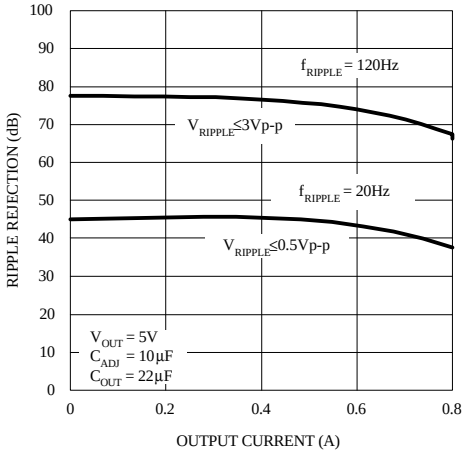
Short-Circuit Current



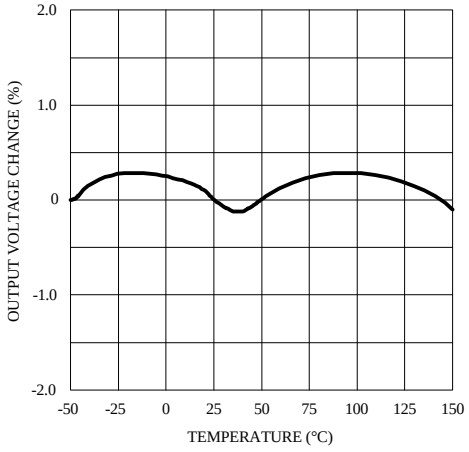
Load Regulation



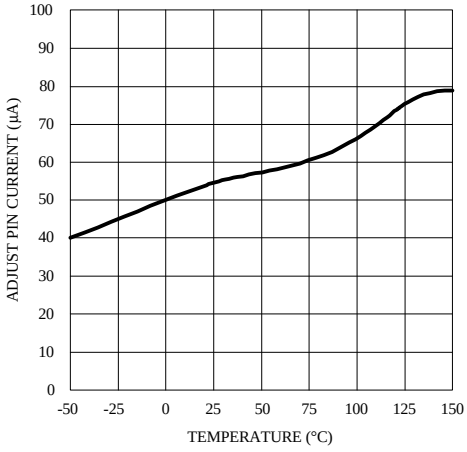
Ripple Rejection vs. Current



Temperature Stability

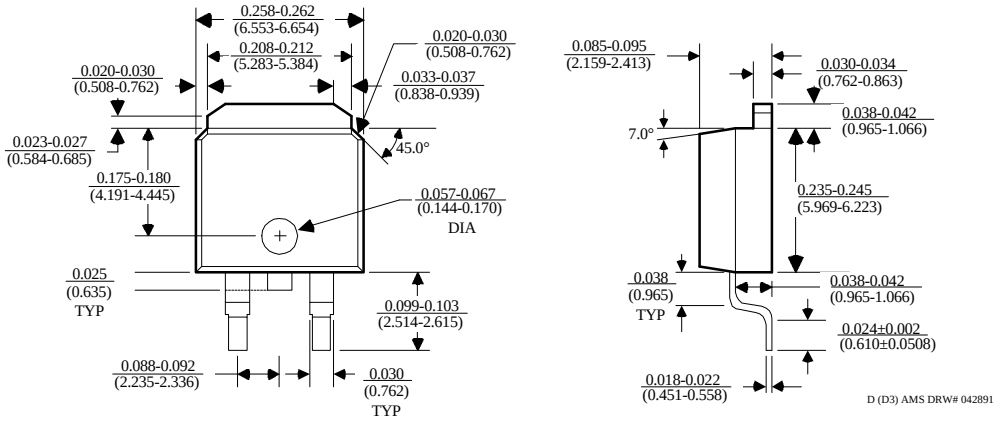


Adjust Pin Current

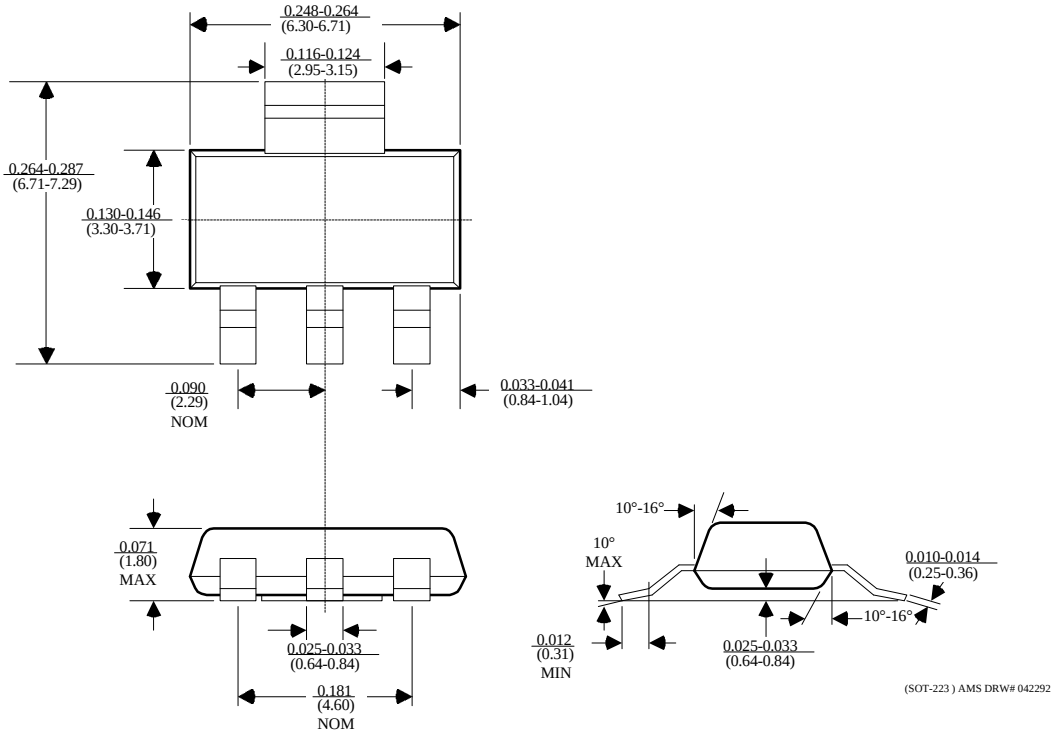


PACKAGE DIMENSIONS inches (millimeters) unless otherwise noted.

TO-252 PLASTIC PACKAGE (D)



3 LEAD SOT-223 PLASTIC PACKAGE



Low Voltage Temperature Sensors

Data Sheet

TMP35/TMP36/TMP37

FEATURES

Low voltage operation (2.7 V to 5.5 V)
Calibrated directly in °C
10 mV/°C scale factor (20 mV/°C on TMP37)
±2°C accuracy over temperature (typ)
±0.5°C linearity (typ)
Stable with large capacitive loads
Specified −40°C to +125°C, operation to +150°C
Less than 50 µA quiescent current
Shutdown current 0.5 µA max
Low self-heating
Qualified for automotive applications

APPLICATIONS

Environmental control systems
Thermal protection
Industrial process control
Fire alarms
Power system monitors
CPU thermal management

GENERAL DESCRIPTION

The TMP35/TMP36/TMP37 are low voltage, precision centigrade temperature sensors. They provide a voltage output that is linearly proportional to the Celsius (centigrade) temperature. The TMP35/TMP36/TMP37 do not require any external calibration to provide typical accuracies of ±1°C at +25°C and ±2°C over the −40°C to +125°C temperature range.

The low output impedance of the TMP35/TMP36/TMP37 and its linear output and precise calibration simplify interfacing to temperature control circuitry and ADCs. All three devices are intended for single-supply operation from 2.7 V to 5.5 V maximum. The supply current runs well below 50 µA, providing very low self-heating—less than 0.1°C in still air. In addition, a shutdown function is provided to cut the supply current to less than 0.5 µA.

The TMP35 is functionally compatible with the LM35/LM45 and provides a 250 mV output at 25°C. The TMP35 reads temperatures from 10°C to 125°C. The TMP36 is specified from −40°C to +125°C, provides a 750 mV output at 25°C, and operates to 125°C from a single 2.7 V supply. The TMP36 is functionally compatible with the LM50. Both the TMP35 and TMP36 have an output scale factor of 10 mV/°C.

FUNCTIONAL BLOCK DIAGRAM

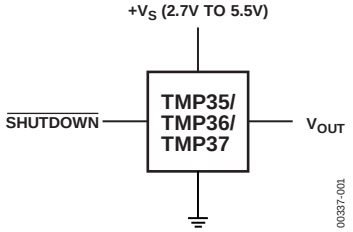


Figure 1.

PIN CONFIGURATIONS

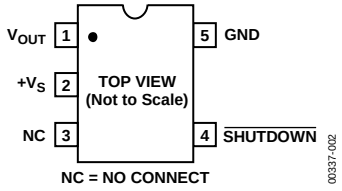


Figure 2. RJ-5 (SOT-23)

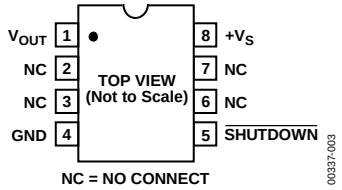


Figure 3. R-8 (SOIC_N)

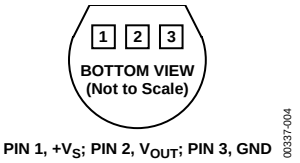


Figure 4. T-3 (TO-92)

The TMP37 is intended for applications over the range of 5°C to 100°C and provides an output scale factor of 20 mV/°C. The TMP37 provides a 500 mV output at 25°C. Operation extends to 150°C with reduced accuracy for all devices when operating from a 5 V supply.

The TMP35/TMP36/TMP37 are available in low cost 3-lead TO-92, 8-lead SOIC_N, and 5-lead SOT-23 surface-mount packages.

Rev. H **Document Feedback**
Information furnished by Analog Devices is believed to be accurate and reliable. However, no responsibility is assumed by Analog Devices for its use, nor for any infringements of patents or other rights of third parties that may result from its use. Specifications subject to change without notice. No license is granted by implication or otherwise under any patent or patent rights of Analog Devices. Trademarks and registered trademarks are the property of their respective owners.

One Technology Way, P.O. Box 9106, Norwood, MA 02062-9106, U.S.A.
Tel: 781.329.4700 **©1996–2015 Analog Devices, Inc. All rights reserved.**
Technical Support **www.analog.com**

TABLE OF CONTENTS

Features ... 1
Applications ... 1
General Description ... 1
Functional Block Diagram ... 1
Pin Configurations ... 1
Revision History ... 2
Specifications ... 3
Absolute Maximum Ratings ... 4
Thermal Resistance ... 4
ESD Caution ... 4
Typical Performance Characteristics ... 5
Functional Description ... 8
Applications Information ... 9
Shutdown Operation ... 9
Mounting Considerations ... 9
Thermal Environment Effects ... 9
Basic Temperature Sensor Connections ... 10
Fahrenheit Thermometers ... 10
Average and Differential Temperature Measurement ... 12
Microprocessor Interrupt Generator ... 13
Thermocouple Signal Conditioning with Cold-Junction Compensation ... 14
Using TMP35/TMP36/TMP37 Sensors in Remote Locations ... 15
Temperature to 4–20 mA Loop Transmitter ... 15
Temperature-to-Frequency Converter ... 16
Driving Long Cables or Heavy Capacitive Loads ... 17
Commentary on Long-Term Stability ... 17
Outline Dimensions ... 18
Ordering Guide ... 19
Automotive Products ... 19

REVISION HISTORY

5/15—Rev. G to Rev. H

Changed TMP3x to TMP35/TMP36/TMP37 ... Throughout
Changes to Figure 28 ... 12
Changes to Ordering Guide ... 19

11/13—Rev. F to Rev. G

Change to Table 1, Long-Term Stability Parameter ... 3
Change to Caption for Figure 38 ... 18
Changes to Ordering Guide ... 19

11/10—Rev. E to Rev. F

Changes to Features ... 1
Updated Outline Dimensions ... 18
Changes to Ordering Guide ... 19
Added Automotive Products Section ... 20

8/08—Rev. D to Rev. E

Updated Outline Dimensions ... 18
Changes to Ordering Guide ... 19

3/05—Rev. C to Rev. D

Updated Format ... Universal
Changes to Specifications ... 3
Additions to Absolute Maximum Ratings ... 4
Updated Outline Dimensions ... 18
Changes to Ordering Guide ... 19

10/02—Rev. B to Rev. C

Changes to Specifications ... 3
Deleted Text from Commentary on Long-Term Stability Section ... 13
Updated Outline Dimensions ... 14

9/01—Rev. A to Rev. B

Edits to Specifications ... 2
Addition of New Figure 1 ... 2
Deletion of Wafer Test Limits Section ... 3

6/97—Rev. 0 to Rev. A

3/96—Revision 0: Initial Version

SPECIFICATIONS

V_S = 2.7 V to 5.5 V, −40°C ≤ T_A ≤ +125°C, unless otherwise noted.

Table 1.

Parameter[1]	Symbol	Test Conditions/Comments	Min	Typ	Max	Unit
ACCURACY						
TMP35/TMP36/TMP37 (F Grade)		T_A = 25°C		±1	±2	°C
TMP35/TMP36/TMP37 (G Grade)		T_A = 25°C		±1	±3	°C
TMP35/TMP36/TMP37 (F Grade)		Over rated temperature		±2	±3	°C
TMP35/TMP36/TMP37 (G Grade)		Over rated temperature		±2	±4	°C
Scale Factor, TMP35		10°C ≤ T_A ≤ 125°C		10		mV/°C
Scale Factor, TMP36		−40°C ≤ T_A ≤ +125°C		10		mV/°C
Scale Factor, TMP37		5°C ≤ T_A ≤ 85°C		20		mV/°C
		5°C ≤ T_A ≤ 100°C		20		mV/°C
		3.0 V ≤ V_S ≤ 5.5 V				
Load Regulation		0 µA ≤ I_L ≤ 50 µA				
		−40°C ≤ T_A ≤ +105°C		6	20	m°C/µA
		−105°C ≤ T_A ≤ +125°C		25	60	m°C/µA
Power Supply Rejection Ratio	PSRR	T_A = 25°C		30	100	m°C/V
		3.0 V ≤ V_S ≤ 5.5 V		50		m°C/V
Linearity				0.5		°C
Long-Term Stability		T_A = 150°C for 1000 hours		0.4		°C
SHUTDOWN						
Logic High Input Voltage	V_{IH}	V_S = 2.7 V	1.8			V
Logic Low Input Voltage	V_{IL}	V_S = 5.5 V			400	mV
OUTPUT						
TMP35 Output Voltage		T_A = 25°C		250		mV
TMP36 Output Voltage		T_A = 25°C		750		mV
TMP37 Output Voltage		T_A = 25°C		500		mV
Output Voltage Range			100		2000	mV
Output Load Current	I_L		0		50	µA
Short-Circuit Current	I_{SC}	Note 2			250	µA
Capacitive Load Driving	C_L	No oscillations[2]	1000	10000		pF
Device Turn-On Time		Output within ±1°C, 100 kΩ||100 pF load[2]		0.5	1	ms
POWER SUPPLY						
Supply Range	V_S		2.7		5.5	V
Supply Current	I_{SY} (ON)	Unloaded			50	µA
Supply Current (Shutdown)	I_{SY} (OFF)	Unloaded		0.01	0.5	µA

[1] Does not consider errors caused by self-heating.
[2] Guaranteed but not tested.

ABSOLUTE MAXIMUM RATINGS

Table 2.

Parameter[1, 2]	Rating
Supply Voltage	7 V
Shutdown Pin	GND ≤ $\overline{\text{SHUTDOWN}}$ ≤ $+V_S$
Output Pin	GND ≤ V_{OUT} ≤ $+V_S$
Operating Temperature Range	−55°C to +150°C
Die Junction Temperature	175°C
Storage Temperature Range	−65°C to +160°C
IR Reflow Soldering	
Peak Temperature	220°C (0°C/5°C)
Time at Peak Temperature Range	10 sec to 20 sec
Ramp-Up Rate	3°C/sec
Ramp-Down Rate	−6°C/sec
Time 25°C to Peak Temperature	6 min
IR Reflow Soldering—Pb-Free Package	
Peak Temperature	260°C (0°C)
Time at Peak Temperature Range	20 sec to 40 sec
Ramp-Up Rate	3°C/sec
Ramp-Down Rate	−6°C/sec
Time 25°C to Peak Temperature	8 min

[1] Digital inputs are protected; however, permanent damage can occur on unprotected units from high energy electrostatic fields. Keep units in conductive foam or packaging at all times until ready to use. Use proper antistatic handling procedures.
[2] Remove power before inserting or removing units from their sockets.

Stresses at or above those listed under Absolute Maximum Ratings may cause permanent damage to the product. This is a stress rating only; functional operation of the product at these or any other conditions above those indicated in the operational section of this specification is not implied. Operation beyond the maximum operating conditions for extended periods may affect product reliability.

THERMAL RESISTANCE

θ_{JA} is specified for the worst-case conditions, that is, a device in socket.

Table 3. Thermal Resistance

Package Type	θ_{JA}	θ_{JC}	Unit
TO-92 (T-3-1)	162	120	°C/W
SOIC_N (R-8)	158	43	°C/W
SOT-23 (RJ-5)	300	180	°C/W

ESD CAUTION

ESD (electrostatic discharge) sensitive device. Charged devices and circuit boards can discharge without detection. Although this product features patented or proprietary protection circuitry, damage may occur on devices subjected to high energy ESD. Therefore, proper ESD precautions should be taken to avoid performance degradation or loss of functionality.

TYPICAL PERFORMANCE CHARACTERISTICS

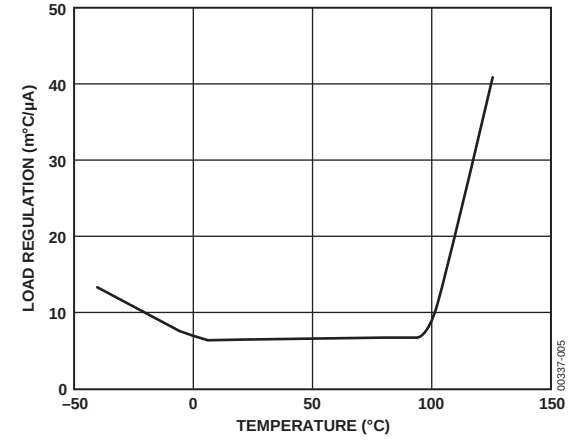


Figure 5. Load Regulation vs. Temperature (m°C/µA)

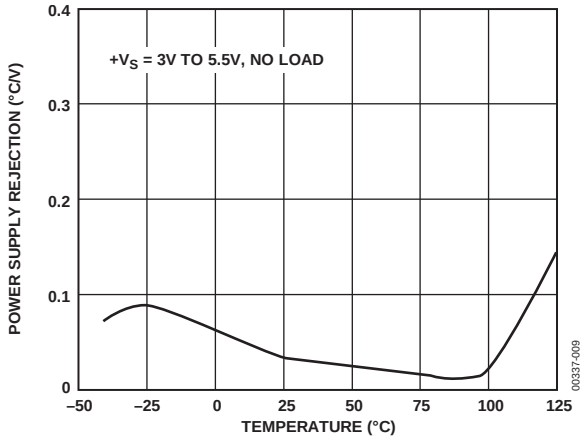


Figure 8. Power Supply Rejection vs. Temperature

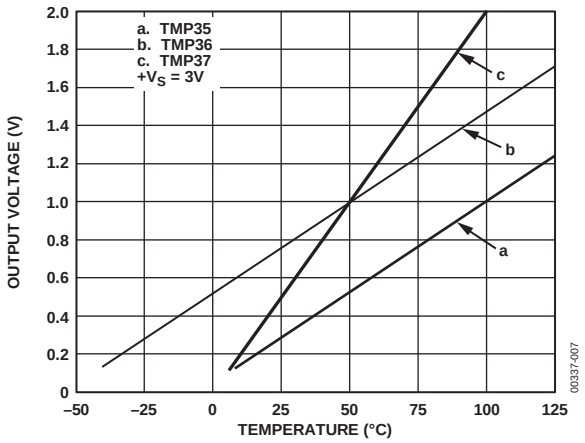


Figure 6. Output Voltage vs. Temperature

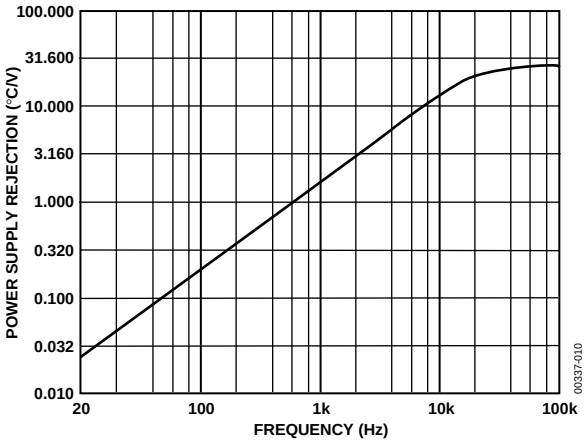


Figure 9. Power Supply Rejection vs. Frequency

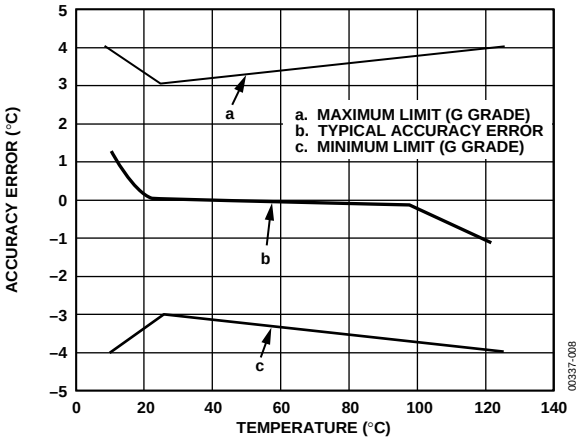


Figure 7. Accuracy Error vs. Temperature

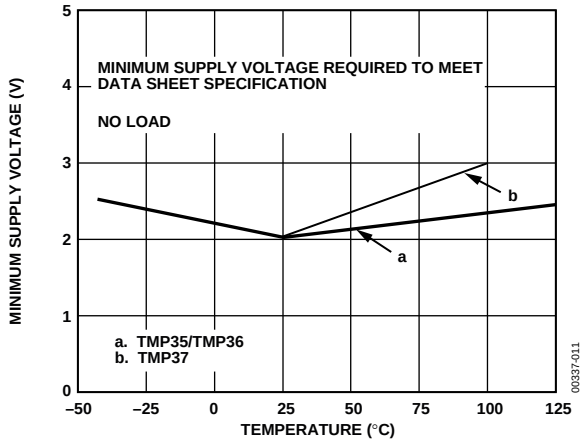


Figure 10. Minimum Supply Voltage vs. Temperature

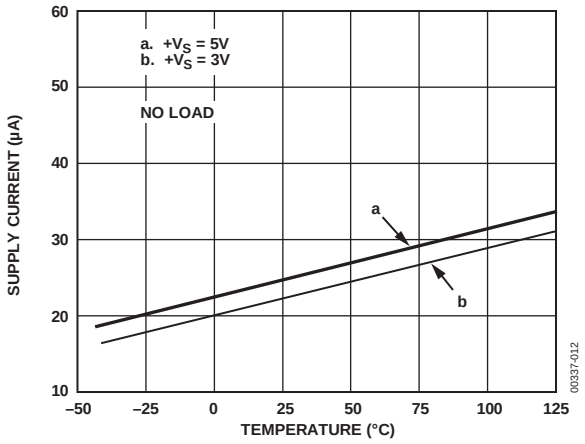


Figure 11. Supply Current vs. Temperature

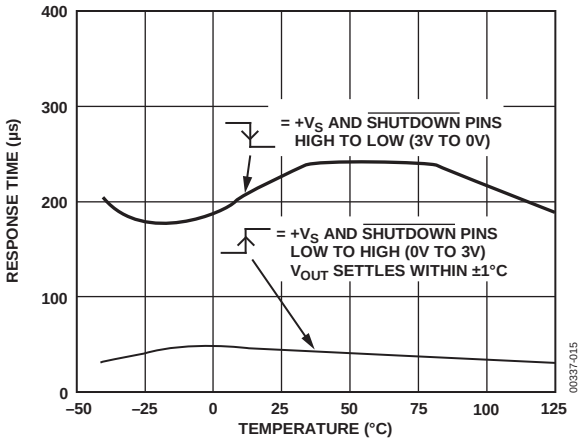


Figure 14. V_{OUT} Response Time for $+V_S$ Power-Up/Power-Down vs. Temperature

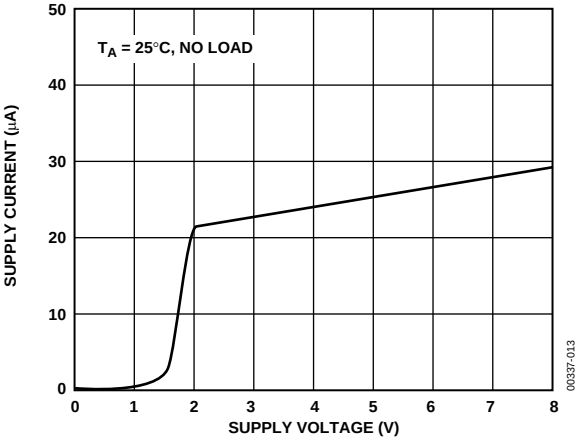


Figure 12. Supply Current vs. Supply Voltage

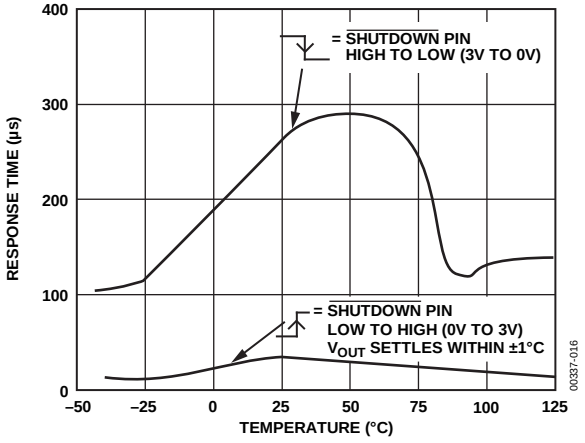


Figure 15. V_{OUT} Response Time for $\overline{SHUTDOWN}$ Pin vs. Temperature

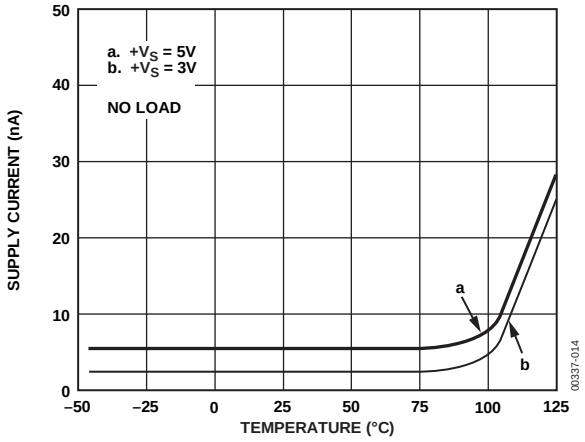


Figure 13. Supply Current vs. Temperature (Shutdown = 0 V)

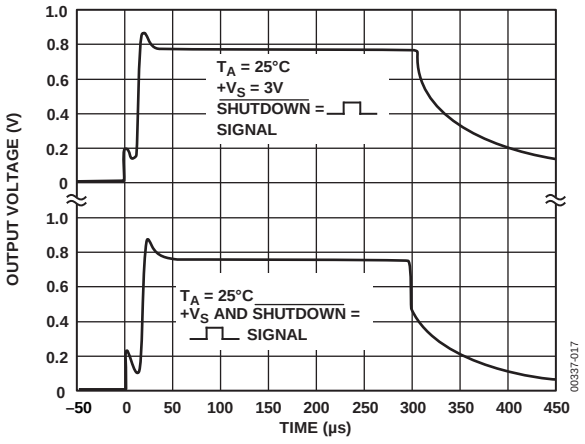


Figure 16. V_{OUT} Response Time to $\overline{SHUTDOWN}$ Pin and $+V_S$ Pin vs. Time

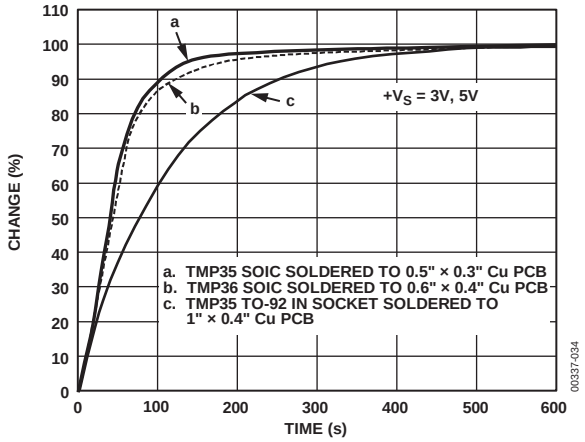


Figure 17. Thermal Response Time in Still Air

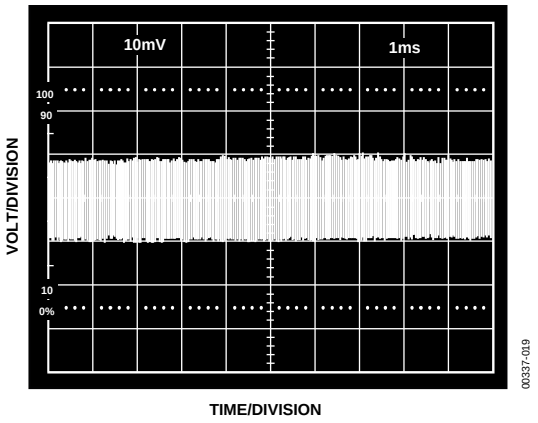


Figure 20. Temperature Sensor Wideband Output Noise Voltage; Gain = 100, BW = 157 kHz

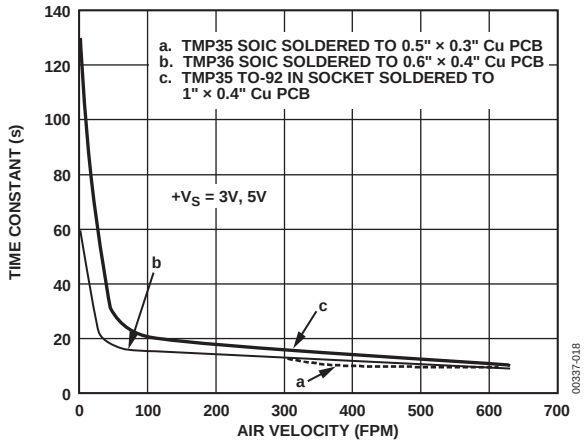


Figure 18. Thermal Response Time Constant in Forced Air

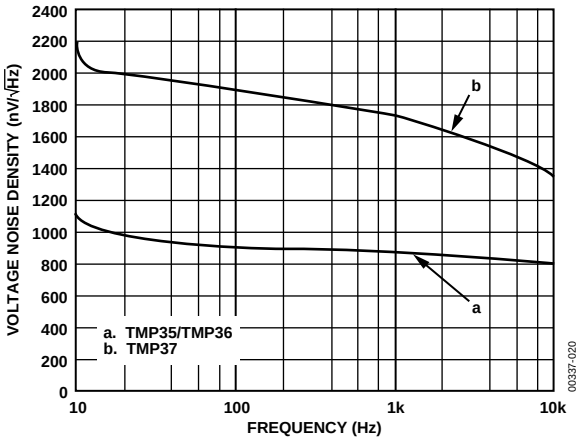


Figure 21. Voltage Noise Spectral Density vs. Frequency

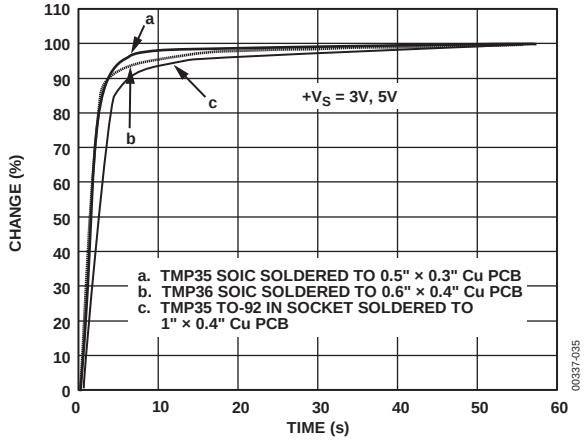


Figure 19. Thermal Response Time in Stirred Oil Bath

FUNCTIONAL DESCRIPTION

An equivalent circuit for the TMP35/TMP36/TMP37 micropower, centigrade temperature sensors is shown in Figure 22. The core of the temperature sensor is a band gap core that comprises transistors Q1 and Q2, biased by Q3 to approximately 8 μA. The band gap core operates both Q1 and Q2 at the same collector current level; however, because the emitter area of Q1 is 10 times that of Q2, the V_{BE} of Q1 and the V_{BE} of Q2 are not equal by the following relationship:

$$\Delta V_{BE} = V_T \times \ln\left(\frac{A_{E,Q1}}{A_{E,Q2}}\right)$$

Resistors R1 and R2 are used to scale this result to produce the output voltage transfer characteristic of each temperature sensor and, simultaneously, R2 and R3 are used to scale the V_{BE} of Q1 as an offset term in V_{OUT}. Table 4 summarizes the differences in the output characteristics of the three temperature sensors.

The output voltage of the temperature sensor is available at the emitter of Q4, which buffers the band gap core and provides load current drive. The current gain of Q4, working with the available base current drive from the previous stage, sets the short-circuit current limit of these devices to 250 μA.

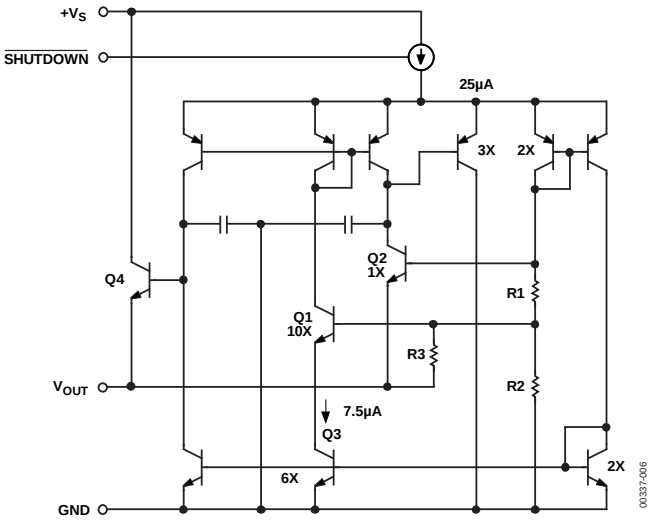


Figure 22. Temperature Sensor Simplified Equivalent Circuit

Table 4. TMP35/TMP36/TMP37 Output Characteristics

Sensor	Offset Voltage (V)	Output Voltage Scaling (mV/°C)	Output Voltage at 25°C (mV)
TMP35	0	10	250
TMP36	0.5	10	750
TMP37	0	20	500

APPLICATIONS INFORMATION

SHUTDOWN OPERATION

All TMP35/TMP36/TMP37 devices include a shutdown capability, which reduces the power supply drain to less than 0.5 µA maximum. This feature, available only in the SOIC_N and the SOT-23 packages, is TTL/CMOS level-compatible, provided that the temperature sensor supply voltage is equal in magnitude to the logic supply voltage. Internal to the TMP35/TMP36/TMP37 at the $\overline{\text{SHUTDOWN}}$ pin, a pull-up current source to $+V_S$ is connected. This allows the $\overline{\text{SHUTDOWN}}$ pin to be driven from an open-collector/drain driver. A logic low, or zero-volt condition, on the $\overline{\text{SHUTDOWN}}$ pin is required to turn off the output stage. During shutdown, the output of the temperature sensors becomes high impedance where the potential of the output pin is then determined by external circuitry. If the shutdown feature is not used, it is recommended that the $\overline{\text{SHUTDOWN}}$ pin be connected to $+V_S$ (Pin 8 on the SOIC_N; Pin 2 on the SOT-23).

The shutdown response time of these temperature sensors is shown in Figure 14, Figure 15, and Figure 16.

MOUNTING CONSIDERATIONS

If the TMP35/TMP36/TMP37 temperature sensors are thermally attached and protected, they can be used in any temperature measurement application where the maximum temperature range of the medium is between −40°C and +125°C. Properly cemented or glued to the surface of the medium, these sensors are within 0.01°C of the surface temperature. Caution should be exercised, especially with T-3 packages, because the leads and any wiring to the device can act as heat pipes, introducing errors if the surrounding air-surface interface is not isothermal. Avoiding this condition is easily achieved by dabbing the leads of the temper-ature sensor and the hookup wires with a bead of thermally conductive epoxy. This ensures that the TMP35/TMP36/TMP37 die temperature is not affected by the surrounding air temperature. Because plastic IC packaging technology is used, excessive mechanical stress should be avoided when fastening the device with a clamp or a screw-on heat tab. Thermally conductive epoxy or glue, which must be electrically nonconductive, is recommended under typical mounting conditions.

These temperature sensors, as well as any associated circuitry, should be kept insulated and dry to avoid leakage and corrosion. In wet or corrosive environments, any electrically isolated metal or ceramic well can be used to shield the temperature sensors. Condensation at very cold temperatures can cause errors and should be avoided by sealing the device, using electrically non-conductive epoxy paints or dip or any one of the many printed circuit board coatings and varnishes.

THERMAL ENVIRONMENT EFFECTS

The thermal environment in which the TMP35/TMP36/TMP37 sensors are used determines two important characteristics: self-heating effects and thermal response time. Figure 23 illustrates a thermal model of the TMP35/TMP36/TMP37 temperature sensors, which is useful in under-standing these characteristics.

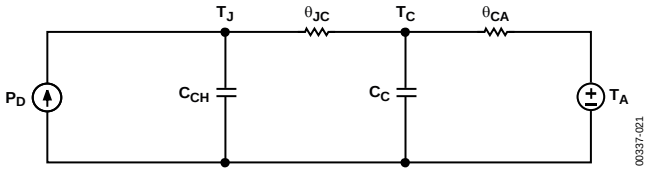


Figure 23. Thermal Circuit Model

In the T-3 package, the thermal resistance junction-to-case, θ_{JC}, is 120°C/W. The thermal resistance case-to-ambient, C_A, is the difference between θ_{JA} and θ_{JC}, and is determined by the char-acteristics of the thermal connection. The power dissipation of the temperature sensor, P_D, is the product of the total voltage across the device and its total supply current, including any current delivered to the load. The rise in die temperature above the ambient temperature of the medium is given by

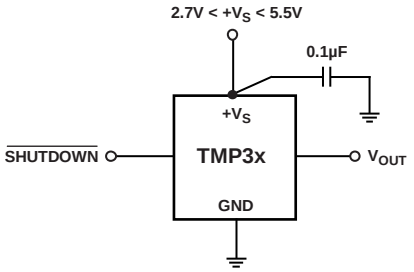
$$T_J = P_D \times (\theta_{JC} + \theta_{CA}) + T_A$$

Thus, the die temperature rise of a TMP35 SOT-23 package mounted into a socket in still air at 25°C and driven from a 5 V supply is less than 0.04°C.

The transient response of the TMP35/TMP36/TMP37 sensors to a step change in the temperature is determined by the thermal resistances and the thermal capacities of the die, C_{CH}, and the case, C_C. The thermal capacity of C_C varies with the measurement medium because it includes anything in direct contact with the package. In all practical cases, the thermal capacity of C_C is the limiting factor in the thermal response time of the sensor and can be represented by a single-pole RC time constant response. Figure 17 and Figure 19 show the thermal response time of the TMP35/TMP36/TMP37 sensors under various conditions. The thermal time constant of a temperature sensor is defined as the time required for the sensor to reach 63.2% of the final value for a step change in the temperature. For example, the thermal time constant of a TMP35 SOIC package sensor mounted onto a 0.5" × 0.3" PCB is less than 50 sec in air, whereas in a stirred oil bath, the time constant is less than 3 sec.

BASIC TEMPERATURE SENSOR CONNECTIONS

Figure 24 illustrates the basic circuit configuration for the TMP35/TMP36/TMP37 temperature sensors. The table in Figure 24 shows the pin assignments of the temperature sensors for the three package types. For the SOT-23, Pin 3 is labeled NC, as are Pin 2, Pin 3, Pin 6, and Pin 7 on the SOIC_N package. It is recommended that no electrical connections be made to these pins. If the shutdown feature is not needed on the SOT-23 or on the SOIC_N package, the $\overline{\text{SHUTDOWN}}$ pin should be connected to $+V_S$.



PIN ASSIGNMENTS

PACKAGE	$+V_S$	GND	V_{OUT}	$\overline{\text{SHUTDOWN}}$
SOIC_N	8	4	1	5
SOT-23	2	5	1	4
TO-92	1	3	2	NA

00337-022

Figure 24. Basic Temperature Sensor Circuit Configuration

Note the 0.1 µF bypass capacitor on the input. This capacitor should be a ceramic type, have very short leads (surface-mount is preferable), and be located as close as possible in physical proximity to the temperature sensor supply pin. Because these temperature sensors operate on very little supply current and may be exposed to very hostile electrical environments, it is important to minimize the effects of radio frequency interference (RFI) on these devices. The effect of RFI on these temperature sensors specifically and on analog ICs in general is manifested as abnormal dc shifts in the output voltage due to the rectification of the high frequency ambient noise by the IC. When the devices are operated in the presence of high frequency radiated or conducted noise, a large value tantalum capacitor (±2.2 µF) placed across the 0.1 µF ceramic capacitor may offer additional noise immunity.

FAHRENHEIT THERMOMETERS

Although the TMP35/TMP36/TMP37 temperature sensors are centigrade temperature sensors, a few components can be used to convert the output voltage and transfer characteristics to directly read Fahrenheit temperatures. Figure 25 shows an example of a simple Fahrenheit thermometer using either the TMP35 or the TMP37. Using the TMP35, this circuit can be used to sense temperatures from 41°F to 257°F with an output transfer characteristic of 1 mV/°F; using the TMP37, this circuit can be used to sense temperatures from 41°F to 212°F with an output transfer characteristic of 2 mV/°F. This particular approach does not lend itself to the TMP36 because of its inherent 0.5 V output offset. The circuit is constructed with an AD589, a 1.23 V voltage reference, and four resistors whose values for each sensor are shown in the table in Figure 25. The scaling of the output resistance levels ensures minimum output loading on the temp-erature sensors. A generalized expression for the transfer equation of the circuit is given by

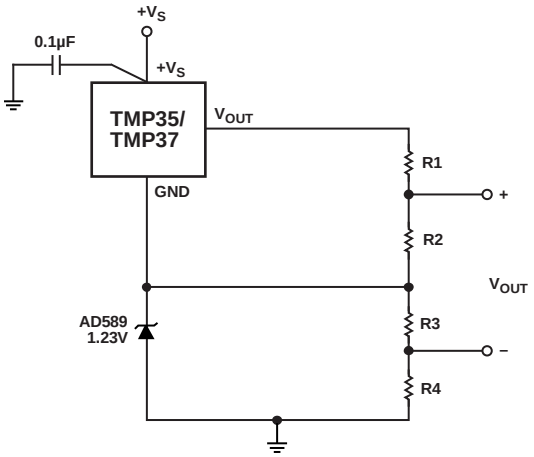
$$V_{OUT} = \left(\frac{R1}{R1+R2}\right)(TMP35) + \left(\frac{R3}{R3+R4}\right)(AD589)$$

where:

TMP35 is the output voltage of the TMP35 or the TMP37 at the measurement temperature, T_M.

AD589 is the output voltage of the reference, that is, 1.23 V.

The output voltage of this circuit is not referenced to the circuit's common ground. If this output voltage were applied directly to the input of an ADC, the ADC common ground should be adjusted accordingly.



SENSOR	TCV_{OUT}	R1 (kΩ)	R2 (kΩ)	R3 (kΩ)	R4 (kΩ)
TMP35	1mV/°F	45.3	10	10	374
TMP37	2mV/°F	45.3	10	10	182

00337-023

Figure 25. TMP35/TMP37 Fahrenheit Thermometers

The same circuit principles can be applied to the TMP36, but because of the inherent offset of the TMP36, the circuit uses only two resistors, as shown in Figure 26. In this circuit, the output voltage transfer characteristic is 1 mV/°F but is referenced to the common ground of the circuit; however, there is a 58 mV (58°F) offset in the output voltage. For example, the output voltage of the circuit reads 18 mV if the TMP36 is placed in a −40°F ambient environment and 315 mV at +257°F.

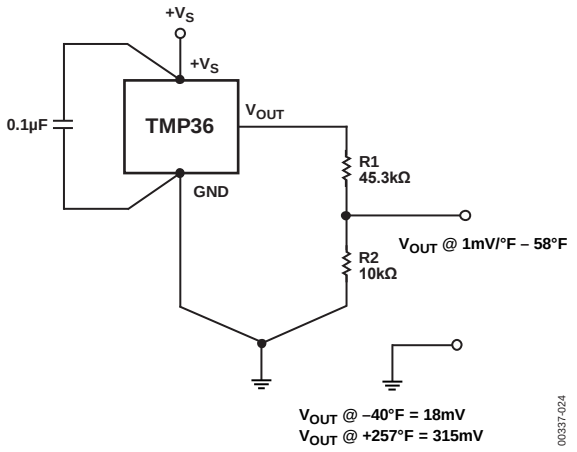
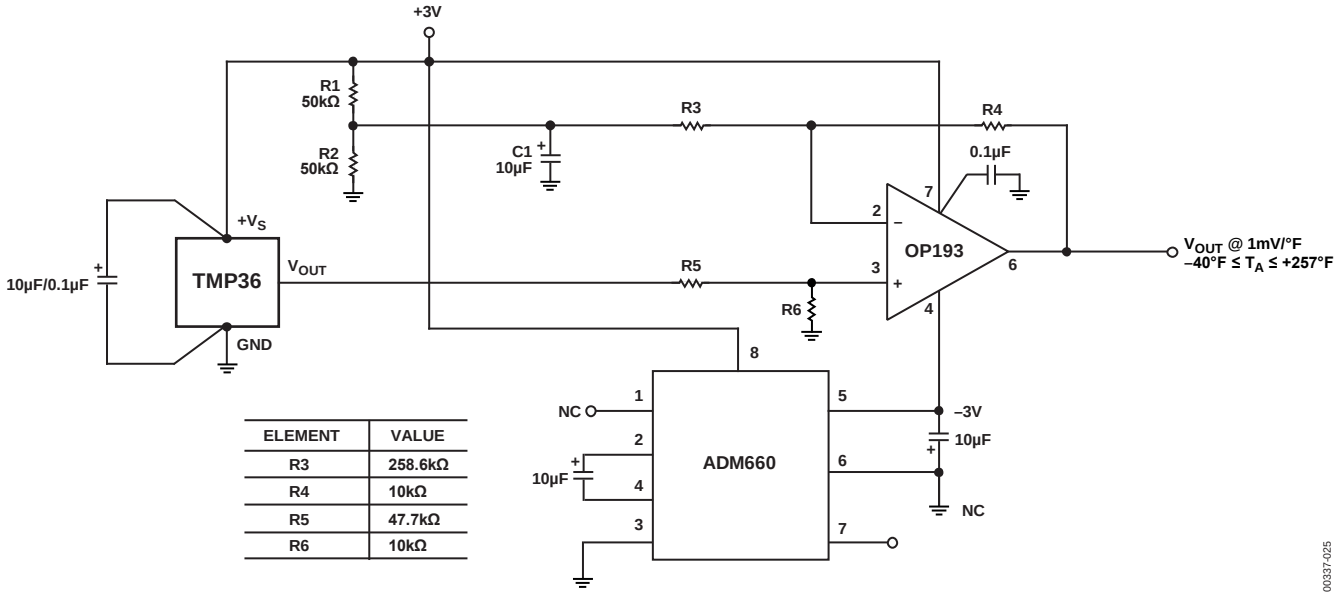


Figure 26. TMP36 Fahrenheit Thermometer Version 1

At the expense of additional circuitry, the offset produced by the circuit in Figure 26 can be avoided by using the circuit in Figure 27. In this circuit, the output of the TMP36 is conditioned by a single-supply, micropower op amp, the OP193. Although the entire circuit operates from a single 3 V supply, the output voltage of the circuit reads the temperature directly, with a transfer characteristic of 1 mV/°F, without offset. This is accomplished through an ADM660, which is a supply voltage inverter. The 3 V supply is inverted and applied to the V− terminal of the OP193. Thus, for a temperature range between −40°F and +257°F, the output of the circuit reads −40 mV to +257 mV. A general expression for the transfer equation of the circuit is given by

$$V_{OUT} = \left(\frac{R6}{R5+R6}\right)\left(1+\frac{R4}{R3}\right)(TMP36) - \left(\frac{R4}{R3}\right)\left(\frac{V_S}{2}\right)$$



ELEMENT	VALUE
R3	258.6kΩ
R4	10kΩ
R5	47.7kΩ
R6	10kΩ

Figure 27. TMP36 Fahrenheit Thermometer Version 2

AVERAGE AND DIFFERENTIAL TEMPERATURE MEASUREMENT

In many commercial and industrial environments, temperature sensors often measure the average temperature in a building, or the difference in temperature between two locations on a factory floor or in an industrial process. The circuits in Figure 28 and Figure 29 demonstrate an inexpensive approach to average and differential temperature measurement.

In Figure 28, an OP193 sums the outputs of three temperature sensors to produce an output voltage scaled by 10 mV/°C that represents the average temperature at three locations. The circuit can be extended to include as many temperature sensors as required as long as the transfer equation of the circuit is maintained. In this application, it is recommended that one temperature sensor type be used throughout the circuit; otherwise, the output voltage of the circuit cannot produce an accurate reading of the various ambient conditions.

The circuit in Figure 29 illustrates how a pair of TMP35/TMP36/TMP37 sensors used with an OP193 configured as a difference amplifier can read the difference in temperature between two locations. In these applications, it is always possible that one temperature sensor is reading a temperature below that of the other sensor. To accommodate this condition, the output of the OP193 is offset to a voltage at one-half the supply via R5 and R6. Thus, the output voltage of the circuit is measured relative to this point, as shown in Figure 29. Using the TMP36, the output voltage of the circuit is scaled by 10 mV/°C. To minimize the error in the difference between the two measured temperatures, a common, readily available thin-film resistor network is used for R1 to R4.

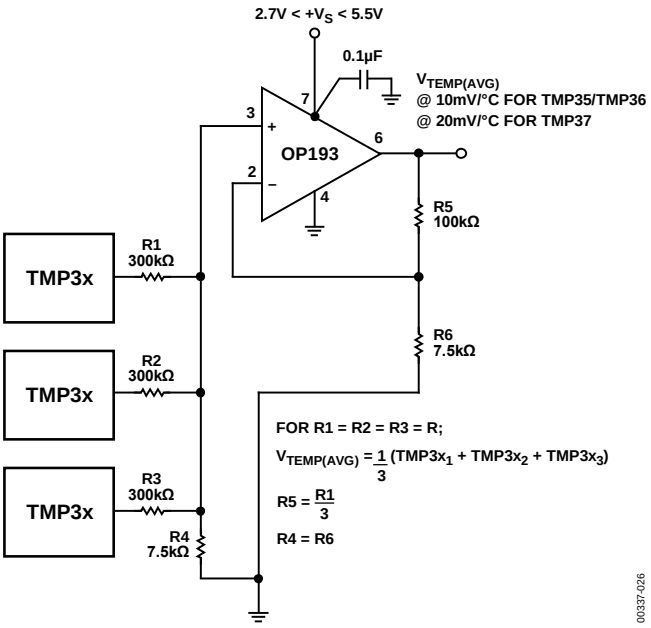


Figure 28. Configuring Multiple Sensors for Average Temperature Measurements

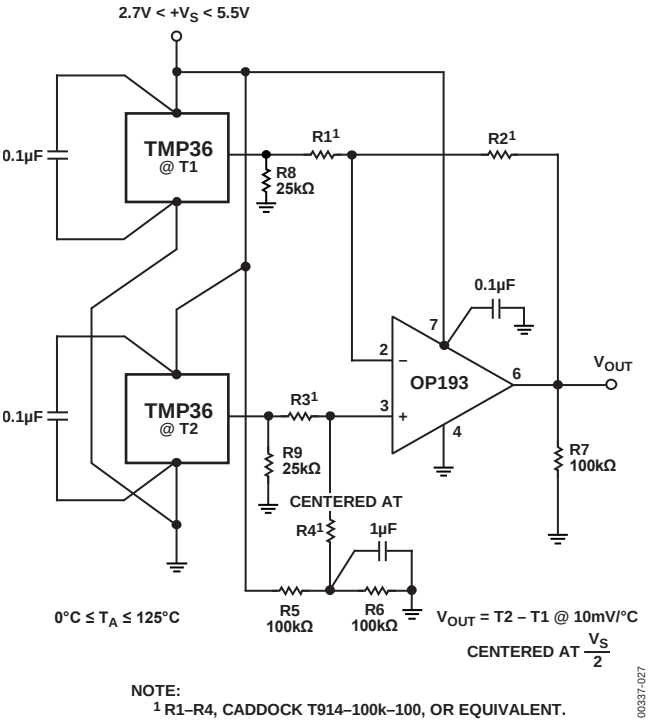


Figure 29. Configuring Multiple Sensors for Differential Temperature Measurements

MICROPROCESSOR INTERRUPT GENERATOR

These inexpensive temperature sensors can be used with a voltage reference and an analog comparator to configure an interrupt generator for microprocessor applications. With the popularity of fast microprocessors, the need to indicate a microprocessor overtemperature condition has grown tremendously. The circuit in Figure 30 demonstrates one way to generate an interrupt using a TMP35, a CMP402 analog comparator, and a REF191, a 2 V precision voltage reference.

The circuit is designed to produce a logic high interrupt signal if the microprocessor temperature exceeds 80°C. This 80°C trip point was arbitrarily chosen (final value set by the microprocessor thermal reference design) and is set using an R3 to R4 voltage divider of the REF191 output voltage. Because the output of the TMP35 is scaled by 10 mV/°C, the voltage at the inverting terminal of the CMP402 is set to 0.8 V.

Because temperature is a slowly moving quantity, the possibility for comparator chatter exists. To avoid this condition, hysteresis is used around the comparator. In this application, a hysteresis of 5°C about the trip point was arbitrarily chosen; the ultimate value for hysteresis should be determined by the end application. The output logic voltage swing of the comparator with R1 and R2 determines the amount of comparator hysteresis. Using a 3.3 V supply, the output logic voltage swing of the CMP402 is 2.6 V; therefore, for a hysteresis of 5°C (50 mV at 10 mV/°C), R1 is set to 20 kΩ, and R2 is set to 1 MΩ. An expression for the hysteresis of this circuit is given by

$$V_{HYS} = \left(\frac{R1}{R2}\right)\left(V_{LOGIC\ SWING,\ CMP402}\right)$$

Because this circuit is probably used in close proximity to high speed digital circuits, R1 is split into equal values and a 1000 pF capacitor is used to form a low-pass filter on the output of the TMP35. Furthermore, to prevent high frequency noise from contaminating the comparator trip point, a 0.1 µF capacitor is used across R4.

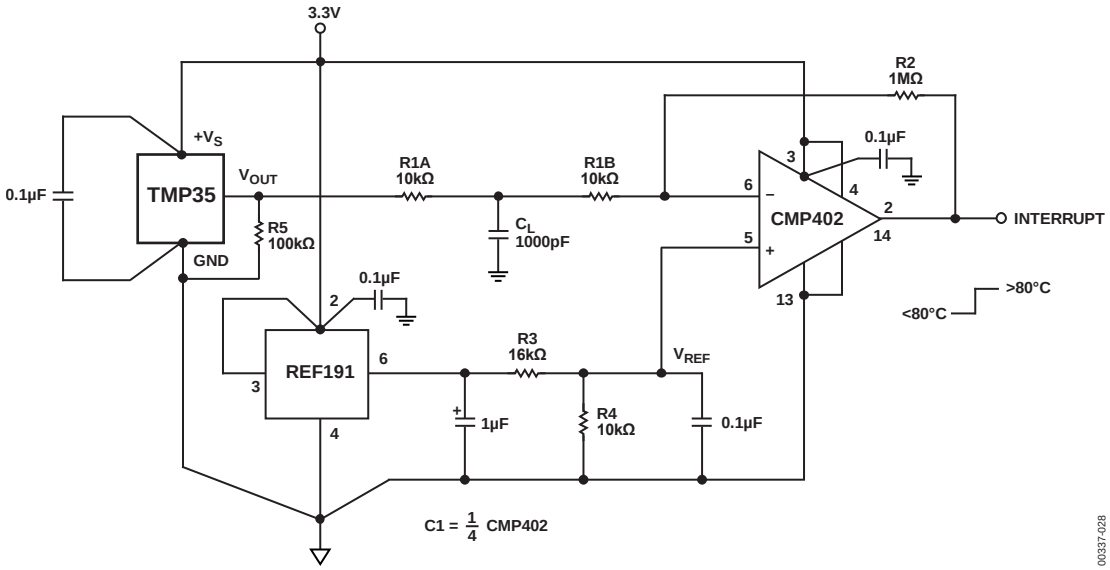


Figure 30. Microprocessor Overtemperature Interrupt Generator

THERMOCOUPLE SIGNAL CONDITIONING WITH COLD-JUNCTION COMPENSATION

The circuit in Figure 31 conditions the output of a Type K thermocouple, while providing cold-junction compensation for temperatures between 0°C and 250°C. The circuit operates from a single 3.3 V to 5.5 V supply and is designed to produce an output voltage transfer characteristic of 10 mV/°C.

A Type K thermocouple exhibits a Seebeck coefficient of approximately 41 µV/°C; therefore, at the cold junction, the TMP35, with a temperature coefficient of 10 mV/°C, is used with R1 and R2 to introduce an opposing cold-junction temperature coefficient of −41 µV/°C. This prevents the isothermal, cold-junction connection between the PCB tracks of the circuit and the wires of the thermocouple from introducing an error in the measured temperature. This compensation works extremely well for circuit ambient temperatures in the range of 20°C to 50°C. Over a 250°C measurement temperature range, the thermocouple produces an output voltage change of 10.151 mV. Because the required output full-scale voltage of the circuit is 2.5 V, the gain of the circuit is set to 246.3. Choosing R4 equal to 4.99 kΩ sets R5 equal to 1.22 MΩ. Because the closest 1% value for R5 is 1.21 MΩ, a 50 kΩ potentiometer is used with R5 for fine trim of the full-scale output voltage. Although the OP193 is a superior single-supply, micropower operational amplifier, its output stage is not rail-to-rail; therefore, the 0°C output voltage level is 0.1 V. If this circuit is digitized by a single-supply ADC, the ADC common should be adjusted to 0.1 V accordingly.

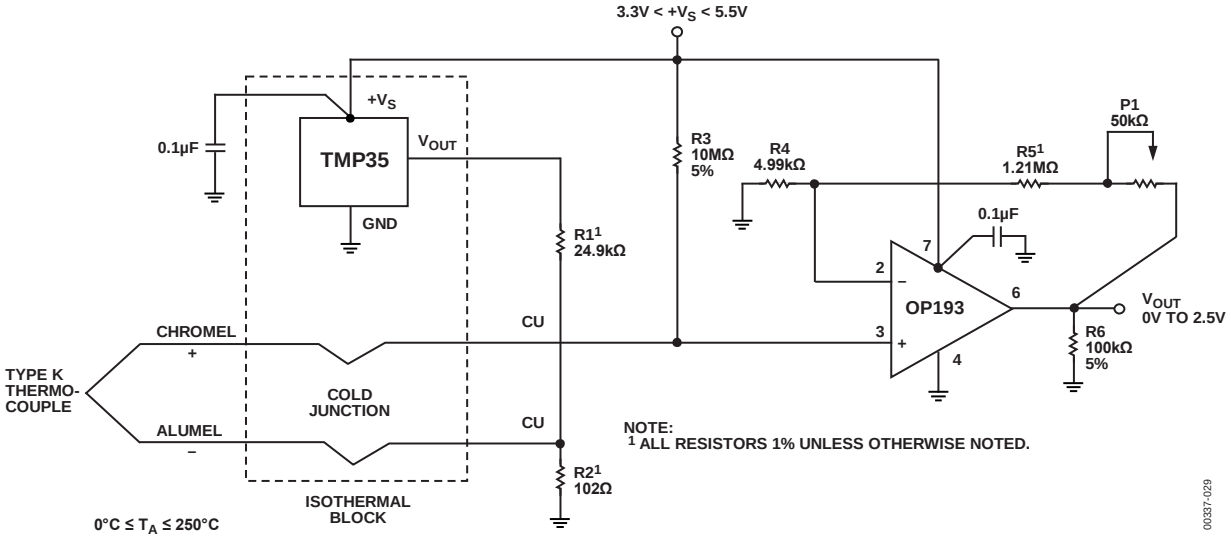
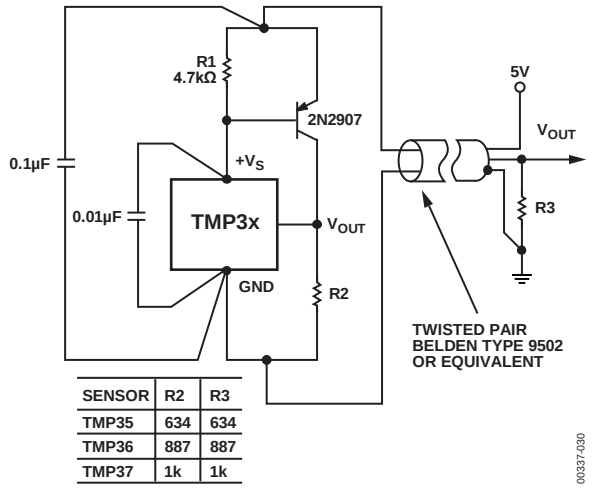


Figure 31. Single-Supply, Type K Thermocouple Signal Conditioning Circuit with Cold-Junction Compensation

USING TMP35/TMP36/TMP37 SENSORS IN REMOTE LOCATIONS

In many industrial environments, sensors are required to operate in the presence of high ambient noise. These noise sources take many forms, for example, SCR transients, relays, radio transmitters, arc welders, and ac motors. They can also be used at considerable distances from the signal conditioning circuitry. These high noise environments are typically in the form of electric fields, so the voltage output of the temperature sensor can be susceptible to contamination from these noise sources.

Figure 32 illustrates a way to convert the output voltage of a TMP35/TMP36/TMP37 sensor into a current to be transmitted down a long twisted pair shielded cable to a ground referenced receiver. The temperature sensors are not capable of high output current operation; thus, a standard PNP transistor is used to boost the output current drive of the circuit. As shown in the table in Figure 32, the values of R2 and R3 were chosen to produce an arbitrary full-scale output current of 2 mA. Lower values for the full-scale current are not recommended. The minimum-scale output current produced by the circuit could be contaminated by ambient magnetic fields operating in the near vicinity of the circuit/cable pair. Because the circuit uses an external transistor, the minimum recommended operating voltage for this circuit is 5 V. To minimize the effects of EMI (or RFI), both the circuit and the temperature sensor supply pins are bypassed with good quality ceramic capacitors.



SENSOR	R2	R3
TMP35	634	634
TMP36	887	887
TMP37	1k	1k

Figure 32. Remote, 2-Wire Boosted Output Current Temperature Sensor

TEMPERATURE TO 4–20 mA LOOP TRANSMITTER

In many process control applications, 2-wire transmitters are used to convey analog signals through noisy ambient environments. These current transmitters use a zero-scale signal current of 4 mA, which can be used to power the signal conditioning circuitry of the transmitter. The full-scale output signal in these transmitters is 20 mA.

Figure 33 illustrates a circuit that transmits temperature information in this fashion. Using a TMP35/TMP36/TMP37 as the temperature sensor, the output current is linearly proportional to the temperature of the medium. The entire circuit operates from the 3 V output of the REF193. The REF193 requires no external trimming because of its tight initial output voltage tolerance and the low supply current of the TMP35/TMP36/TMP37, the OP193, and the REF193. The entire circuit consumes less than 3 mA from a total budget of 4 mA. The OP193 regulates the output current to satisfy the current summation at the noninverting node of the OP193. A generalized expression for the KCL equation at Pin 3 of the OP193 is given by

$$I_{OUT} = \left(\frac{1}{R7}\right) \times \left(\frac{TMP3x \times R3}{R1} + \frac{V_{REF} \times R3}{R2}\right)$$

For each temperature sensor, Table 5 provides the values for the components P1, P2, and R1 to R4.

Table 5. Circuit Element Values for Loop Transmitter

Sensor	R1	P1	R2	P2	R3	R4
TMP35	97.6 kΩ	5 kΩ	1.58 MΩ	100 kΩ	140 kΩ	56.2 kΩ
TMP36	97.6 kΩ	5 kΩ	931 kΩ	50 kΩ	97.6 kΩ	47 kΩ
TMP37	97.6 kΩ	5 kΩ	10.5 kΩ	500 Ω	84.5 kΩ	8.45 kΩ

The 4 mA offset trim is provided by P2, and P1 provides the full-scale gain trim of the circuit at 20 mA. These two trims do not interact because the noninverting input of the OP193 is held at a virtual ground. The zero-scale and full-scale output currents of the circuit are adjusted according to the operating temperature range of each temperature sensor. The Schottky diode, D1, is required in this circuit to prevent loop supply power-on transients from pulling the noninverting input of the OP193 more than 300 mV below its inverting input. Without this diode, such transients can cause phase reversal of the operational amplifier and possible latch-up of the transmitter. The loop supply voltage compliance of the circuit is limited by the maximum applied input voltage to the REF193; it is from 9 V to 18 V.

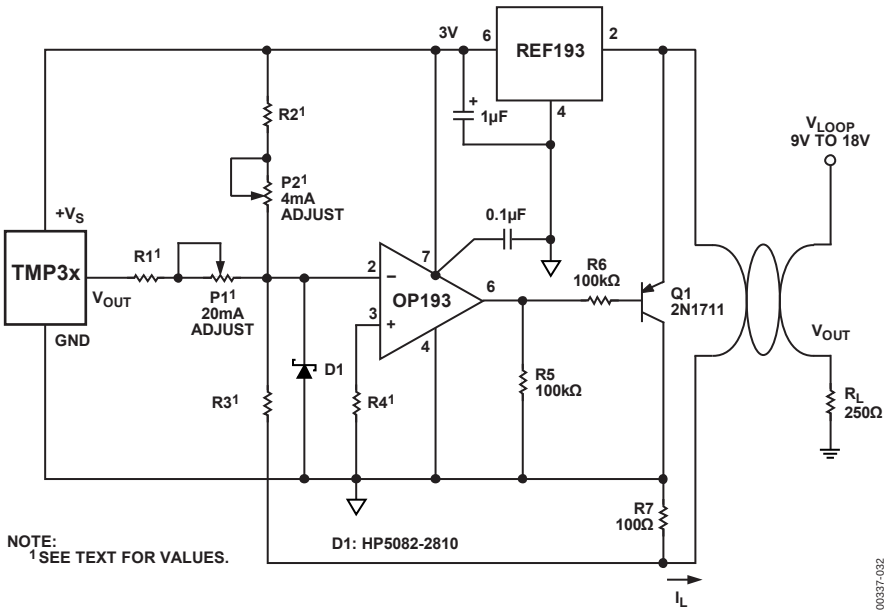


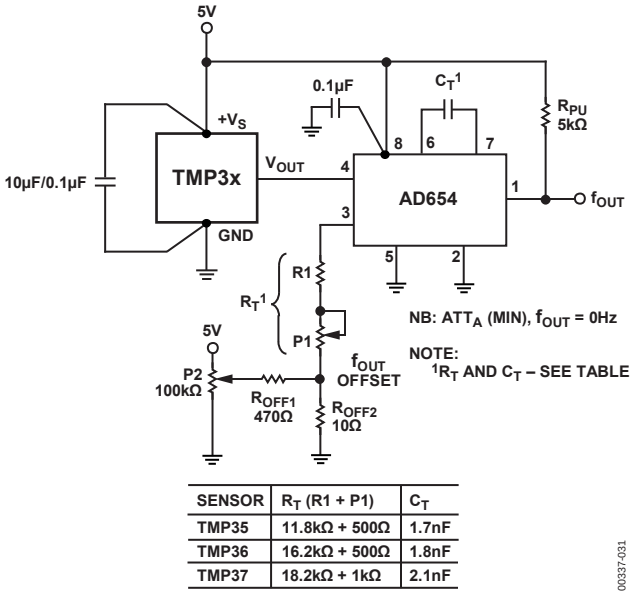
Figure 33. Temperature to 4–20 mA Loop Transmitter

TEMPERATURE-TO-FREQUENCY CONVERTER

Another common method of transmitting analog information from a remote location is to convert a voltage to an equivalent value in the frequency domain. This is readily done with any of the low cost, monolithic voltage-to-frequency converters (VFCs) available. These VFCs feature a robust, open-collector output transistor for easy interfacing to digital circuitry. The digital signal produced by the VFC is less susceptible to contamination from external noise sources and line voltage drops because the only important information is the frequency of the digital signal. When the conversions between temperature and frequency are done accurately, the temperature data from the sensors can be reliably transmitted.

The circuit in Figure 34 illustrates a method by which the outputs of these temperature sensors can be converted to a frequency using the AD654. The output signal of the AD654 is a square wave that is proportional to the dc input voltage across Pin 4 and Pin 3. The transfer equation of the circuit is given by

$$f_{OUT} = \left(\frac{V_{TPM} - V_{OFFSET}}{10 \times (R_T \times C_T)} \right)$$



SENSOR	R_T (R1 + P1)	C_T
TMP35	11.8kΩ + 500Ω	1.7nF
TMP36	16.2kΩ + 500Ω	1.8nF
TMP37	18.2kΩ + 1kΩ	2.1nF

Figure 34. Temperature-to-Frequency Converter

An offset trim network (f_{OUT} OFFSET) is included with this circuit to set f_{OUT} to 0 Hz when the minimum output voltage of the temperature sensor is reached. Potentiometer P1 is required to calibrate the absolute accuracy of the AD654. The table in Figure 34 illustrates the circuit element values for each of the three sensors. The nominal offset voltage required for 0 Hz output from the TMP35 is 50 mV; for the TMP36 and TMP37, the offset voltage required is 100 mV. For the circuit values shown, the output frequency transfer characteristic of the circuit was set at 50 Hz/°C in all cases. At the receiving end, a frequency-to-voltage converter (FVC) can be used to convert the frequency back to a dc voltage for further processing. One such FVC is the AD650.

For complete information about the AD650 and the AD654, consult the individual data sheets for those devices.

DRIVING LONG CABLES OR HEAVY CAPACITIVE LOADS

Although the TMP35/TMP36/TMP37 temperature sensors can drive capacitive loads up to 10,000 pF without oscillation, output voltage transient response times can be improved by using a small resistor in series with the output of the temperature sensor, as shown in Figure 35. As an added benefit, this resistor forms a low-pass filter with the cable capacitance, which helps to reduce bandwidth noise. Because the temperature sensor is likely to be used in environments where the ambient noise level can be very high, this resistor helps to prevent rectification by the devices of the high frequency noise. The combination of this resistor and the supply bypass capacitor offers the best protection.

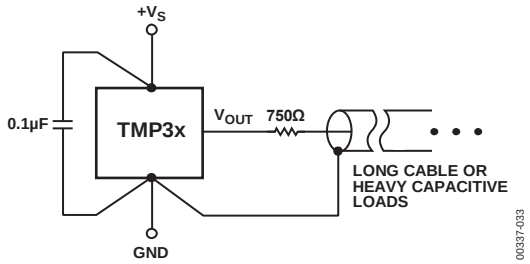


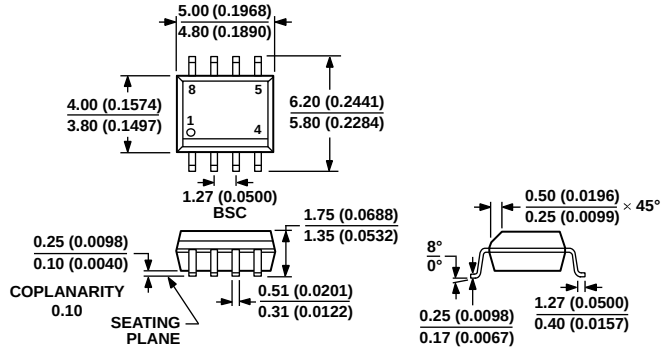
Figure 35. Driving Long Cables or Heavy Capacitive Loads

COMMENTARY ON LONG-TERM STABILITY

The concept of long-term stability has been used for many years to describe the amount of parameter shift that occurs during the lifetime of an IC. This is a concept that has been typically applied to both voltage references and monolithic temperature sensors. Unfortunately, integrated circuits cannot be evaluated at room temperature (25°C) for 10 years or more to determine this shift. As a result, manufacturers very typically perform accelerated lifetime testing of integrated circuits by operating ICs at elevated temperatures (between 125°C and 150°C) over a shorter period of time (typically, between 500 and 1000 hours).

As a result of this operation, the lifetime of an integrated circuit is significantly accelerated due to the increase in rates of reaction within the semiconductor material.

OUTLINE DIMENSIONS

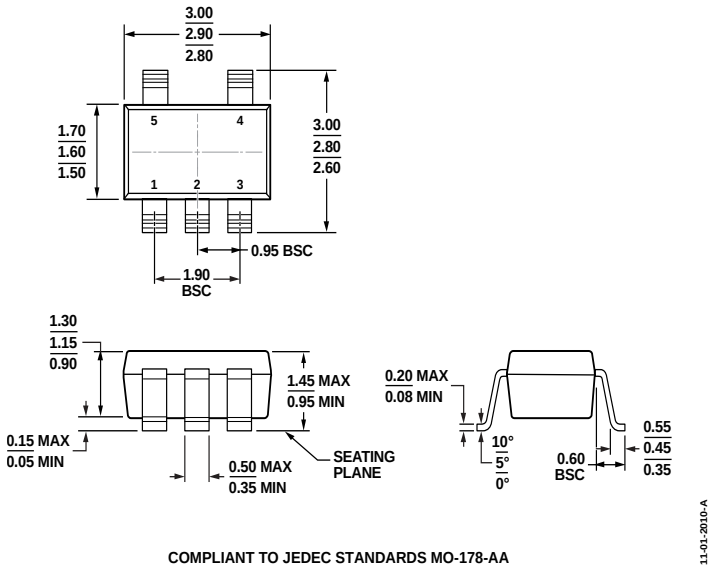


COMPLIANT TO JEDEC STANDARDS MS-012-AA

CONTROLLING DIMENSIONS ARE IN MILLIMETERS; INCH DIMENSIONS (IN PARENTHESES) ARE ROUNDED-OFF MILLIMETER EQUIVALENTS FOR REFERENCE ONLY AND ARE NOT APPROPRIATE FOR USE IN DESIGN.

012407-A

Figure 36. 8-Lead Standard Small Outline Package [SOIC_N]
Narrow Body
(R-8)
Dimensions shown in millimeters and (inches)



COMPLIANT TO JEDEC STANDARDS MO-178-AA

11-01-2010-A

Figure 37. 5-Lead Small Outline Transistor Package [SOT-23]
(RJ-5)
Dimensions shown in millimeters

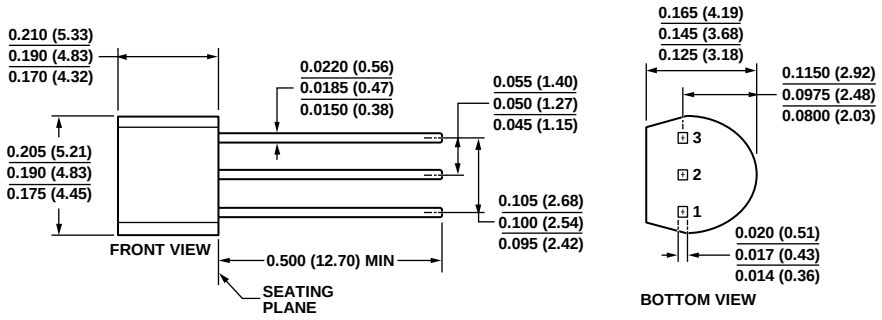


Figure 38. 3-Pin Plastic Header-Style Package [TO-92]
(T-3-1)
Dimensions shown in inches and (millimeters)

ORDERING GUIDE

Model[1, 2]	Accuracy at 25°C (°C max)	Linear Operating Temperature Range	Package Description	Package Option	Branding
TMP35FSZ-REEL	±2.0	10°C to 125°C	8-Lead Standard Small Outline Package (SOIC_N)	R-8	
TMP35GRTZ-REEL7	±3.0	10°C to 125°C	5-Lead Small Outline Transistor Package (SOT-23)	RJ-5	#T11
TMP35GT9Z	±3.0	10°C to 125°C	3-Pin Plastic Header-Style Package (TO-92)	T-3-1	
ADW75001Z-0REEL7	±3.0	−40°C to +125°C	5-Lead Small Outline Transistor Package (SOT-23)	RJ-5	#T6G
TMP36FS	±2.0	−40°C to +125°C	8-Lead Standard Small Outline Package (SOIC_N)	R-8	
TMP36FS-REEL	±2.0	−40°C to +125°C	8-Lead Standard Small Outline Package (SOIC_N)	R-8	
TMP36FSZ	±2.0	−40°C to +125°C	8-Lead Standard Small Outline Package (SOIC_N)	R-8	
TMP36FSZ-REEL	±2.0	−40°C to +125°C	8-Lead Standard Small Outline Package (SOIC_N)	R-8	
TMP36GRT-REEL7	±3.0	−40°C to +125°C	5-Lead Small Outline Transistor Package (SOT-23)	RJ-5	T6G
TMP36GRTZ-REEL7	±3.0	−40°C to +125°C	5-Lead Small Outline Transistor Package (SOT-23)	RJ-5	#T6G
TMP36GSZ	±3.0	−40°C to +125°C	8-Lead Standard Small Outline Package (SOIC_N)	R-8	
TMP36GSZ-REEL	±3.0	−40°C to +125°C	8-Lead Standard Small Outline Package (SOIC_N)	R-8	
TMP36GSZ-REEL7	±3.0	−40°C to +125°C	8-Lead Standard Small Outline Package (SOIC_N)	R-8	
TMP36GT9	±3.0	−40°C to +125°C	3-Pin Plastic Header-Style Package (TO-92)	T-3-1	
TMP36GT9Z	±3.0	−40°C to +125°C	3-Pin Plastic Header-Style Package (TO-92)	T-3-1	
TMP36-PT7		−40°C to +125°C	Chips or Die		
TMP37FT9Z	±2.0	5°C to 100°C	3-Pin Plastic Header-Style Package (TO-92)	T-3-1	
TMP37GRTZ-REEL7	±3.0	5°C to 100°C	5-Lead Small Outline Transistor Package (SOT-23)	RJ-5	#T12

[1] Z = RoHS Compliant Part.
[2] W = Qualified for Automotive Applications.

AUTOMOTIVE PRODUCTS

The ADW75001Z-0REEL7 model is available with controlled manufacturing to support the quality and reliability requirements of automotive applications. Note that this automotive model may have specifications that differ from the commercial models; therefore, designers should review the Specifications section of this data sheet carefully. Only automotive grade products shown are available for use in automotive applications. Contact your local Analog Devices account representative for specific product ordering information and to obtain the specific Automotive Reliability reports for these models.

©1996–2015 Analog Devices, Inc. All rights reserved. Trademarks and registered trademarks are the property of their respective owners.
D00337-0-5/15(H)

P2N2222A

Amplifier Transistors

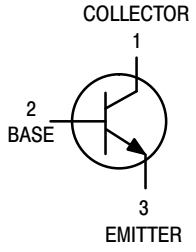
NPN Silicon

ON Semiconductor®

http://onsemi.com

Features

- These are Pb−Free Devices*



MAXIMUM RATINGS (T_A = 25°C unless otherwise noted)

Characteristic	Symbol	Value	Unit
Collector − Emitter Voltage	V_{CEO}	40	Vdc
Collector − Base Voltage	V_{CBO}	75	Vdc
Emitter − Base Voltage	V_{EBO}	6.0	Vdc
Collector Current − Continuous	I_C	600	mAdc
Total Device Dissipation @ T_A = 25°C Derate above 25°C	P_D	625 5.0	mW mW/°C
Total Device Dissipation @ T_C = 25°C Derate above 25°C	P_D	1.5 12	W mW/°C
Operating and Storage Junction Temperature Range	T_J, T_{stg}	−55 to +150	°C

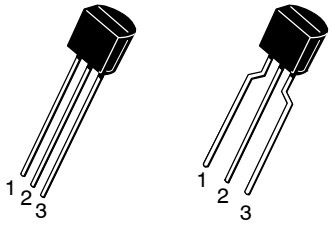
THERMAL CHARACTERISTICS

Characteristic	Symbol	Max	Unit
Thermal Resistance, Junction to Ambient	$R_{\theta JA}$	200	°C/W
Thermal Resistance, Junction to Case	$R_{\theta JC}$	83.3	°C/W

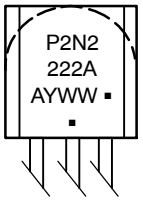
Stresses exceeding Maximum Ratings may damage the device. Maximum Ratings are stress ratings only. Functional operation above the Recommended Operating Conditions is not implied. Extended exposure to stresses above the Recommended Operating Conditions may affect device reliability.

*For additional information on our Pb−Free strategy and soldering details, please download the ON Semiconductor Soldering and Mounting Techniques Reference Manual, SOLDERRM/D.

TO−92
CASE 29
STYLE 17



MARKING DIAGRAM



A = Assembly Location
Y = Year
WW = Work Week
▪ = Pb−Free Package
(Note: Microdot may be in either location)

ORDERING INFORMATION

Device	Package	Shipping†
P2N2222AG	TO−92 (Pb−Free)	5000 Units/Bulk
P2N2222ARL1G	TO−92 (Pb−Free)	2000/Tape & Ammo

†For information on tape and reel specifications, including part orientation and tape sizes, please refer to our Tape and Reel Packaging Specification Brochure, BRD8011/D.

© Semiconductor Components Industries, LLC, 2013

ELECTRICAL CHARACTERISTICS (T_A = 25°C unless otherwise noted)

Characteristic	Symbol	Min	Max	Unit
OFF CHARACTERISTICS				
Collector – Emitter Breakdown Voltage (I_C = 10 mAdc, I_B = 0)	$V_{(BR)CEO}$	40	–	Vdc
Collector – Base Breakdown Voltage (I_C = 10 μAdc, I_E = 0)	$V_{(BR)CBO}$	75	–	Vdc
Emitter – Base Breakdown Voltage (I_E = 10 μAdc, I_C = 0)	$V_{(BR)EBO}$	6.0	–	Vdc
Collector Cutoff Current (V_{CE} = 60 Vdc, $V_{EB(off)}$ = 3.0 Vdc)	I_{CEX}	–	10	nAdc
Collector Cutoff Current (V_{CB} = 60 Vdc, I_E = 0) (V_{CB} = 60 Vdc, I_E = 0, T_A = 150°C)	I_{CBO}	– –	0.01 10	μAdc
Emitter Cutoff Current (V_{EB} = 3.0 Vdc, I_C = 0)	I_{EBO}	–	10	nAdc
Collector Cutoff Current (V_{CE} = 10 V)	I_{CEO}	–	10	nAdc
Base Cutoff Current (V_{CE} = 60 Vdc, $V_{EB(off)}$ = 3.0 Vdc)	I_{BEX}	–	20	nAdc
ON CHARACTERISTICS				
DC Current Gain (I_C = 0.1 mAdc, V_{CE} = 10 Vdc) (I_C = 1.0 mAdc, V_{CE} = 10 Vdc) (I_C = 10 mAdc, V_{CE} = 10 Vdc) (I_C = 10 mAdc, V_{CE} = 10 Vdc, T_A = −55°C) (I_C = 150 mAdc, V_{CE} = 10 Vdc) (Note 1) (I_C = 150 mAdc, V_{CE} = 1.0 Vdc) (Note 1) (I_C = 500 mAdc, V_{CE} = 10 Vdc) (Note 1)	h_{FE}	 35 50 75 35 100 50 40	 – – – – 300 – –	–
Collector – Emitter Saturation Voltage (Note 1) (I_C = 150 mAdc, I_B = 15 mAdc) (I_C = 500 mAdc, I_B = 50 mAdc)	$V_{CE(sat)}$	 – –	 0.3 1.0	Vdc
Base – Emitter Saturation Voltage (Note 1) (I_C = 150 mAdc, I_B = 15 mAdc) (I_C = 500 mAdc, I_B = 50 mAdc)	$V_{BE(sat)}$	 0.6 –	 1.2 2.0	Vdc
SMALL–SIGNAL CHARACTERISTICS				
Current – Gain – Bandwidth Product (Note 2) (I_C = 20 mAdc, V_{CE} = 20 Vdc, f = 100 MHz)C	f_T	300	–	MHz
Output Capacitance (V_{CB} = 10 Vdc, I_E = 0, f = 1.0 MHz)	C_{obo}	–	8.0	pF
Input Capacitance (V_{EB} = 0.5 Vdc, I_C = 0, f = 1.0 MHz)	C_{ibo}	–	25	pF
Input Impedance (I_C = 1.0 mAdc, V_{CE} = 10 Vdc, f = 1.0 kHz) (I_C = 10 mAdc, V_{CE} = 10 Vdc, f = 1.0 kHz)	h_{ie}	 2.0 0.25	 8.0 1.25	kΩ
Voltage Feedback Ratio (I_C = 1.0 mAdc, V_{CE} = 10 Vdc, f = 1.0 kHz) (I_C = 10 mAdc, V_{CE} = 10 Vdc, f = 1.0 kHz)	h_{re}	 – –	 8.0 4.0	X 10^{-4}
Small–Signal Current Gain (I_C = 1.0 mAdc, V_{CE} = 10 Vdc, f = 1.0 kHz) (I_C = 10 mAdc, V_{CE} = 10 Vdc, f = 1.0 kHz)	h_{fe}	 50 75	 300 375	–
Output Admittance (I_C = 1.0 mAdc, V_{CE} = 10 Vdc, f = 1.0 kHz) (I_C = 10 mAdc, V_{CE} = 10 Vdc, f = 1.0 kHz)	h_{oe}	 5.0 25	 35 200	μMhos
Collector Base Time Constant (I_E = 20 mAdc, V_{CB} = 20 Vdc, f = 31.8 MHz)	$rb'C_c$	–	150	ps
Noise Figure (I_C = 100 μAdc, V_{CE} = 10 Vdc, R_S = 1.0 kΩ, f = 1.0 kHz)	N_F	–	4.0	dB

1. Pulse Test: Pulse Width ≤ 300 μs, Duty Cycle ≤ 2.0%.
2. f_T is defined as the frequency at which $|h_{fe}|$ extrapolates to unity.

ELECTRICAL CHARACTERISTICS (T_A = 25°C unless otherwise noted) (Continued)

Characteristic		Symbol	Min	Max	Unit
SWITCHING CHARACTERISTICS					
Delay Time	(V_{CC} = 30 Vdc, $V_{BE(off)}$ = −2.0 Vdc, I_C = 150 mAdc, I_{B1} = 15 mAdc) (Figure 1)	t_d	–	10	ns
Rise Time		t_r	–	25	ns
Storage Time	(V_{CC} = 30 Vdc, I_C = 150 mAdc, I_{B1} = I_{B2} = 15 mAdc) (Figure 2)	t_s	–	225	ns
Fall Time		t_f	–	60	ns

SWITCHING TIME EQUIVALENT TEST CIRCUITS

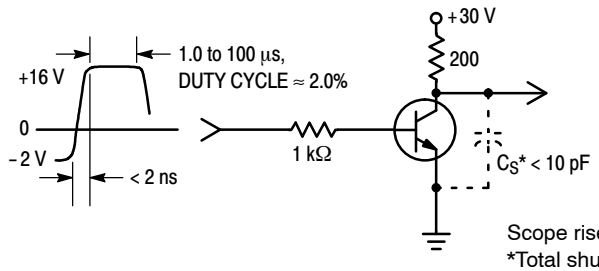
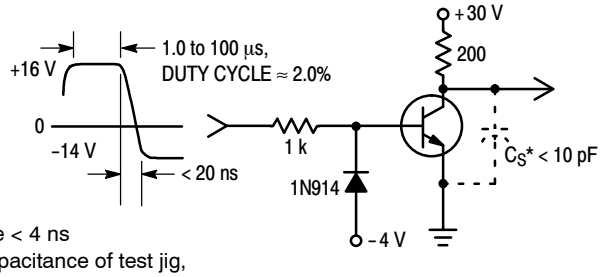


Figure 1. Turn−On Time



Scope rise time < 4 ns
*Total shunt capacitance of test jig, connectors, and oscilloscope.

Figure 2. Turn−Off Time

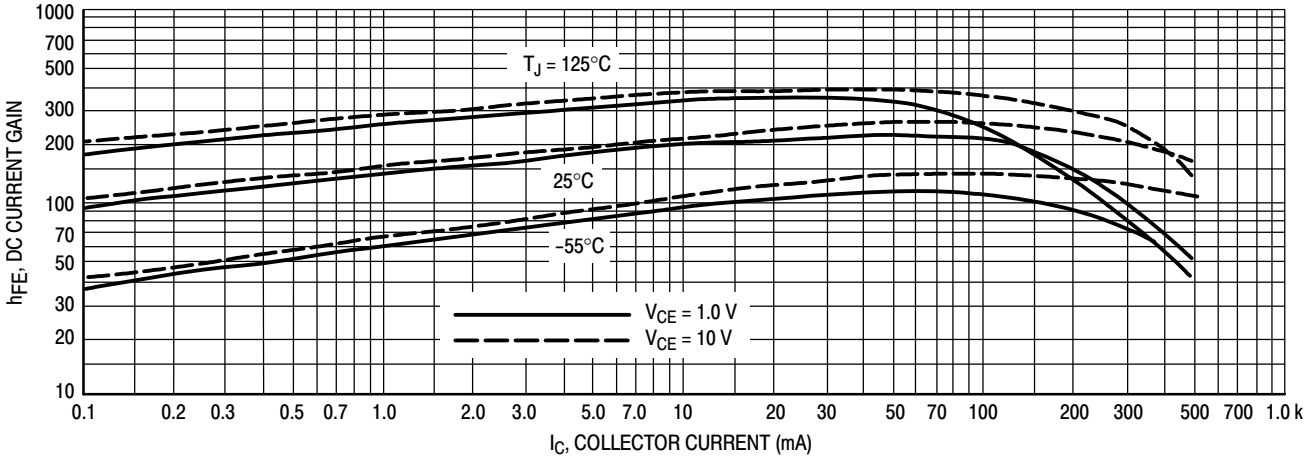


Figure 3. DC Current Gain

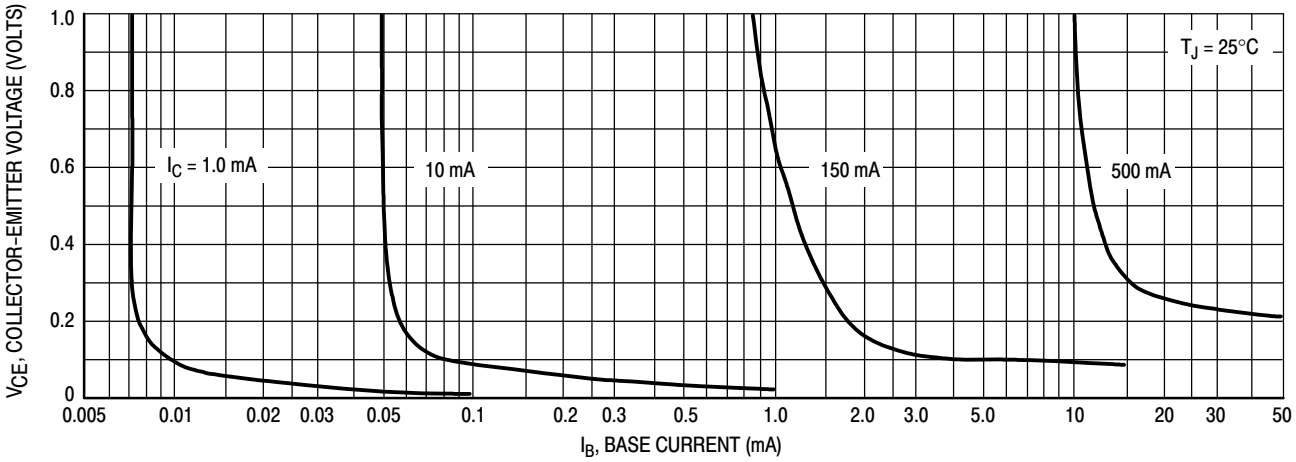


Figure 4. Collector Saturation Region

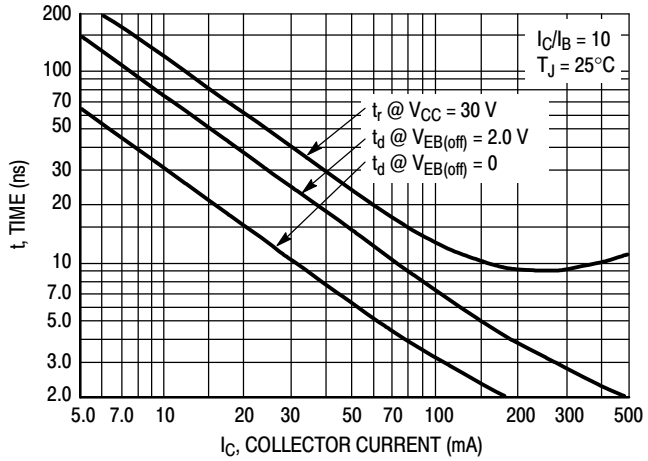


Figure 5. Turn – On Time

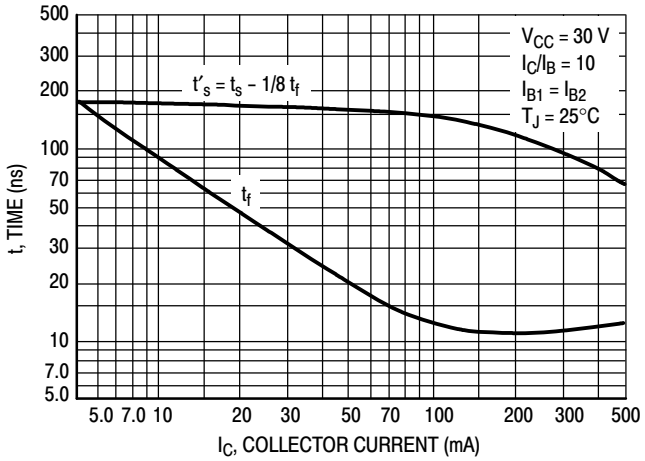


Figure 6. Turn – Off Time

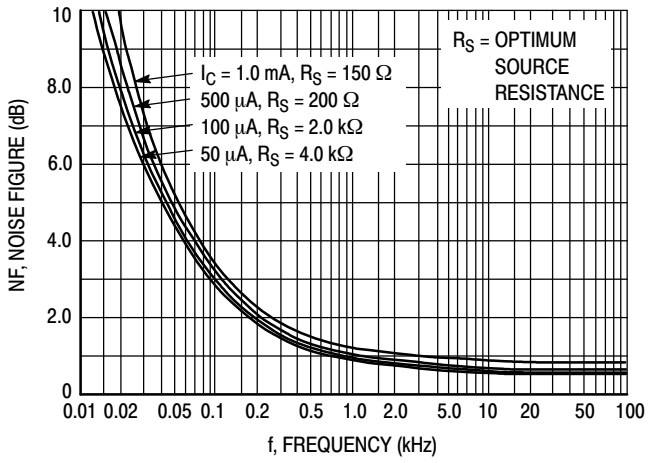


Figure 7. Frequency Effects

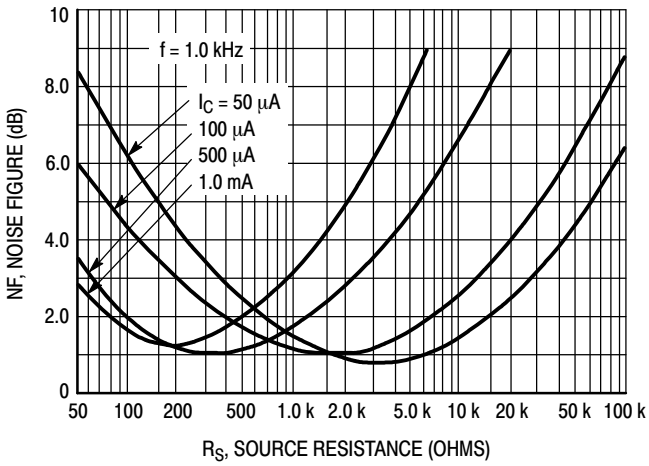


Figure 8. Source Resistance Effects

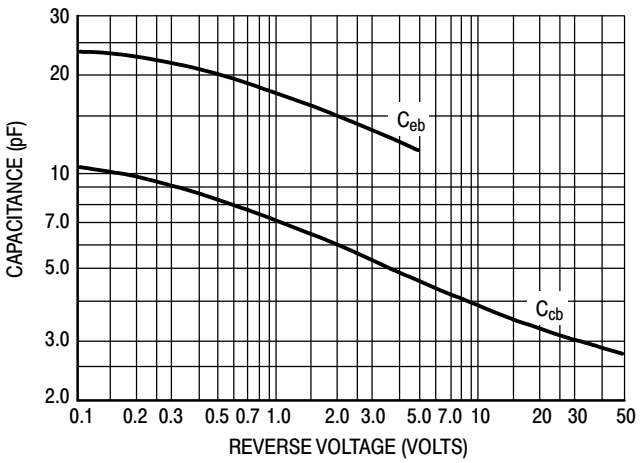


Figure 9. Capacitances

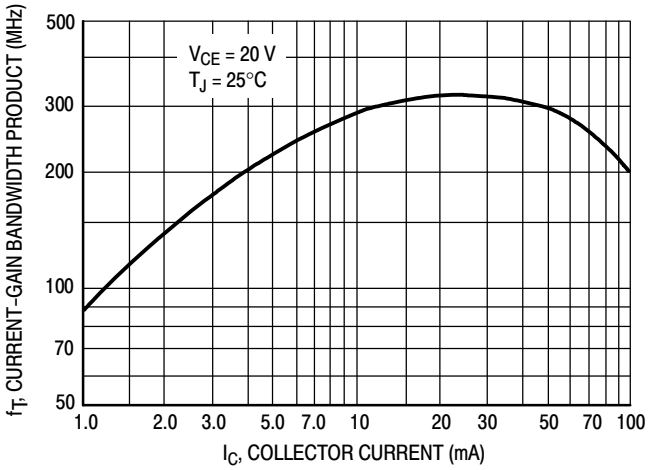


Figure 10. Current−Gain Bandwidth Product

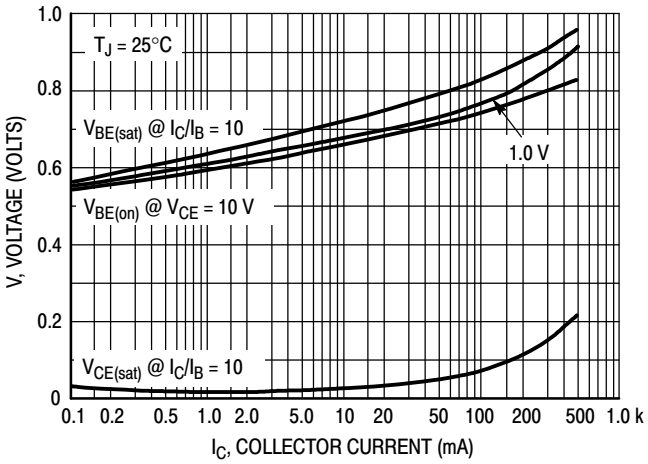


Figure 11. “On” Voltages

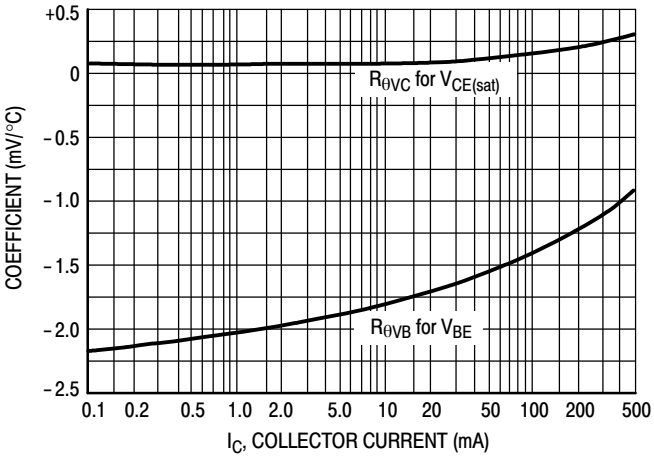


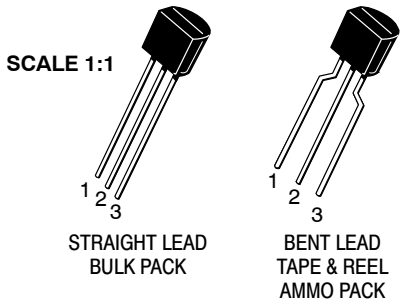
Figure 12. Temperature Coefficients

MECHANICAL CASE OUTLINE
PACKAGE DIMENSIONS

ON Semiconductor®

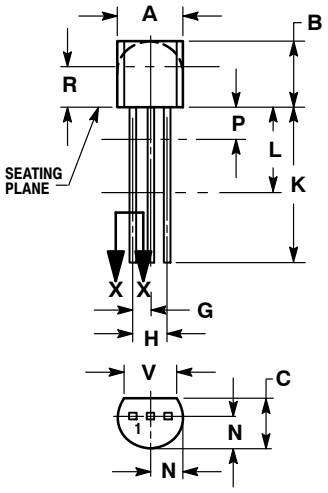
TO–92 (TO–226)
CASE 29–11
ISSUE AM

DATE 09 MAR 2007

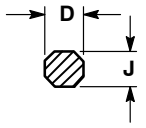


STRAIGHT LEAD
BULK PACK

BENT LEAD
TAPE & REEL
AMMO PACK



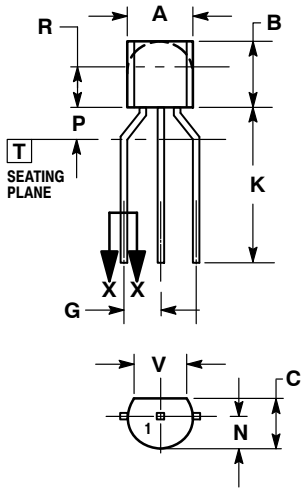
STRAIGHT LEAD
BULK PACK



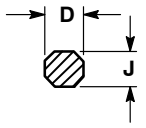
SECTION X–X

NOTES:
1. DIMENSIONING AND TOLERANCING PER ANSI Y14.5M, 1982.
2. CONTROLLING DIMENSION: INCH.
3. CONTOUR OF PACKAGE BEYOND DIMENSION R IS UNCONTROLLED.
4. LEAD DIMENSION IS UNCONTROLLED IN P AND BEYOND DIMENSION K MINIMUM.

	INCHES		MILLIMETERS	
DIM	MIN	MAX	MIN	MAX
A	0.175	0.205	4.45	5.20
B	0.170	0.210	4.32	5.33
C	0.125	0.165	3.18	4.19
D	0.016	0.021	0.407	0.533
G	0.045	0.055	1.15	1.39
H	0.095	0.105	2.42	2.66
J	0.015	0.020	0.39	0.50
K	0.500	---	12.70	---
L	0.250	---	6.35	---
N	0.080	0.105	2.04	2.66
P	---	0.100	---	2.54
R	0.115	---	2.93	---
V	0.135	---	3.43	---



BENT LEAD
TAPE & REEL
AMMO PACK



SECTION X–X

NOTES:
1. DIMENSIONING AND TOLERANCING PER ASME Y14.5M, 1994.
2. CONTROLLING DIMENSION: MILLIMETERS.
3. CONTOUR OF PACKAGE BEYOND DIMENSION R IS UNCONTROLLED.
4. LEAD DIMENSION IS UNCONTROLLED IN P AND BEYOND DIMENSION K MINIMUM.

	MILLIMETERS	
DIM	MIN	MAX
A	4.45	5.20
B	4.32	5.33
C	3.18	4.19
D	0.40	0.54
G	2.40	2.80
J	0.39	0.50
K	12.70	---
N	2.04	2.66
P	1.50	4.00
R	2.93	---
V	3.43	---

STYLES ON PAGE 2

DOCUMENT NUMBER:	**98ASB42022B**	Electronic versions are uncontrolled except when accessed directly from the Document Repository. Printed versions are uncontrolled except when stamped "CONTROLLED COPY" in red.
STATUS:	**ON SEMICONDUCTOR STANDARD**	
NEW STANDARD:		
DESCRIPTION:	**TO–92 (TO–226)**	**PAGE 1 OF 3**

TO−92 (TO−226)
CASE 29−11
ISSUE AM

DATE 09 MAR 2007

STYLE 1:
PIN 1. EMITTER
2. BASE
3. COLLECTOR

STYLE 2:
PIN 1. BASE
2. EMITTER
3. COLLECTOR

STYLE 3:
PIN 1. ANODE
2. ANODE
3. CATHODE

STYLE 4:
PIN 1. CATHODE
2. CATHODE
3. ANODE

STYLE 5:
PIN 1. DRAIN
2. SOURCE
3. GATE

STYLE 6:
PIN 1. GATE
2. SOURCE & SUBSTRATE
3. DRAIN

STYLE 7:
PIN 1. SOURCE
2. DRAIN
3. GATE

STYLE 8:
PIN 1. DRAIN
2. GATE
3. SOURCE & SUBSTRATE

STYLE 9:
PIN 1. BASE 1
2. EMITTER
3. BASE 2

STYLE 10:
PIN 1. CATHODE
2. GATE
3. ANODE

STYLE 11:
PIN 1. ANODE
2. CATHODE & ANODE
3. CATHODE

STYLE 12:
PIN 1. MAIN TERMINAL 1
2. GATE
3. MAIN TERMINAL 2

STYLE 13:
PIN 1. ANODE 1
2. GATE
3. CATHODE 2

STYLE 14:
PIN 1. EMITTER
2. COLLECTOR
3. BASE

STYLE 15:
PIN 1. ANODE 1
2. CATHODE
3. ANODE 2

STYLE 16:
PIN 1. ANODE
2. GATE
3. CATHODE

STYLE 17:
PIN 1. COLLECTOR
2. BASE
3. EMITTER

STYLE 18:
PIN 1. ANODE
2. CATHODE
3. NOT CONNECTED

STYLE 19:
PIN 1. GATE
2. ANODE
3. CATHODE

STYLE 20:
PIN 1. NOT CONNECTED
2. CATHODE
3. ANODE

STYLE 21:
PIN 1. COLLECTOR
2. EMITTER
3. BASE

STYLE 22:
PIN 1. SOURCE
2. GATE
3. DRAIN

STYLE 23:
PIN 1. GATE
2. SOURCE
3. DRAIN

STYLE 24:
PIN 1. EMITTER
2. COLLECTOR/ANODE
3. CATHODE

STYLE 25:
PIN 1. MT 1
2. GATE
3. MT 2

STYLE 26:
PIN 1. V_{CC}
2. GROUND 2
3. OUTPUT

STYLE 27:
PIN 1. MT
2. SUBSTRATE
3. MT

STYLE 28:
PIN 1. CATHODE
2. ANODE
3. GATE

STYLE 29:
PIN 1. NOT CONNECTED
2. ANODE
3. CATHODE

STYLE 30:
PIN 1. DRAIN
2. GATE
3. SOURCE

STYLE 31:
PIN 1. GATE
2. DRAIN
3. SOURCE

STYLE 32:
PIN 1. BASE
2. COLLECTOR
3. EMITTER

STYLE 33:
PIN 1. RETURN
2. INPUT
3. OUTPUT

STYLE 34:
PIN 1. INPUT
2. GROUND
3. LOGIC

STYLE 35:
PIN 1. GATE
2. COLLECTOR
3. EMITTER

ISSUE	REVISION	DATE
AM	ADDED BENT−LEAD TAPE & REEL VERSION. REQ. BY J. SUPINA.	09 MAR 2007

ON Semiconductor and ON are registered trademarks of Semiconductor Components Industries, LLC (SCILLC). SCILLC reserves the right to make changes without further notice to any products herein. SCILLC makes no warranty, representation or guarantee regarding the suitability of its products for any particular purpose, nor does SCILLC assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation special, consequential or incidental damages. "Typical" parameters which may be provided in SCILLC data sheets and/or specifications can and do vary in different applications and actual performance may vary over time. All operating parameters, including "Typicals" must be validated for each customer application by customer's technical experts. SCILLC does not convey any license under its patent rights nor the rights of others. SCILLC products are not designed, intended, or authorized for use as components in systems intended for surgical implant into the body, or other applications intended to support or sustain life, or for any other application in which the failure of the SCILLC product could create a situation where personal injury or death may occur. Should Buyer purchase or use SCILLC products for any such unintended or unauthorized application, Buyer shall indemnify and hold SCILLC and its officers, employees, subsidiaries, affiliates, and distributors harmless against all claims, costs, damages, and expenses, and reasonable attorney fees arising out of, directly or indirectly, any claim of personal injury or death associated with such unintended or unauthorized use, even if such claim alleges that SCILLC was negligent regarding the design or manufacture of the part. SCILLC is an Equal Opportunity/Affirmative Action Employer. This literature is subject to all applicable copyright laws and is not for resale in any manner.

onsemi, ONSEMI , and other names, marks, and brands are registered and/or common law trademarks of Semiconductor Components Industries, LLC dba "**onsemi**" or its affiliates and/or subsidiaries in the United States and/or other countries. **onsemi** owns the rights to a number of patents, trademarks, copyrights, trade secrets, and other intellectual property. A listing of **onsemi**'s product/patent coverage may be accessed at www.onsemi.com/site/pdf/Patent−Marking.pdf. **onsemi** reserves the right to make changes at any time to any products or information herein, without notice. The information herein is provided "as−is" and **onsemi** makes no warranty, representation or guarantee regarding the accuracy of the information, product features, availability, functionality, or suitability of its products for any particular purpose, nor does **onsemi** assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation special, consequential or incidental damages. Buyer is responsible for its products and applications using **onsemi** products, including compliance with all laws, regulations and safety requirements or standards, regardless of any support or applications information provided by **onsemi**. "Typical" parameters which may be provided in **onsemi** data sheets and/or specifications can and do vary in different applications and actual performance may vary over time. All operating parameters, including "Typicals" must be validated for each customer application by customer's technical experts. **onsemi** does not convey any license under any of its intellectual property rights nor the rights of others. **onsemi** products are not designed, intended, or authorized for use as a critical component in life support systems or any FDA Class 3 medical devices or medical devices with a same or similar classification in a foreign jurisdiction or any devices intended for implantation in the human body. Should Buyer purchase or use **onsemi** products for any such unintended or unauthorized application, Buyer shall indemnify and hold **onsemi** and its officers, employees, subsidiaries, affiliates, and distributors harmless against all claims, costs, damages, and expenses, and reasonable attorney fees arising out of, directly or indirectly, any claim of personal injury or death associated with such unintended or unauthorized use, even if such claim alleges that **onsemi** was negligent regarding the design or manufacture of the part. **onsemi** is an Equal Opportunity/Affirmative Action Employer. This literature is subject to all applicable copyright laws and is not for resale in any manner.

PUBLICATION ORDERING INFORMATION

LITERATURE FULFILLMENT:
Email Requests to: orderlit@onsemi.com

onsemi Website: www.onsemi.com

TECHNICAL SUPPORT
North American Technical Support:
Voice Mail: 1 800−282−9855 Toll Free USA/Canada
Phone: 011 421 33 790 2910

Europe, Middle East and Africa Technical Support:
Phone: 00421 33 790 2910
For additional information, please contact your local Sales Representative

◊

SONGLE RELAY

松乐继电器 ® SONGLE RELAY	RELAY ISO9002	SRD

1. MAIN FEATURES

- Switching capacity available by 10A in spite of small size design for highdensity P.C. board mounting technique.
- UL,CUL,TUV recognized.
- Selection of plastic material for high temperature and better chemical solution performance.
- Sealed types available.
- Simple relay magnetic circuit to meet low cost of mass production.

2. APPLICATIONS

- Domestic appliance, office machine, audio, equipment, automobile, etc.
  (Remote control TV receiver, monitor display, audio equipment high rushing current use application.)

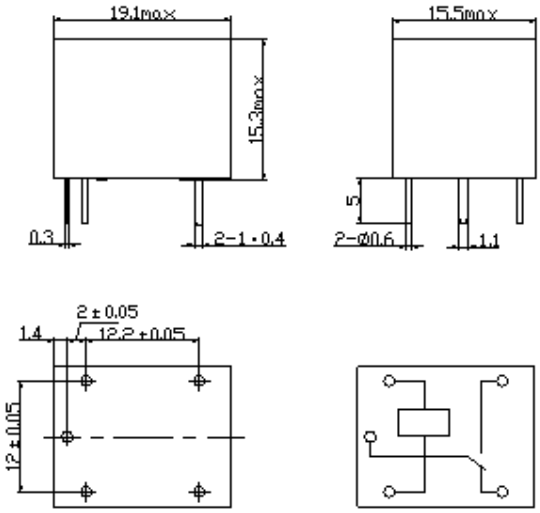
3. ORDERING INFORMATION

SRD	XX VDC	S	L	C
Model of relay	Nominal coil voltage	Structure	Coil sensitivity	Contact form
SRD	03、05、06、09、12、24、48VDC	S:Sealed type	L:0.36W	A:1 form A
				B:1 form B
		F:Flux free type	D:0.45W	C:1 form C

4. RATING

CCC	FILE NUMBER:CH0052885-2000	7A/240VDC
CCC	FILE NUMBER:CH0036746-99	10A/250VDC
UL /CUL	FILE NUMBER: E167996	10A/125VAC 28VDC
TUV	FILE NUMBER: R9933789	10A/240VAC 28VDC

5. DIMENSION(unit:mm) DRILLING(unit:mm) WIRING DIAGRAM



6. COIL DATA CHART (AT20°C)

Coil Sensitivity	Coil Voltage Code	Nominal Voltage (VDC)	Nominal Current (mA)	Coil Resistance (Ω) ±10%	Power Consumption (W)	Pull-In Voltage (VDC)	Drop-Out Voltage (VDC)	Max-Allowable Voltage (VDC)
SRD (High Sensitivity)	03	03	120	25	abt. 0.36W	75%Max.	10% Min.	120%
	05	05	71.4	70				
	06	06	60	100				
	09	09	40	225				
	12	12	30	400				
	24	24	15	1600				
	48	48	7.5	6400				
SRD (Standard)	03	03	150	20	abt. 0.45W	75% Max.	10% Min.	110%
	05	05	89.3	55				
	06	06	75	80				
	09	09	50	180				
	12	12	37.5	320				
	24	24	18.7	1280				
	48	48	10	4500	abt. 0.51W			

7. CONTACT RATING

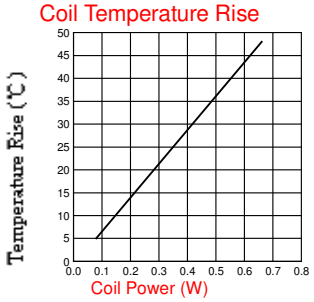
Item \ Type	SRD	
	FORM C	FORM A
Contact Capacity Resistive Load (cosΦ=1)	7A 28VDC 10A 125VAC 7A 240VAC	10A 28VDC 10A 240VAC
Inductive Load (cosΦ=0.4 L/R=7msec)	3A 120VAC 3A 28VDC	5A 120VAC 5A 28VDC
Max. Allowable Voltage	250VAC/110VDC	250VAC/110VDC
Max. Allowable Power Force	800VAC/240W	1200VA/300W
Contact Material	AgCdO	AgCdO

8. PERFORMANCE (at initial value)

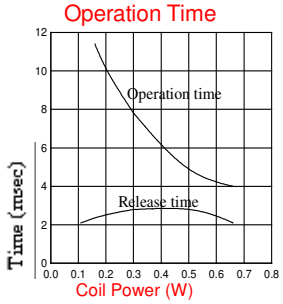
Item \ Type	SRD
Contact Resistance	100mΩ Max.
Operation Time	10msec Max.
Release Time	5msec Max.
Dielectric Strength Between coil & contact Between contacts	 1500VAC 50/60HZ (1 minute) 1000VAC 50/60HZ (1 minute)
Insulation Resistance	100 MΩ Min. (500VDC)
Max. ON/OFF Switching Mechanically Electrically	 300 operation/min 30 operation/min
Ambient Temperature	-25°C to +70°C
Operating Humidity	45 to 85% RH
Vibration Endurance Error Operation	 10 to 55Hz Double Amplitude 1.5mm 10 to 55Hz Double Amplitude 1.5mm
Shock Endurance Error Operation	 100G Min. 10G Min.
Life Expectancy Mechanically Electrically	 10^7 operations. Min. (no load) 10^5 operations. Min. (at rated coil voltage)
Weight	abt. 10grs.

9.REFERENCE DATA

Coil Temperature Rise

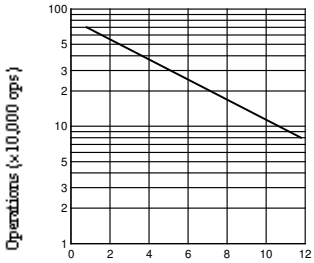


Operation Time

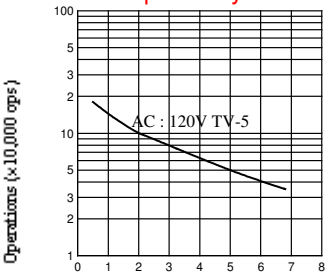


Life Expectancy

AC120V/DC24V cosΦ=1



Life Expectancy



ESP8266EX

Datasheet

Version 6.8

Espressif Systems

Copyright © 2022

About This Guide

This document introduces the specifications of ESP8266EX.

Release Notes

Date	Version	Release Notes
2015.12	V4.6	Updated Chapter 3.
2016.02	V4.7	Updated Section 3.6 and Section 4.1.
2016.04	V4.8	Updated Chapter 1.
2016.08	V4.9	Updated Chapter 1.
2016.11	V5.0	Added Appendix II “Learning Resources”.
2016.11	V5.1	Changed the power consumption during Deep-sleep from 10 μA to 20 μA in Table 5-2.
2016.11	V5.2	Changed the crystal frequency range from “26 MHz to 52 MHz” to “24 MHz to 52 MHz” in Section 3.3.
2016.12	V5.3	Changed the minimum working voltage from 3.0 V to 2.5 V.
2017.04	V5.4	Changed chip input and output impedance from 50 Ω to 39 + j6 Ω.
2017.10	V5.5	Updated Chapter 3 regarding the range of clock amplitude to 0.8 V ~ 1.5 V.
2017.11	V5.6	Updated VDDPST from 1.8 V ~ 3.3 V to 1.8 V ~ 3.6 V.
2017.11	V5.7	• Corrected a typo in the description of SDIO_DATA_0 in Table 2-1; • Added the testing conditions for the data in Table 5-2.
2018.02	V5.8	• Updated Wi-Fi protocols in Section 1.1; • Updated description of the integrated Tensilica processor in 3.1.

Date	Version	Release Notes
2018.09	V5.9	• Update document cover; • Added a note for Table 1-1; • Updated Wi-Fi key features in Section 1.1; • Updated description of the Wi-Fi function in 3.5; • Updated pin layout diagram; • Fixed a typo in Table 2-1; • Removed Section AHB and AHB module; • Restructured Section Power Management; • Fixed a typo in Section UART; • Removed description of transmission angle in Section IR Remote Control; • Other optimization (wording).
2018.11	V6.0	• Added an SPI pin in Table 4-2; • Updated the diagram of packing information.
2019.08	V6.1	Removed description of the GPIO function in Section 4.1.
2019.08	V6.2	Updated notes on CHIP_EN in Section 5.1
2019.12	V6.3	Add feedback links.
2020.04	V6.4	• Removed the description of “Antenna diversity”; • Updated the feedback links.
2020.07	V6.5	• Updated the description of HSPI in Section 4.3; • Updated links in Appendix.
2020.10	V6.6	• Fixed a typo in Figure 2-1; • Updated the link of *ESP8266 Pin List.*
2022.07	v6.7	• Updated Figure 2-1; • Updated the link of *ESP8266 Hardware Resources.*
2022.10	v6.8	Updated typos in Chapter 6.

Documentation Change Notification

Espressif provides email notifications to keep customers updated on changes to technical documentation. Please subscribe at https://www.espressif.com/en/subscribe.

Certification

Download certificates for Espressif products from https://www.espressif.com/en/certificates**.**

Table of Contents

1. Overview 1
   - 1.1. Wi-Fi Key Features 1
   - 1.2. Specifications 2
   - 1.3. Applications 3

2. Pin Definitions 4

3. Functional Description 7
   - 3.1. CPU, Memory, and Flash 7
     - 3.1.1. CPU 7
     - 3.1.2. Memory 7
     - 3.1.3. External Flash 8
   - 3.2. Clock 8
     - 3.2.1. High Frequency Clock 8
     - 3.2.2. External Clock Requirements 9
   - 3.3. Radio 9
     - 3.3.1. Channel Frequencies 9
     - 3.3.2. 2.4 GHz Receiver 10
     - 3.3.3. 2.4 GHz Transmitter 10
     - 3.3.4. Clock Generator 10
   - 3.4. Wi-Fi 10
     - 3.4.1. Wi-Fi Radio and Baseband 11
     - 3.4.2. Wi-Fi MAC 11
   - 3.5. Power Management 11

4. Peripheral Interface 13
   - 4.1. General Purpose Input/Output Interface (GPIO) 13
   - 4.2. Secure Digital Input/Output Interface (SDIO) 13
   - 4.3. Serial Peripheral Interface (SPI/HSPI) 14
     - 4.3.1. General SPI (Master/Slave) 14
     - 4.3.2. HSPI (Master/Slave) 14
   - 4.4. I2C Interface 15
   - 4.5. I2S Interface 15
   - 4.6. Universal Asynchronous Receiver Transmitter (UART) 15
   - 4.7. Pulse-Width Modulation (PWM) 16
   - 4.8. IR Remote Control 17

4.9. ADC (Analog-to-Digital Converter) 17

5. Electrical Specifications 19

5.1. Electrical Characteristics 19

5.2. RF Power Consumption 20

5.3. Wi-Fi Radio Characteristics 21

6. Package Information 22

I. Appendix - Pin List 23

II. Appendix - Learning Resources 24

II.1. Must-Read Documents 24

II.2. Must-Have Resources 24

1. Overview

Espressif's ESP8266EX delivers highly integrated Wi-Fi SoC solution to meet users' continuous demands for efficient power usage, compact design and reliable performance in the Internet of Things industry.

With the complete and self-contained Wi-Fi networking capabilities, ESP8266EX can perform either as a standalone application or as the slave to a host MCU. When ESP8266EX hosts the application, it promptly boots up from the flash. The integrated high-speed cache helps to increase the system performance and optimize the system memory. Also, ESP8266EX can be applied to any microcontroller design as a Wi-Fi adaptor through SPI/SDIO or UART interfaces.

ESP8266EX integrates antenna switches, RF balun, power amplifier, low noise receive amplifier, filters and power management modules. The compact design minimizes the PCB size and requires minimal external circuitries.

Besides the Wi-Fi functionalities, ESP8266EX also integrates an enhanced version of Tensilica's L106 Diamond series 32-bit processor and on-chip SRAM. It can be interfaced with external sensors and other devices through the GPIOs. Software Development Kit (SDK) provides sample codes for various applications.

Espressif Systems' Smart Connectivity Platform (ESCP) enables sophisticated features including:

- Fast switch between sleep and wakeup mode for energy-efficient purpose;
- Adaptive radio biasing for low-power operation
- Advance signal processing
- Spur cancellation and RF co-existence mechanisms for common cellular, Bluetooth, DDR, LVDS, LCD interference mitigation

1.1. Wi-Fi Key Features

- 802.11 b/g/n support
- 802.11 n support (2.4 GHz), up to 72.2 Mbps
- Defragmentation
- 2 x virtual Wi-Fi interface
- Automatic beacon monitoring (hardware TSF)
- Support Infrastructure BSS Station mode/SoftAP mode/Promiscuous mode

1.2. Specifications

Table 1-1. Specifications

Categories	Items	Parameters
Wi-Fi	Certification	Wi-Fi Alliance
	Protocols	802.11 b/g/n (HT20)
	Frequency Range	2.4 GHz ~ 2.5 GHz (2400 MHz ~ 2483.5 MHz)
	TX Power	802.11 b: +20 dBm
		802.11 g: +17 dBm
		802.11 n: +14 dBm
	Rx Sensitivity	802.11 b: –91 dbm (11 Mbps)
		802.11 g: –75 dbm (54 Mbps)
		802.11 n: –72 dbm (MCS7)
	Antenna	PCB Trace, External, IPEX Connector, Ceramic Chip
Hardware	CPU	Tensilica L106 32-bit processor
	Peripheral Interface	UART/SDIO/SPI/I2C/I2S/IR Remote Control
		GPIO/ADC/PWM/LED Light & Button
	Operating Voltage	2.5 V ~ 3.6 V
	Operating Current	Average value: 80 mA
	Operating Temperature Range	–40 °C ~ 125 °C
	Package Size	QFN32-pin (5 mm x 5 mm)
	External Interface	-
Software	Wi-Fi Mode	Station/SoftAP/SoftAP+Station
	Security	WPA/WPA2
	Encryption	WEP/TKIP/AES
	Firmware Upgrade	UART Download / OTA (via network)
	Software Development	Supports Cloud Server Development / Firmware and SDK for fast on-chip programming
	Network Protocols	IPv4, TCP/UDP/HTTP
	User Configuration	AT Instruction Set, Cloud Server, Android/iOS App

📖 ***Note:***

The TX power can be configured based on the actual user scenarios.

1.3. Applications

- Home appliances
- Home automation
- Smart plugs and lights
- Industrial wireless control
- Baby monitors
- IP cameras
- Sensor networks
- Wearable electronics
- Wi-Fi location-aware devices
- Security ID tags
- Wi-Fi position system beacons

2. Pin Definitions

Figure 2-1 shows the pin layout for 32-pin QFN package.

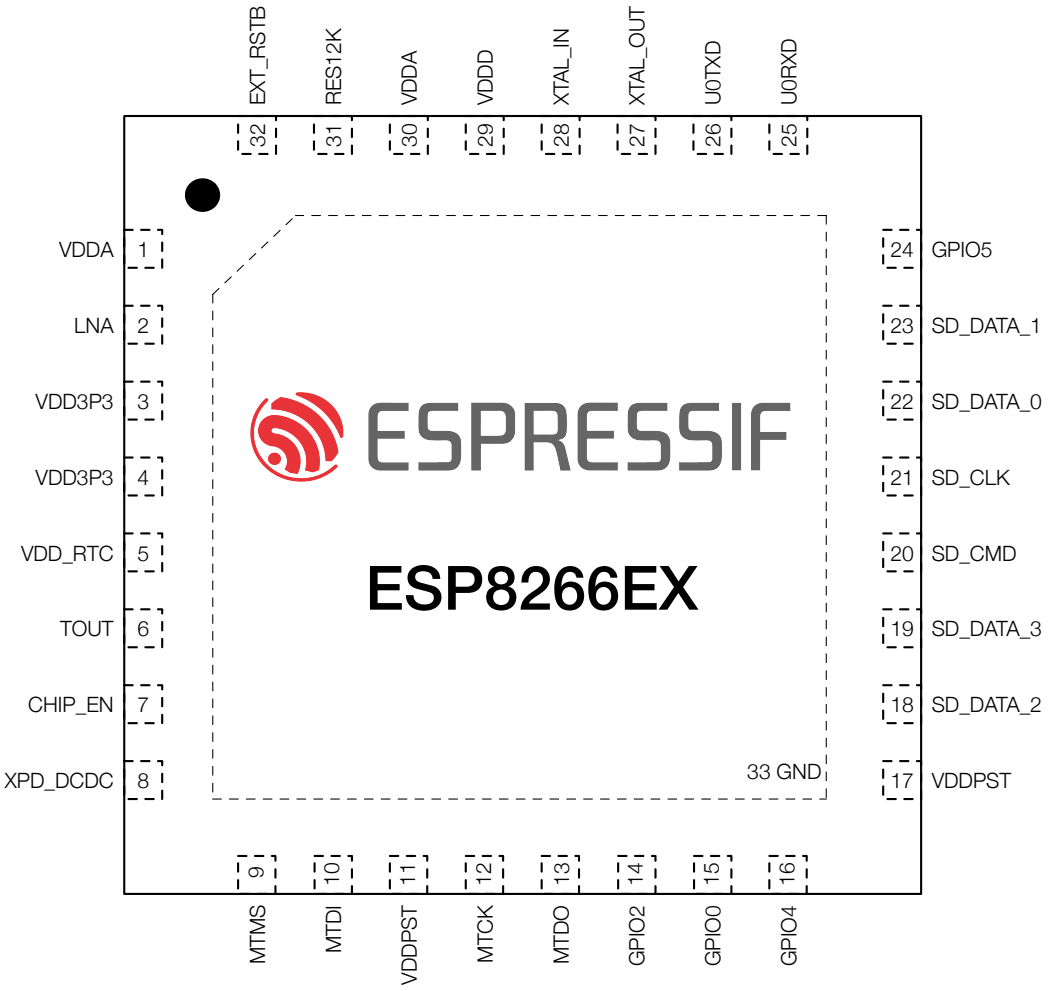


Figure 2-1. Pin Layout (Top View)

Table 2-1 lists the definitions and functions of each pin.

Table 2-1. ESP8266EX Pin Definitions

Pin	Name	Type	Function
1	VDDA	P	Analog Power 2.5 V ~ 3.6 V
2	LNA	I/O	RF antenna interface Chip output impedance = 39 + j6 Ω. It is suggested to retain the π-type matching network to match the antenna.
3	VDD3P3	P	Amplifier Power 2.5 V ~ 3.6 V

Pin	Name	Type	Function
4	VDD3P3	P	Amplifier Power 2.5 V ~ 3.6 V
5	VDD_RTC	P	NC (1.1 V)
6	TOUT	I	ADC pin. It can be used to test the power-supply voltage of VDD3P3 (Pin3 and Pin4) and the input power voltage of TOUT (Pin 6). However, these two functions cannot be used simultaneously.
7	CHIP_EN	I	Chip Enable High: On, chip works properly Low: Off, small current consumed
8	XPD_DCDC	I/O	Deep-sleep wakeup (need to be connected to EXT_RSTB); GPIO16
9	MTMS	I/O	GPIO 14; HSPI_CLK
10	MTDI	I/O	GPIO 12; HSPI_MISO
11	VDDPST	P	Digital/IO Power Supply (1.8 V ~ 3.6 V)
12	MTCK	I/O	GPIO 13; HSPI_MOSI; UART0_CTS
13	MTDO	I/O	GPIO 15; HSPI_CS; UART0_RTS
14	GPIO2	I/O	UART TX during flash programming; GPIO2
15	GPIO0	I/O	GPIO0; SPI_CS2
16	GPIO4	I/O	GPIO4
17	VDDPST	P	Digital/IO Power Supply (1.8 V ~ 3.6 V)
18	SDIO_DATA_2	I/O	Connect to SD_D2 (Series R: 20 Ω); SPIHD; HSPIHD; GPIO9
19	SDIO_DATA_3	I/O	Connect to SD_D3 (Series R: 200 Ω); SPIWP; HSPIWP; GPIO10
20	SDIO_CMD	I/O	Connect to SD_CMD (Series R: 200 Ω); SPI_CS0; GPIO11
21	SDIO_CLK	I/O	Connect to SD_CLK (Series R: 200 Ω); SPI_CLK; GPIO6
22	SDIO_DATA_0	I/O	Connect to SD_D0 (Series R: 200 Ω); SPI_MISO; GPIO7
23	SDIO_DATA_1	I/O	Connect to SD_D1 (Series R: 200 Ω); SPI_MOSI; GPIO8
24	GPIO5	I/O	GPIO5
25	U0RXD	I/O	UART Rx during flash programming; GPIO3
26	U0TXD	I/O	UART TX during flash programming; GPIO1; SPI_CS1
27	XTAL_OUT	I/O	Connect to crystal oscillator output, can be used to provide BT clock input
28	XTAL_IN	I/O	Connect to crystal oscillator input
29	VDDD	P	Analog Power 2.5 V ~ 3.6 V
30	VDDA	P	Analog Power 2.5 V ~ 3.6 V

Pin	Name	Type	Function
31	RES12K	I	Serial connection with a 12 kΩ resistor and connect to the ground
32	EXT_RSTB	I	External reset signal (Low voltage level: active)

📖 ***Note:***

1. *GPIO2, GPIO0, and MTDO are used to select booting mode and the SDIO mode;*
2. *U0TXD should not be pulled externally to a low logic level during the powering-up.*

3. Functional Description

The functional diagram of ESP8266EX is shown as in Figure 3-1.

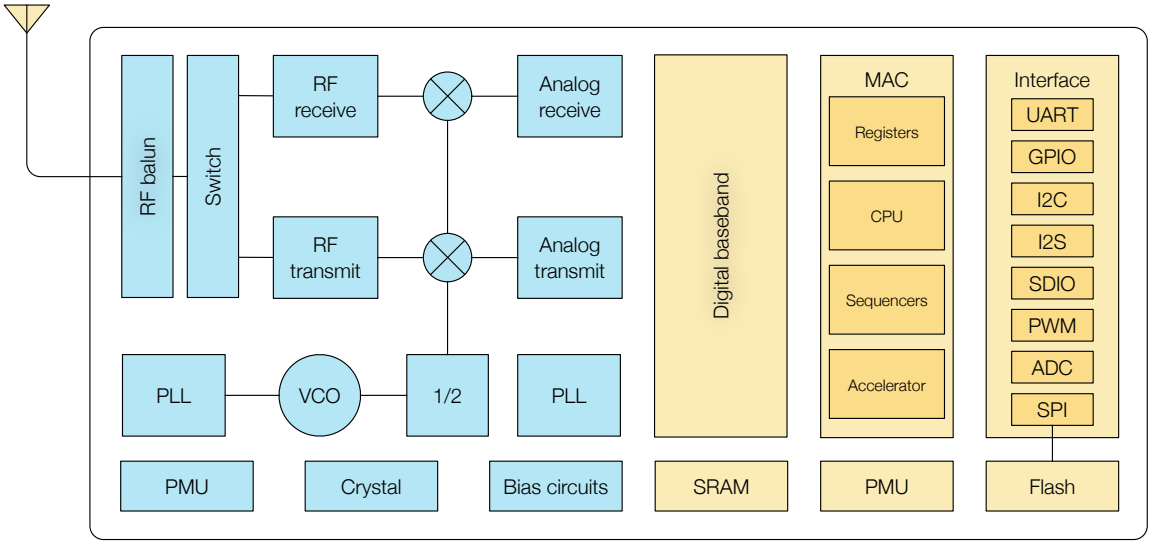


Figure 3-1. Functional Block Diagram

3.1. CPU, Memory, and Flash

3.1.1. CPU

The ESP8266EX integrates a Tensilica L106 32-bit RISC processor, which achieves extra-low power consumption and reaches a maximum clock speed of 160 MHz. The Real-Time Operating System (RTOS) and Wi-Fi stack allow 80% of the processing power to be available for user application programming and development. The CPU includes the interfaces as below:

- Programmable RAM/ROM interfaces (iBus), which can be connected with memory controller, and can also be used to visit flash.
- Data RAM interface (dBus), which can connected with memory controller.
- AHB interface which can be used to visit the register.

3.1.2. Memory

ESP8266EX Wi-Fi SoC integrates memory controller and memory units including SRAM and ROM. MCU can access the memory units through iBus, dBus, and AHB interfaces. All memory units can be accessed upon request, while a memory arbiter will decide the running sequence according to the time when these requests are received by the processor.

According to our current version of SDK, SRAM space available to users is assigned as below.

- RAM size < 50 kB, that is, when ESP8266EX is working under the Station mode and connects to the router, the maximum programmable space accessible in Heap + Data section is around 50 kB.
- There is no programmable ROM in the SoC. Therefore, user program must be stored in an external SPI flash.

3.1.3. External Flash

ESP8266EX uses external SPI flash to store user programs, and supports up to 16 MB memory capacity theoretically.

The minimum flash memory of ESP8266EX is shown below:

- OTA disabled: 512 kB at least
- OTA enabled: 1 MB at least

> ⚠ **Notice:**
>
> *SPI mode supported: Standard SPI, Dual SPI and Quad SPI. The correct SPI mode should be selected when flashing bin files to ESP8266. Otherwise, the downloaded firmware/program may not be working properly.*

3.2. Clock

3.2.1. High Frequency Clock

The high frequency clock on ESP8266EX is used to drive both transmit and receive mixers. This clock is generated from internal crystal oscillator and external crystal. The crystal frequency ranges from 24 MHz to 52 MHz.

The internal calibration inside the crystal oscillator ensures that a wide range of crystals can be used, nevertheless the quality of the crystal is still a factor to consider to have reasonable phase noise and good Wi-Fi sensitivity. Refer to Table 3-1 to measure the frequency offset.

Table 3-1. High Frequency Clock Specifications

Parameter	Symbol	Min	Max	Unit
Frequency	FXO	24	52	MHz
Loading capacitance	CL	-	32	pF
Motional capacitance	CM	2	5	pF
Series resistance	RS	0	65	Ω
Frequency tolerance	ΔFXO	–15	15	ppm

Parameter	Symbol	Min	Max	Unit
Frequency vs temperature (–25 °C ~ 75 °C)	ΔFXO,Temp	–15	15	ppm

3.2.2. External Clock Requirements

An externally generated clock is available with the frequency ranging from 24 MHz to 52 MHz. The following characteristics are expected to achieve good performance of radio.

Table 3-2. External Clock Reference

Parameter	Symbol	Min	Max	Unit
Clock amplitude	VXO	0.8	1.5	Vpp
External clock accuracy	ΔFXO,EXT	–15	15	ppm
Phase noise @1-kHz offset, 40-MHz clock	-	-	–120	dBc/Hz
Phase noise @10-kHz offset, 40-MHz clock	-	-	–130	dBc/Hz
Phase noise @100-kHz offset, 40-MHz clock	-	-	–138	dBc/Hz

3.3. Radio

ESP8266EX radio consists of the following blocks.

- 2.4 GHz receiver
- 2.4 GHz transmitter
- High speed clock generators and crystal oscillator
- Bias and regulators
- Power management

3.3.1. Channel Frequencies

The RF transceiver supports the following channels according to IEEE802.11 b/g/n standards.

Table 3-3. Frequency Channel

Channel No.	Frequency (MHz)	Channel No.	Frequency (MHz)
1	2412	8	2447
2	2417	9	2452
3	2422	10	2457
4	2427	11	2462
5	2432	12	2467

Channel No.	Frequency (MHz)	Channel No.	Frequency (MHz)
6	2437	13	2472
7	2442	14	2484

3.3.2. 2.4 GHz Receiver

The 2.4 GHz receiver down-converts the RF signals to quadrature baseband signals and converts them to the digital domain with 2 high resolution high speed ADCs. To adapt to varying signal channel conditions, RF filters, automatic gain control (AGC), DC offset cancelation circuits and baseband filters are integrated within ESP8266EX.

3.3.3. 2.4 GHz Transmitter

The 2.4 GHz transmitter up-converts the quadrature baseband signals to 2.4 GHz, and drives the antenna with a high-power CMOS power amplifier. The function of digital calibration further improves the linearity of the power amplifier, enabling a state of art performance of delivering +19.5 dBm average TX power for 802.11b transmission and +18 dBm for 802.11n (MSC0) transmission.

Additional calibrations are integrated to offset any imperfections of the radio, such as:

- Carrier leakage
- I/Q phase matching
- Baseband nonlinearities

These built-in calibration functions reduce the product test time and make the test equipment unnecessary.

3.3.4. Clock Generator

The clock generator generates quadrature 2.4 GHz clock signals for the receiver and transmitter. All components of the clock generator are integrated on the chip, including all inductors, varactors, loop filters, linear voltage regulators and dividers.

The clock generator has built-in calibration and self test circuits. Quadrature clock phases and phase noise are optimized on-chip with patented calibration algorithms to ensure the best performance of the receiver and transmitter.

3.4. Wi-Fi

ESP8266EX implements TCP/IP and full 802.11 b/g/n WLAN MAC protocol. It supports Basic Service Set (BSS) STA and SoftAP operations under the Distributed Control Function (DCF). Power management is handled with minimum host interaction to minimize active-duty period.

3.4.1. Wi-Fi Radio and Baseband

The ESP8266EX Wi-Fi Radio and Baseband support the following features:

- 802.11 b and 802.11 g
- 802.11 n MCS0-7 in 20 MHz bandwidth
- 802.11 n 0.4 μs guard-interval
- up to 72.2 Mbps of data rate
- Receiving STBC 2 x 1
- Up to 20.5 dBm of transmitting power
- Adjustable transmitting power

3.4.2. Wi-Fi MAC

The ESP8266EX Wi-Fi MAC applies low-level protocol functions automatically, as follows:

- 2 × virtual Wi-Fi interfaces
- Infrastructure BSS Station mode/SoftAP mode/Promiscuous mode
- Request To Send (RTS), Clear To Send (CTS) and Immediate Block ACK
- Defragmentation
- CCMP (CBC-MAC, counter mode), TKIP (MIC, RC4), WEP (RC4) and CRC
- Automatic beacon monitoring (hardware TSF)
- Dual and single antenna Bluetooth co-existence support with optional simultaneous receive (Wi-Fi/Bluetooth) capability

3.5. Power Management

ESP8266EX is designed with advanced power management technologies and intended for mobile devices, wearable electronics and the Internet of Things applications.

The low-power architecture operates in the following modes:

- Active mode: The chip radio is powered on. The chip can receive, transmit, or listen.
- Modem-sleep mode: The CPU is operational. The Wi-Fi and radio are disabled.
- Light-sleep mode: The CPU and all peripherals are paused. Any wake-up events (MAC, host, RTC timer, or external interrupts) will wake up the chip.
- Deep-sleep mode: Only the RTC is operational and all other part of the chip are powered off.

Table 3-4. Power Consumption by Power Modes

Power Mode	Description	Power Consumption
Active (RF working)	Wi-Fi TX packet	Please refer to Table 5-2.
	Wi-Fi RX packet	
Modem-sleep①	CPU is working	15 mA
Light-sleep②	-	0.9 mA
Deep-sleep③	Only RTC is working	20 uA
Shut down	-	0.5 uA

📖 ***Notes:***

① ***Modem-sleep*** *mode is used in the applications that require the CPU to be working, as in PWM or I2S applications. According to 802.11 standards (like U-APSD), it shuts down the Wi-Fi Modem circuit while maintaining a Wi-Fi connection with no data transmission to optimize power consumption. E.g., in DTIM3, maintaining a sleep of 300 ms with a wakeup of 3 ms cycle to receive AP's Beacon packages at interval requires about 15 mA current.*

② *During* ***Light-sleep*** *mode, the CPU may be suspended in applications like Wi-Fi switch. Without data transmission, the Wi-Fi Modem circuit can be turned off and CPU suspended to save power consumption according to the 802.11 standards (U-APSD). E.g. in DTIM3, maintaining a sleep of 300 ms with a wakeup of 3ms to receive AP's Beacon packages at interval requires about 0.9 mA current.*

③ *During* ***Deep-sleep*** *mode, Wi-Fi is turned off. For applications with long time lags between data transmission, e.g. a temperature sensor that detects the temperature every 100 s, sleeps for 300 s and wakes up to connect to the AP (taking about 0.3 ~ 1 s), the overall average current is less than 1 mA. The current of 20 μA is acquired at the voltage of 2.5 V.*

4. Peripheral Interface

4.1. General Purpose Input/Output Interface (GPIO)

ESP8266EX has 17 GPIO pins which can be assigned to various functions by programming the appropriate registers.

Each GPIO PAD can be configured with internal pull-up or pull-down (XPD_DCDC can only be configured with internal pull-down, other GPIO PAD can only be configured with internal pull-up), or set to high impedance. When configured as an input, the data are stored in software registers; the input can also be set to edge-trigger or level trigger CPU interrupts. In short, the IO pads are bi-directional, non-inverting and tristate, which includes input and output buffer with tristate control inputs.

These pins, when working as GPIOs, can be multiplexed with other functions such as I2C, I2S, UART, PWM, and IR Remote Control, etc.

4.2. Secure Digital Input/Output Interface (SDIO)

ESP8266EX has one Slave SDIO, the definitions of which are described as Table 4-1, which supports 25 MHz SDIO v1.1 and 50 MHz SDIO v2.0, and 1 bit/4 bit SD mode and SPI mode.

Table 4-1. Pin Definitions of SDIOs

Pin Name	Pin Num	IO	Function Name
SDIO_CLK	21	IO6	SDIO_CLK
SDIO_DATA0	22	IO7	SDIO_DATA0
SDIO_DATA1	23	IO8	SDIO_DATA1
SDIO_DATA_2	18	IO9	SDIO_DATA_2
SDIO_DATA_3	19	IO10	SDIO_DATA_3
SDIO_CMD	20	IO11	SDIO_CMD

4.3. Serial Peripheral Interface (SPI/HSPI)

ESP8266EX has two SPIs.

- One general Slave/Master SPI
- One general Slave/Master HSPI

Functions of all these pins can be implemented via hardware.

4.3.1. General SPI (Master/Slave)

Table 4-2. Pin Definitions of SPIs

Pin Name	Pin Num	IO	Function Name
SDIO_CLK	21	IO6	SPICLK
SDIO_DATA0	22	IO7	SPIQ/MISO
SDIO_DATA1	23	IO8	SPID/MOSI
SDIO_DATA_2	18	IO9	SPIHD
SDIO_DATA_3	19	IO10	SPIWP
U0TXD	26	IO1	SPICS1
GPIO0	15	IO0	SPICS2
SDIO_CMD	20	IO11	SPICS0

📖 ***Note:***

SPI mode can be implemented via software programming. The clock frequency is 80 MHz at maximum when working as a master, 20 MHz at maximum when working as a slave.

4.3.2. HSPI (Master/Slave)

Table 4-3. Pin Definitions of HSPI

Pin Name	Pin Num	IO	Function Name
MTMS	9	IO14	HSPICLK
MTDI	10	IO12	HSPIQ/MISO
MTCK	12	IO13	HSPID/MOSI
MTDO	13	IO15	HPSICS

📖 ***Note:***

SPI mode can be implemented via software programming. The clock frequency is 20 MHz at maximum.

4.4. I2C Interface

ESP8266EX has one I2C, which is realized via software programming, used to connect with other microcontrollers and other peripheral equipments such as sensors. The pin definition of I2C is as below.

Table 4-4. Pin Definitions of I2C

Pin Name	Pin Num	IO	Function Name
MTMS	9	IO14	I2C_SCL
GPIO2	14	IO2	I2C_SDA

Both I2C Master and I2C Slave are supported. I2C interface functionality can be realized via software programming, and the clock frequency is 100 kHz at maximum.

4.5. I2S Interface

ESP8266EX has one I2S data input interface and one I2S data output interface, and supports the linked list DMA. I2S interfaces are mainly used in applications such as data collection, processing, and transmission of audio data, as well as the input and output of serial data. For example, LED lights (WS2812 series) are supported. The pin definition of I2S is shown in Table 4-5.

Table 4-5. Pin Definitions of I2S

I2S Data Input			
Pin Name	Pin Num	IO	Function Name
MTDI	10	IO12	I2SI_DATA
MTCK	12	IO13	I2SI_BCK
MTMS	9	IO14	I2SI_WS
MTDO	13	IO15	I2SO_BCK
U0RXD	25	IO3	I2SO_DATA
GPIO2	14	IO2	I2SO_WS

4.6. Universal Asynchronous Receiver Transmitter (UART)

ESP8266EX has two UART interfaces UART0 and UART1, the definitions are shown in Table 4-6.

Table 4-6. Pin Definitions of UART

Pin Type	Pin Name	Pin Num	IO	Function Name
UART0	U0RXD	25	IO3	U0RXD
	U0TXD	26	IO1	U0TXD
	MTDO	13	IO15	U0RTS
	MTCK	12	IO13	U0CTS
UART1	GPIO2	14	IO2	U1TXD
	SD_D1	23	IO8	U1RXD

Data transfers to/from UART interfaces can be implemented via hardware. The data transmission speed via UART interfaces reaches 115200 x 40 (4.5 Mbps).

UART0 can be used for communication. It supports flow control. Since UART1 features only data transmit signal (TX), it is usually used for printing log.

> 📖 ***Note:***
>
> *By default, UART0 outputs some printed information when the device is powered on and booting up. The baud rate of the printed information is relevant to the frequency of the external crystal oscillator. If the frequency of the crystal oscillator is 40 MHz, then the baud rate for printing is 115200; if the frequency of the crystal oscillator is 26 MHz, then the baud rate for printing is 74880. If the printed information exerts any influence on the functionality of the device, it is suggested to block the printing during the power-on period by changing (U0TXD, U0RXD) to (MTDO, MTCK).*

4.7. Pulse-Width Modulation (PWM)

ESP8266EX has four PWM output interfaces. They can be extended by users themselves. The pin definitions of the PWM interfaces are defined as below.

Table 4-7. Pin Definitions of PWM

Pin Name	Pin Num	IO	Function Name
MTDI	10	IO12	PWM0
MTDO	13	IO15	PWM1
MTMS	9	IO14	PWM2
GPIO4	16	IO4	PWM3

The functionality of PWM interfaces can be implemented via software programming. For example, in the LED smart light demo, the function of PWM is realized by interruption of the timer, the minimum resolution reaches as high as 44 ns. PWM frequency range is adjustable from 1000 µs to 10000 µs, i.e., between 100 Hz and 1 kHz. When the PWM

frequency is 1 kHz, the duty ratio will be 1/22727, and a resolution of over 14 bits will be achieved at 1 kHz refresh rate.

4.8. IR Remote Control

ESP8266EX currently supports one infrared remote control interface. For detailed pin definitions, please see Table 4-8 below.

Table 4-8. Pin Definitions of IR Remote Control

Pin Name	Pin Num	IO	Function Name
MTMS	9	IO14	IR TX
GPIO5	24	IO 5	IR Rx

The functionality of Infrared remote control interface can be implemented via software programming. NEC coding, modulation, and demodulation are supported by this interface. The frequency of modulated carrier signal is 38 kHz, while the duty ratio of the square wave is 1/3. The transmission range is around 1m which is determined by two factors: one is the maximum current drive output, the other is internal current-limiting resistance value in the infrared receiver. The larger the resistance value, the lower the current, so is the power, and vice versa.

4.9. ADC (Analog-to-Digital Converter)

ESP8266EX is embedded with a 10-bit precision SAR ADC. TOUT (Pin6) is defined as below:

Table 4-9. Pin Definition of ADC

Pin Name	Pin Num	Function Name
TOUT	6	ADC Interface

The following two measurements can be implemented using ADC (Pin6). However, they cannot be implemented at the same time.

- Measure the power supply voltage of VDD3P3 (Pin3 and Pin4).

Hardware Design	TOUT must be floating.
RF Initialization Parameter	The 107th byte of ***esp_init_data_default.bin*** (0 ~ 127 bytes), `vdd33_const` must be set to `0xFF`.
RF Calibration Process	Optimize the RF circuit conditions based on the testing results of VDD3P3 (Pin3 and Pin4).
User Programming	Use `system_get_vdd33` instead of `system_adc_read`.

- Measure the input voltage of TOUT (Pin6).

Hardware Design	The input voltage range is 0 to 1.0 V when TOUT is connected to external circuit.
RF Initialization Parameter	The value of the 107th byte of ***esp_init_data_default.bin*** (0 ~ 127 bytes), `vdd33_const` must be set to the real power supply voltage of Pin3 and Pin4. The unit and effective value range of `vdd33_const` is 0.1 V and 18 to 36, respectively, thus making the working power voltage range of ESP8266EX between 1.8 V and 3.6 V.
RF Calibration Process	Optimize the RF circuit conditions based on the value of `vdd33_const`. The permissible error is ± 0.2 V.
User Programming	Use `system_adc_read` instead of `system_get_vdd33`.

📖 ***Notes:***

esp_init_data_default.bin *is provided in SDK package which contains RF initialization parameters (0 ~ 127 bytes). The name of the 107th byte in* ***esp_init_data_default.bin*** *is* `vdd33_const`*, which is defined as below:*

- *When vdd33_const = 0xff, the power voltage of Pin3 and Pin4 will be tested by the internal self-calibration process of ESP8266EX itself. RF circuit conditions should be optimized according to the testing results.*
- *When 18 = < vdd33_const = < 36, ESP8266EX RF Calibration and optimization process is implemented via (vdd33_const/10).*
- *When vdd33_const < 18 or 36 < vdd33_const < 255, vdd33_const is invalid. ESP8266EX RF Calibration and optimization process is implemented via the default value 3.3 V.*

5. Electrical Specifications

5.1. Electrical Characteristics

Table 5-1. Electrical Characteristics

Parameters		Conditions	Min	Typical	Max	Unit
Operating Temperature Range		-	–40	Normal	125	°C
Maximum Soldering Temperature		IPC/JEDEC J-STD-020	-	-	260	°C
Working Voltage Value		-	2.5	3.3	3.6	V
I/O	V_{IL}	-	–0.3	-	0.25 V_{IO}	V
	V_{IH}		0.75 V_{IO}		3.6	
	V_{OL}	-	-	-	0.1 V_{IO}	
	V_{OH}		0.8 V_{IO}		-	
	I_{MAX}	-	-	-	12	mA
Electrostatic Discharge (HBM)		TAMB = 25 °C	-	-	2	KV
Electrostatic Discharge (CDM)		TAMB = 25 °C	-	-	0.5	KV

Notes on CHIP_EN:

The figure below shows ESP8266EX power-up and reset timing. Details about the parameters are listed in Table 5-2.

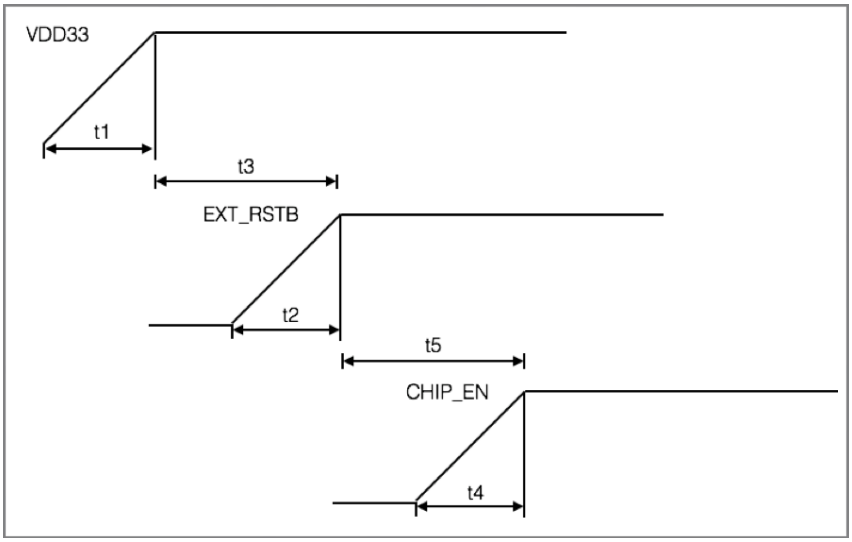


Figure 5-1. ESP8266EX Power-up and Reset Timing

Table 5-2. Description of ESP8266EX Power-up and Reset Timing Parameters

	Description	Min	Max	Unit
t1	The rise-time of VDD33	10	2000	µs
t2	The rise-time of EXT_RSTB	0	2	ms
t3	EXT_RSTB goes high after VDD33	0.1	-	ms
t4	The rise-time of CHIP_EN	0	2	ms
t5	CHIP_EN goes high after EXT_RSTB	0.1	-	ms

5.2. RF Power Consumption

Unless otherwise specified, the power consumption measurements are taken with a 3.0 V supply at 25 °C of ambient temperature. All transmitters' measurements are based on a 50% duty cycle.

Table 5-3. Power Consumption

Parameters	Min	Typical	Max	Unit
TX802.11 b, CCK 11 Mbps, P_{OUT} = +17 dBm	-	170	-	mA
TX802.11 g, OFDM 54Mbps, P_{OUT} = +15 dBm	-	140	-	mA
TX802.11 n, MCS7, P_{OUT} = +13 dBm	-	120	-	mA
Rx802.11 b, 1024 bytes packet length, –80 dBm	-	50	-	mA
Rx802.11 g, 1024 bytes packet length, –70 dBm	-	56	-	mA
Rx802.11 n, 1024 bytes packet length, –65 dBm	-	56	-	mA

5.3. Wi-Fi Radio Characteristics

The following data are from tests conducted at room temperature, with a 3.3 V power supply.

Table 5-3. Wi-Fi Radio Characteristics

Parameters	Min	Typical	Max	Unit
Input frequency	2412	-	2484	MHz
Output impedance	-	39 + j6	-	Ω
Output power of PA for 72.2 Mbps	15.5	16.5	17.5	dBm
Output power of PA for 11b mode	19.5	20.5	21.5	dBm
Sensitivity				
DSSS, 1 Mbps	-	–98	-	dBm
CCK, 11 Mbps	-	–91	-	dBm
6 Mbps (1/2 BPSK)	-	–93	-	dBm
54 Mbps (3/4 64-QAM)	-	–75	-	dBm
HT20, MCS7 (65 Mbps, 72.2 Mbps)	-	–72	-	dBm
Adjacent Channel Rejection				
OFDM, 6 Mbps	-	37	-	dB
OFDM, 54 Mbps	-	21	-	dB
HT20, MCS0	-	37	-	dB
HT20, MCS7	-	20	-	dB

6. Package Information

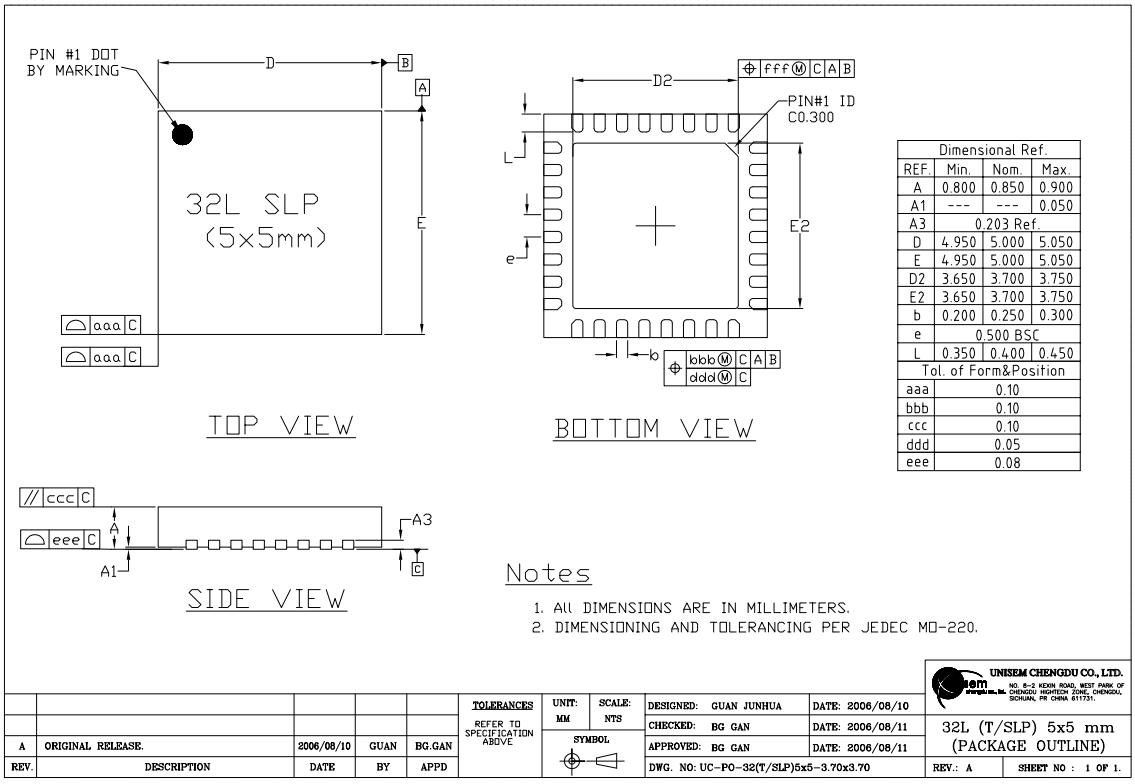


Figure 6-1. ESP8266EX Package

I. Appendix - Pin List

For detailed pin information, please see *ESP8266 Pin List*.

- Digital Die Pin List
- Buffer Sheet
- Register List
- Strapping List

> 📖 ***Notes:***
>
> - *INST_NAME refers to the IO_MUX REGISTER defined in* ***eagle_soc.h****, for example MTDI_U refers to PERIPHS_IO_MUX_MTDI_U.*
> - *Net Name refers to the pin name in schematic.*
> - *Function refers to the multifunction of each pin pad.*
> - *Function number 1 ~ 5 correspond to FUNCTION 0 ~ 4 in SDK. For example, set MTDI to GPIO12 as follows.*
>   - `#define FUNC_GPIO12  3 //defined in eagle_soc.h`
>   - `PIN_FUNC_SELECT(PERIPHS_IO_MUX_MTDI_U,FUNC_GPIO12)`

II. Appendix - Learning Resources

II.1. Must-Read Documents

- *ESP8266 Quick Start Guide*

  Description: This document is a quick user guide to getting started with ESP8266. It includes an introduction to the ESP-LAUNCHER, instructions on how to download firmware to the board and run it, how to compile the AT application, as well as the structure and debugging method of RTOS SDK. Basic documentation and other related resources for the ESP8266 are also provided.
- *ESP8266 SDK Getting Started Guide*

  Description: This document takes ESP-LAUNCHER and ESP-WROOM-02 as examples of how to use the ESP8266 SDK. The contents include preparations before compilation, SDK compilation and firmware download.
- *ESP8266 Pin List*

  Description: This link directs you to a list containing the type and function of every ESP8266 pin.
- *ESP8266 Hardware Design Guideline*

  Description: This document provides a technical description of the ESP8266 series of products, including ESP8266EX, ESP-LAUNCHER and ESP-WROOM.
- *ESP8266 Technical Reference*

  Description: This document provides an introduction to the interfaces integrated on ESP8266. Functional overview, parameter configuration, function description, application demos and other pieces of information are included.
- *ESP8266 Hardware Resources*

  Description: This zip package includes manufacturing BOMs, schematics and PCB layouts of ESP8266 boards and modules.
- *FAQ*

II.2. Must-Have Resources

- *ESP8266 SDKs*

  Description: This webpage provides links both to the latest version of the ESP8266 SDK and the older ones.
- *ESP8266 Tools*

Description: This webpage provides links to both the ESP8266 flash download tools and the ESP8266 performance evaluation tools.

- *ESP8266 Apps*
- *ESP8266 Certification and Test Guide*
- *ESP8266 BBS*
- *ESP8266 Resources*

Espressif IOT Team

www.espressif.com

Disclaimer and Copyright Notice

Information in this document, including URL references, is subject to change without notice.

THIS DOCUMENT IS PROVIDED AS IS WITH NO WARRANTIES WHATSOEVER, INCLUDING ANY WARRANTY OF MERCHANTABILITY, NON-INFRINGEMENT, FITNESS FOR ANY PARTICULAR PURPOSE, OR ANY WARRANTY OTHERWISE ARISING OUT OF ANY PROPOSAL, SPECIFICATION OR SAMPLE.

All liability, including liability for infringement of any proprietary rights, relating to use of information in this document is disclaimed. No licenses express or implied, by estoppel or otherwise, to any intellectual property rights are granted herein.

The Wi-Fi Alliance Member logo is a trademark of the Wi-Fi Alliance. The Bluetooth logo is a registered trademark of Bluetooth SIG.

All trade names, trademarks and registered trademarks mentioned in this document are property of their respective owners, and are hereby acknowledged.

Copyright © 2022 Espressif Inc. All rights reserved.