



**Universidad de Valladolid**



**ESCUELA DE INGENIERÍAS  
INDUSTRIALES**

**UNIVERSIDAD DE VALLADOLID**

**ESCUELA DE INGENIERÍAS INDUSTRIALES**

**Grado en Ingeniería Mecánica**

**IMPLANTACIÓN, ESTUDIO Y CONTROL DE  
CALIDAD DE VARIABLES CLIMÁTICAS EN EL  
ÁMBITO DE LA METROLOGÍA 4.0.**

**Autor:**

**de la Fuente Ruiz, Sonia**

**Tutor:**

**Santos Martín, Francisco Javier**

Dpto. CMeIM/EGI/ICGF/IM/IPF

Área de Ingeniería de los  
Procesos de Fabricación

**Valladolid, Junio 2019**



## AGRADECIMIENTOS

---

En primer lugar dar las gracias a mis padres, por creer en mí y aportarme todo y cuanto ha sido necesario en cada momento con el fin último de lograr mis metas y formarme en lo que siempre quise.

También, a mi tutor Francisco Javier que me ha brindado su ayuda y guiado en la consecución de este proyecto.

Y, por último, no menos importante, a mis compañeros de Universidad con los que los días en esta dura etapa han sido más llevaderos, pero sobre todo, a una gran compañera y amiga que siempre llevaré conmigo, Patricia, que gracias a su gran apoyo, nos teníamos la una a la otra en los momentos que se hacían más cuesta arriba.

"Sólo aquellos que arriesgan yendo muy lejos pueden encontrar lo lejos que pueden llegar". –T.S. Eliot.

## RESUMEN

---

Tras la incipiente inserción de la Industria 4.0 en las empresas, este TFG se ha enfocado concretamente en la Metrología 4.0, para lo cual se ha analizado y caracterizado el comportamiento de dos sensores que miden variables climáticas de forma simultánea.

Para ello se ha utilizado Arduino junto con una plataforma de libre acceso, Thinger.io, lo que ha permitido exportar los datos recogidos a una hoja de cálculo para su posterior tratamiento y extracción de conclusiones, de manera que se logre una optimización de los recursos, mejora en la productividad, así como la garantía de una plena satisfacción del usuario.

Para una mayor profundización, se ha establecido una función modelo la cual desencadenará el posterior cálculo de una estimación de la incertidumbre de medida, lo que mostrará la idoneidad de estos sensores como medio para el control y medición de variables climáticas, gracias a la óptima reproducibilidad que ofrecen.

**Palabras clave:** calidad, Metrología 4.0, calibración, trazabilidad, microcontrolador.

## ABSTRACT

---

After the incipient implantation of Industry 4.0 in companies, this TFG has focused specifically on Metrology 4.0, for which the behavior of two sensors that measure climate variables simultaneously has been analyzed and characterized.

To this end, Arduino has been used together with a free access platform, Thinger.io, which has allowed the data collected to be exported to a spreadsheet for further processing and extraction of conclusions, in order to optimize resources, improvement in productivity, as well as the guarantee of full user satisfaction.

For further study, a model function has been established, which will trigger the subsequent calculation of an estimation of the measurement uncertainty, that will show the suitability of these sensors as a means for the control and measurement of climatic variables, thanks to their optimal reproducibility.

**Keywords:** quality, Metrology 4.0, calibration, traceability, microcontroller.

# ÍNDICE

<b>CAPÍTULO 1. OBJETIVOS, ALCANCE Y ESTRUCTURA DEL TFG. ....</b>	<b>1 -</b>
1.1 ANTECEDENTES Y JUSTIFICACIÓN. ....	2 -
1.2 OBJETIVOS. ....	3 -
1.3 ESTRUCTURA DEL TRABAJO. ....	4 -
<b>CAPÍTULO 2. INTRODUCCIÓN A LA INDUSTRIA 4.0 E IMPLEMENTACIÓN EN LA METROLOGÍA.-</b>	<b>7 -</b>
2.1 CRONOLOGÍA REVOLUCIONES INDUSTRIALES. ....	8 -
2.2 INTRODUCCIÓN A LA TRANSFORMACIÓN DIGITAL O INDUSTRIA 4.0. ....	10 -
2.3 CONCEPTOS CLAVES EN LA INDUSTRIA 4.0. ....	11 -
2.4 VENTAJAS E INCONVENIENTES DE LA INDUSTRIA 4.0. ....	15 -
2.5 EJEMPLOS DE IMPLANTACIÓN DE LA INDUSTRIA 4.0. ....	16 -
2.5.1. EJEMPLO 1. TESLA (AUTOMOCIÓN). ....	16 -
2.5.2. EJEMPLO 2. WEARLUMB. ....	18 -
2.5.3. EJEMPLO 3. INTRACE (SECTOR INDUSTRIAL). ....	20 -
2.5.4. EJEMPLO 4. MONITORIZACIÓN DE CULTIVOS (SECTOR AGRÍCOLA). ....	21 -
2.6 METROLOGÍA INDUSTRIAL. ....	21 -
2.6.1 LA METROLOGÍA -> METROLOGÍA INDUSTRIAL. ....	21 -
<b>CAPÍTULO 3. IMPLEMENTACIÓN INDUSTRIA 4.0: ARDUINO Y PLATAFORMA THINGER.IO. ....</b>	<b>25 -</b>
3.1 HARDWARE: ARDUINO. ....	26 -
3.1.1. CONCEPTUALIZACIÓN DE ARDUINO. ....	26 -
3.1.2. HISTORIA. ....	27 -
3.1.3. VENTAJAS. ....	28 -
3.1.4. ESTRUCTURA DE UN PROGRAMA. ....	29 -
3.1.5. FUNCIONES FUNDAMENTALES DE UN PROGRAMA DE ARDUINO. ....	29 -
3.1.6. ALTERNATIVAS. ....	30 -
3.1.7. BUSES DE COMUNICACIÓN. ....	37 -
3.2 SOFTWARE: PLATAFORMA THINGER.IO. ....	42 -
<b>CAPÍTULO 4. INSTRUMENTACIÓN. ....</b>	<b>57 -</b>
4.1 PLACA NODEMCU. ....	58 -
4.1.1. PLACA EMPLEADA EN LA EXPERIMENTACIÓN DEL TFG. ....	59 -
4.2 SENSORES. ....	61 -
4.2.1. SENSOR DHT11. ....	61 -
4.2.2. SENSOR DHT22. ....	63 -
4.2.3. SENSOR BME280. ....	64 -
4.2.4. RESISTENCIAS. ....	65 -
4.2.5. LIBRERÍAS EN EL ENTORNO DE ARDUINO. ....	67 -
<b>CAPÍTULO 5. PARTE EXPERIMENTAL Y RESULTADOS. ....</b>	<b>73 -</b>
5.1 SISTEMA DE ADQUISICIÓN DE DATOS. ....	74 -
5.2 PLANTEAMIENTO. ....	74 -
5.3 ESTADO INICIAL. ....	75 -
5.4 ETAPA DE AJUSTE. ....	77 -
5.5 CALIBRACIÓN. ....	83 -
5.6 ÚLTIMA ETAPA: ESTIMACIÓN DEL MENSURANDO TRAS LA CALIBRACIÓN. ....	89 -

<b>CAPÍTULO 6. ESTUDIO ECONÓMICO.</b> .....	<b>- 117 -</b>
6.1 RECURSOS UTILIZADOS. ....	- 118 -
6.2 COSTES DEL PROYECTO. ....	- 118 -
6.2.1. COSTES DIRECTOS. ....	- 118 -
6.2.2. COSTES INDIRECTOS. ....	- 121 -
6.3 COSTE TOTAL DEL PROYECTO. ....	- 122 -
<b>CAPÍTULO 7. CONCLUSIONES Y LÍNEAS FUTURAS.</b> .....	<b>- 123 -</b>
7.1 CONCLUSIONES. ....	- 124 -
7.2 LÍNEAS FUTURAS. ....	- 125 -
<b>CAPÍTULO 8. BIBLIOGRAFÍA</b> .....	<b>- 127 -</b>
<b>CAPÍTULO 9. ANEXOS</b> .....	<b>- 131 -</b>

## ÍNDICE DE FIGURAS

Figura 1. Historia de las Revoluciones Industriales.....	- 9 -
Figura 2. Secuenciación de las cuatro revoluciones industriales.....	- 9 -
Figura 3. Impresión 3D mediante Modelado por Deposición Fundida (FDM).....	- 13 -
Figura 4. Robot colaborativo (Cobot) en la planta de BMW en Spartanburg (EE.UU.) trabajando junto al personal del taller de producción.....	- 14 -
Figura 5. Conducción remota Tesla a través de una aplicación móvil (izqda.) y vista desde el interior del mismo mediante conducción autónoma (dcha.).....	- 17 -
Figura 6. Ejemplar de uno de los posibles informes que puede ofrecer la camiseta "smart = inteligente". .....	- 19 -
Figura 7. Camiseta Wearlumb junto con una tablet en la que se muestra la aplicación donde se registran y visualizan los datos. ....	- 19 -
Figura 8. Fotografías de la planta junto a los parámetros de configuración para su correcta monitorización en tiempo real en un dispositivo electrónico. ....	- 20 -
Figura 9. Digitalización de los cultivos a través de una plataforma para su gestión.....	- 21 -
Figura 10. Los pilares fundamentales en que se sustenta la infraestructura de la calidad. ....	- 22 -
Figura 11. Infraestructura Metrológica Española. ....	- 23 -
Figura 12. Esquema entorno de Arduino.....	- 28 -
Figura 13. Las dos partes fundamentales de un programa de Arduino. ....	- 29 -
Figura 14. Procesador ESP8266. ....	- 31 -
Figura 15. Circuito configuración modo Deep-Sleep en placa con sensor BME280. ....	- 33 -
Figura 16. Código de programa para el establecimiento de la conexión de la placa con la plataforma Thingier.io. incorporando el modo Deep-Sleep.....	- 36 -
Figura 17. Comparativa modo funcionamiento normal (izqda.) con el modo de ahorro Deep-Sleep (dcha.).....	- 37 -
Figura 18. Componentes conectados en paralelo a las dos líneas del bus, SDA y SCL. Conexión multi-master. ....	- 38 -
Figura 19. Ventajas e inconvenientes del bus I <sup>2</sup> C. ....	- 40 -
Figura 20. Líneas principales en un bus de datos SPI junto a las líneas adicionales (en amarillo)- 41 -	-
Figura 21. Configuración en cascada de los esclavos de un bus SPI.....	- 42 -
Figura 22. Ventajas e inconvenientes del bus SPI.....	- 42 -
Figura 23. Vista principal de acceso a la plataforma Thingier.io. ....	- 43 -
Figura 24. Pantalla principal de Thingier.io una vez funcionando uno de los dos dispositivos registrados.....	- 44 -
Figura 25. Circuito placa con sensor DHT22.....	- 45 -
Figura 26. Conexiones placa (dcha.) junto al sensor AM2302 DHT22 (izqda.) con sus pines correspondientes. ....	- 45 -
Figura 27. Instalación de la librería DHT, última versión.....	- 46 -
Figura 28. Datos al añadir un nuevo dispositivo para hacer esclava a la placa con la plataforma Thingier.io.....	- 46 -
Figura 29. Dispositivos conectados.....	- 47 -
Figura 30. Información relativa al dispositivo 1: IdNODEMCU1.....	- 47 -
Figura 31. Información relativa al dispositivo 2: IdNODEMCU2.....	- 48 -
Figura 32. Cuadro de trabajo de los Data Buckets disponibles en el momento (máx. 10).....	- 48 -
Figura 33. Exportación de datos a otro programa. ....	- 49 -
Figura 34. Dashboard tempBME280 activado el botón de la parte superior derecha [en verde]..- 50 -	-

Figura 35. Operaciones que se pueden insertar dentro de la pestaña Add Widget, WidgetSettings, Type.....	51 -
Figura 36. Diferentes gráficos o formas de visualizar los datos por pantalla en el área de trabajo: Dashboards. ....	51 -
Figura 37. Insignia de la aplicación thinger.io.....	52 -
Figura 38. Creación del Token de dispositivo. ....	52 -
Figura 39. Código QR que permite el acceso al dispositivo desde la aplicación de Android. ....	53 -
Figura 40. Diferentes formas de lectura de los dispositivos en la app. thinger.io desde el smartphone. ....	55 -
Figura 41. Identificación de las versiones V2 y V3 conforme a los chips, de izquierda a derecha respectivamente. ....	59 -
Figura 42. Placa NodeMCU v3- ESP8226 empleada para el desarrollo experimental, vista anterior (izqda.) y vista frontal (dcha). ....	60 -
Figura 43. Pines del sensor DHT11 sin la placa de circuito impreso, PCB. ....	62 -
Figura 44. Pines del sensor DHT11 con la PCB. ....	62 -
Figura 45. Sensor DHT22 empleado para la medida de temperatura y humedad relativa. ....	63 -
Figura 46. Sensor BME280 vista posterior (izqda.) y anterior (dcha).....	64 -
Figura 47. Configuración resistencias Pull-UP cuando el interruptor está accionado (izqda.) y cuando se encuentra abierto (dcha.). ....	66 -
Figura 48. Configuración resistencias Pull-DOWN cuando el interruptor está accionado (dcha.) y cuando se encuentra abierto (izqda.). ....	66 -
Figura 49. Resumen configuraciones resistencias Pull-DOWN y Pull-UP. ....	67 -
Figura 50. Descarga de la Librería Adafruit BME280 desde el Gestor de Librerías de Arduino. .-	69 -
Figura 51. Carga del programa del sensor BME280 a la placa. ....	69 -
Figura 52. Ejecución programa subido a la placa y su visualización a través del monitor serie. .-	70 -
Figura 53. Programa establecimiento conexión placa con sensor BME280 a la plataforma Thingier.io.....	71 -
Figura 54. Instalación dispositivo IdNodeMCU2. ....	71 -
Figura 55. Gráfico comparativo de la humedad relativa medida por los sensores respecto de la referencia (curva azul). ....	75 -
Figura 56. Gráfico comparativo de la temperatura medida por los sensores respecto de la referencia (curva azul). ....	76 -
Figura 57. Gráfico comparativo de la presión medida por el sensor BME280 respecto de la referencia (curva azul). ....	76 -
Figura 58. Resultados calibración del Termohigrómetro empleado referentes a la variable temperatura. ....	78 -
Figura 59. Resultados de calibración correspondientes a la variable humedad relativa. ....	79 -
Figura 60. Gráfico comparativo de la humedad relativa medida por los sensores respecto de la humedad relativa corregida del Laboratorio. ....	79 -
Figura 61. El ajuste existente entre la humedad relativa medida por el sensor DHT22 y la referencia corregida.....	80 -
Figura 62. El ajuste existente entre la humedad relativa medida por el sensor BME280 y la referencia corregida.....	80 -
Figura 63. Gráfico comparativo de temperaturas, expresadas en ( °C), una vez corregida la referencia, Laboratorio. ....	81 -
Figura 64. El ajuste existente entre la temperatura medida por el sensor DHT22 y la referencia corregida.....	81 -
Figura 65. El ajuste existente entre la temperatura medida por el sensor BME280 y la referencia corregida.....	81 -
Figura 66. Resultados de calibración correspondientes a la variable presión. ....	82 -



Figura 67. Gráfico comparativo de presión, expresada en (Pa), una vez corregida la referencia, Laboratorio. ....	83 -
Figura 68. El ajuste existente entre la presión medida por el sensor BME280 y la referencia corregida.....	83 -
Figura 69. Gráfica comparativa de la evolución temporal de la humedad relativa corregida respecto de la de cada uno de los sensores con que se realiza el estudio.....	84 -
Figura 70. Error existente entre la humedad relativa medida por el sensor DHT22 y la referencia, ambos corregidos.....	85 -
Figura 71. Error existente entre la humedad relativa medida por el sensor BME280 y la referencia, ambos corregidos.....	85 -
Figura 72. Gráfica comparativa de la evolución temporal de la temperatura corregida respecto de la de cada uno de los sensores con que se realiza el estudio. ....	86 -
Figura 73. Error existente entre la temperatura medida por el sensor DHT22 y la referencia, ambos corregidos.....	86 -
Figura 74. Error existente entre la temperatura medida por el sensor BME280 y la referencia, ambos corregidos.....	86 -
Figura 75. Gráfica comparativa de la evolución temporal de la presión corregida respecto de la del sensor BME280 corregido con que se realiza el estudio. ....	87 -
Figura 76. Error existente entre la presión medida por el sensor BME280 y la referencia, ambos corregidos.....	88 -
Figura 77. Error existente entre la temperatura del sensor DHT22 con respecto la referencia y sus delimitaciones. ....	91 -
Figura 78. Esquema del proceso de evaluación o tratamiento de la incertidumbre de medida..	92 -
Figura 79. Categorías en que se agrupa la incertidumbre, según el método de evaluación. ....	93 -
Figura 80. Esquema desviación típica estimada. ....	94 -
Figura 81. Tabla t-Student 95,45 % con los factores de seguridad o cobertura, k, correspondientes a sus grados de libertad $\nu_{ef}$ .....	96 -
Figura 82. Función de densidad de distribución Uniforme continua. ....	99 -
Figura 83. Costes totales del proyecto. ....	122 -

## ÍNDICE DE TABLAS

Tabla 1. Las tres versiones de la placa NodeMCU.....	- 59 -
Tabla 2. Correcciones y errores de las variables medidas por los sensores DHT22 y BME280....	- 88 -
Tabla 3. Hoja de cálculo Excel en que se calculan las correcciones de las lecturas de los sensores a partir de las correcciones establecidas.....	- 88 -
Tabla 4. Representación valores de los errores medios, $\bar{\epsilon}$ , (1ª fila) y desviaciones típicas , $[(S_i)]_{\text{error}}$ , (2ª fila) una vez corregidas las muestras. ....	- 89 -
Tabla 5. Muestra de datos con que se realizan las funciones modelo. ....	- 97 -
Tabla 6. Determinación de la incertidumbre típica combinada para la variable temperatura medida por el sensor DHT22 y sus gdl.....	- 101 -
Tabla 7. Determinación de la incertidumbre típica combinada para la variable temperatura medida por el sensor BME280 y sus gdl. ....	- 104 -
Tabla 8. Determinación de la incertidumbre típica combinada para la variable presión medida por el sensor BME280 y sus gdl. ....	- 107 -
Tabla 9. Determinación de la incertidumbre típica combinada para la variable humedad relativa medida por el sensor DHT22 y sus gdl. ....	- 110 -
Tabla 10. Determinación de la incertidumbre típica combinada para la variable humedad relativa medida por el sensor BME280 y sus gdl. ....	- 113 -
Tabla 11. Estimaciones finales de las variables climáticas estudiadas.....	- 114 -
Tabla 12. Determinación de los días hábiles al año.....	- 119 -
Tabla 13. Cálculo horas totales empleadas en la ejecución del proyecto. ....	- 119 -
Tabla 14. Coste total del material amortizable. ....	- 120 -
Tabla 15. Coste total del material no amortizable. ....	- 121 -
Tabla 16. Costes directos totales del proyecto. ....	- 121 -
Tabla 17. Costes indirectos totales del proyecto. ....	- 122 -
Tabla 18. Costes totales realización del proyecto. ....	- 122 -



## **CAPÍTULO 1.**

### **OBJETIVOS, ALCANCE Y ESTRUCTURA DEL TFG.**



## 1.1 ANTECEDENTES Y JUSTIFICACIÓN.

Hoy en día, en el marco general de las decisiones estratégicas de las operaciones de una empresa se acuña exponencialmente el concepto de "Industria 4.0", impregnando virtualmente la globalidad de los aspectos de la vida y permitiendo el desbloqueo de nuevos modelos de productos, de correspondencia con los clientes y nuevas prácticas y destrezas, desarrollando el potencial requerido.

El riesgo de obsolescencia junto con la pérdida de competitividad en el mercado son algunos de los indicios que supone la rápida integración por parte de las empresas en esta nueva era de la Industria. Este avance va más allá de los medios tecnológicos de los que se disponen actualmente, permitiendo la Industria 4.0 la digitalización de las fábricas, así como creando un vínculo extremadamente estrecho entre el ser humano y el medio productivo, dotando, de este modo, de inteligencia y valor añadido a la cadena productiva.

Son diversas las técnicas y/o metodologías para lograr esta fusión entre la persona y el medio de producción, encerrando todas ellas una serie de ventajas generales como son el aumento y un mayor control de la productividad, reducción de costes, mayor eficiencia y calidad, adaptación en un menor tiempo a la demanda variable, flexibilidad en la producción, etc. No obstante, también supone determinadas desventajas como una mayor inversión inicial, requerimiento de un alto nivel de cualificación de la plantilla, entre otras.

Según un estudio llevado por PwC [12], gracias al ingreso de la digitalización en las empresas los costes se verán reducidos en un 19,4 % con un consiguiente aumento de los ingresos de un 11,1 % en un corto/medio plazo de tiempo.

Son un gran número de sectores en los que se está implantando la IA (Inteligencia Artificial) y esta tendencia se encuentra en continua proliferación, algunos de los cuales son: la salud, donde los robots son una herramienta muy fructífera y permiten paliar ciertas limitaciones con que se encuentran los profesionales médicos, añadiendo funciones fuera del alcance de las personas, y también dentro de este ámbito, la impresión 3D resulta muy útil para elaboración de prótesis e implantes, adaptándose al paciente de forma más personalizada y reduciendo los costes que supondría el realizarlos con otros materiales más caros. En la agricultura, se están desarrollando equipos



que permiten el control remoto del cultivo, pudiendo obtener en tiempo real datos como la fecha en que se prevé una mejor recolección, la necesidad de riego, todo ello a través de sensores que a través de un hardware o un dispositivo digital hagan llegar al trabajador toda esta información. Por último, que no menos importante que los anteriores, se citará el sector de la automoción, donde cada día son mayores los avances en robótica colaborativa y en conducción autónoma, situándose en uno de los sectores pioneros en que se está implantando e innovando en Industria 4.0.

En este TFG se va a poner de manifiesto la relevancia que tienen dispositivos muy asequibles en términos económicos para el usuario, que con cierto control sobre ellos y por medio de programación a través de una plataforma de código abierto como es Arduino se puede llevar a cabo el control remoto de variables, en nuestro caso climáticas, dentro un sector industrial o cotidiano para la posterior toma de decisiones, todo ello permitiendo asegurar una correcta trazabilidad. Será a través del empleo de sensores junto con placas programables los que permitirán recabar la información necesaria para su posterior análisis, permitiendo no solo la lectura de los datos, sino posibilitando la interoperabilidad con los mismos, pudiendo actuar enviándolos ciertas órdenes para lograr satisfacer las necesidades del usuario.

## **1.2 OBJETIVOS.**

El objetivo principal de este TFG será poner de manifiesto la importancia que tiene la implantación de las técnicas que conforman la Industria 4.0 con el fin de ofrecer una alternativa a la necesidad de uso que establece la plataforma de Arduino de estar conectado con una placa a través de un USB para mantener la conexión con la misma, revelando como posible solución una plataforma de libre acceso, Thingier.io, que permita la conexión y operatividad con la placa vía WiFi. Para ello nos adentraremos particularmente en la Metrología, mostrando la puesta a punto y evaluación de la confiabilidad de dos tipos de sensores que miden variables climáticas.

Se realizará un estudio comparativo de dos tipos de sensores diferentes que proporcionan datos de variables climáticas, como son la temperatura, la humedad relativa y la presión, las cuales resultarán de gran influencia en una cantidad masiva de procesos productivos, por lo que el control remoto sobre las mismas supone grandes posibilidades a la industria, permitiendo una optimización de los recursos, así como la interferencia sobre los parámetros

que resulten críticos para mejorar en técnicas preventivas que minimicen los trances o fallos que puedan derivar de los mismos.

La sensorización a su vez permitirá conocer en qué instante habrá que intervenir en la línea de producción para obtener un mayor rendimiento.

También podrá comprobarse cómo por medio de la programación, una vez hayan sido calibrados los equipos de medida, disponiendo de datos procedentes de una referencia patrón, con la introducción de ciertos parámetros se podrá alcanzar una medida muy próxima a la real, con una incertidumbre reducida, dentro de un rango permisible; aportando una evidencia cuantitativa de la medida de cada una de las variables recogidas para una misma muestra, expresando el resultado de la misma a través de dos componentes, la estimación y su incertidumbre expandida.

El tratamiento de los datos exportados de la plataforma que se empleará para la lectura y representación de los mismos, facilitará la toma de decisiones a posteriori y la cuantificación de la incertidumbre que estos dispositivos comportan.

### **1.3 ESTRUCTURA DEL TRABAJO.**

Esta memoria está organizada en 7 capítulos, bibliografía y anexos. La primera parte consta de nociones teóricas para ofrecer al lector conocimientos técnicos acerca del software y hardware empleado en el TFG, de manera que le resulte la lectura de los capítulos subsiguientes más entendible y asequible.

Se presenta a continuación la estructura del proyecto especificando mediante una breve reseña lo que se abordará en cada uno de los capítulos:

- **CAPÍTULO 1.** Se muestra la motivación y justificación del proyecto junto con los objetivos del mismo y una estructura concisa de la memoria.

- **CAPÍTULO 2.** Presentación del concepto Industria 4.0 aportando ejemplos que resulten útiles para su comprensión y terminología clave para su desarrollo. También se presentará la Metrología 4.0 como campo de aplicación dentro de la ya mencionada Industria 4.0.

- **CAPÍTULO 3.** Se presenta el entorno de programación: el hardware y el software empleados en el desarrollo del proyecto para el control de los sensores utilizados y la caracterización de los parámetros medidos.



- **CAPÍTULO 4.** Se desarrolla el material empleado para la ejecución del TFG.
- **CAPÍTULO 5.** En este capítulo se expondrá el análisis del comportamiento de los sensores en la medición de las variables climáticas que se estudian y se presentan los resultados.
- **CAPÍTULO 6.** Se realiza el presupuesto del proyecto mediante el análisis de costes directos e indirectos.
- **CAPÍTULO 7.** Conclusiones derivadas del proyecto y líneas futuras abiertas para posteriores investigaciones.
- **CAPÍTULO 8.** En la Bibliografía se detallan los libros, documentos científicos y páginas web consultadas a lo largo del desarrollo de esta memoria.
- **CAPÍTULO 9.** En los anexos se encontrarán los programas ejecutados en el hardware empleado: Arduino.



Universidad de Valladolid

IMPLANTACIÓN, ESTUDIO Y CONTROL DE CALIDAD DE VARIABLES  
CLIMÁTICAS EN EL ÁMBITO DE LA METROLOGÍA 4.0.  
Capítulo 1. Objetivos, alcance y estructura del TFG.



ESCUELA DE INGENIERÍAS  
INDUSTRIALES





## **CAPÍTULO 2.**

# **INTRODUCCIÓN A LA INDUSTRIA 4.0 E IMPLEMENTACIÓN EN LA METROLOGÍA.**

Conforme han ido transcurriendo el paso de los años, la industria ha ido experimentando una serie de innovaciones y cambios, los cuales han derivado en una revolución industrial. Hasta el momento presente se pueden diferenciar cuatro revoluciones industriales, encontrándonos en la era de la ya notable Industria 4.0.

## **2.1 CRONOLOGÍA REVOLUCIONES INDUSTRIALES.**

En este epígrafe se mostrará la evolución histórica de las cuatro revoluciones industriales, detallando de cada una de ellas los rasgos más representativos:

### **1.0 PRIMERA REVOLUCIÓN INDUSTRIAL:**

Incorporación de la tracción hidráulica y máquinas de vapor en sistemas de producción mecánicos. Mecanización de la producción del sector textil.

**1774:** Construcción de la máquina de vapor mejorada por Matthew Boulton y James Watt.

**1784:** Primer telar mecánico.

**1829:** Locomotora de vapor. Primera línea férrea del mundo entre dos ciudades: Liverpool y Manchester.

### **2.0 SEGUNDA REVOLUCIÓN INDUSTRIAL:**

División del trabajo de producción, producción en serie y empleo de sistemas eléctricos. Desarrollo de las industrias química, eléctrica y automovilística.

**1870:** Primera cinta transportadora o línea de ensamblaje.

**1871:** Primera central eléctrica de uso comercial.

**1880:** Thomas Edison patenta la primera bombilla comercialmente factible.

**1886:** Primer automóvil de combustión interna patentado.

**1897:** Primera transmisión de radio, que dio origen a la industria de la radiotelegrafía.

### **3.0 TERCERA REVOLUCIÓN INDUSTRIAL:**

Integración de microelectrónica y tecnología de la información (TI) para la proliferación de la automatización en la producción. Reducción de tiempos y deslocalización de la producción.

**1962:** Primeros ordenadores personales por la empresa italiana Olivetti.

**1969:** Primer controlador programable (PLC) o autómeta, utilizado para regular y controlar la producción.

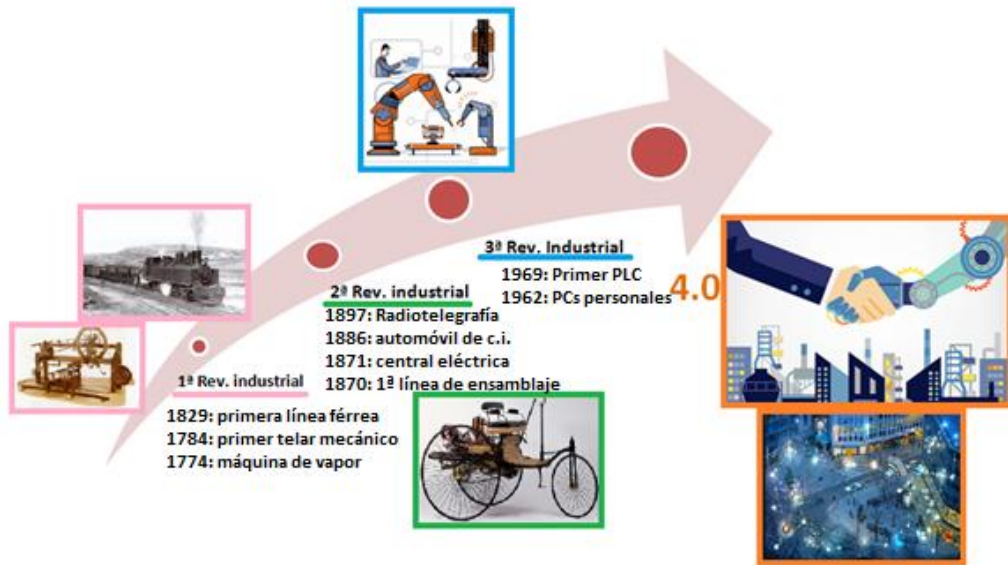


Figura 1. Historia de las Revoluciones Industriales. Fuente: Elaboración propia.

#### 4.0 CUARTA REVOLUCIÓN INDUSTRIAL:

Establecimiento de la interconexión entre elementos inteligentes a lo largo del proceso productivo. Surgimiento de la impresión 3D y robots colaborativos. Digitalización de los medios productivos. Posibilidad de monitorización del proceso de forma remota. Hiperconectividad. Big Data, Internet de las cosas (IoT).

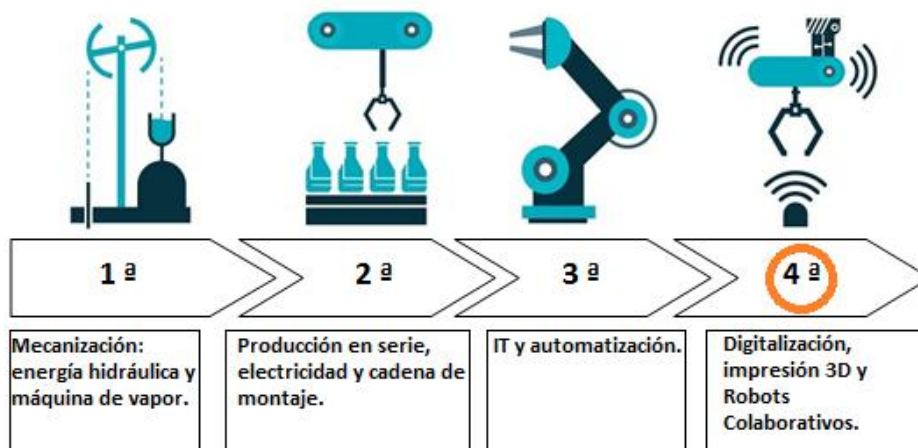


Figura 2. Secuenciación de las cuatro revoluciones industriales. Fuente: Elaboración propia.

## 2.2 INTRODUCCIÓN A LA TRANSFORMACIÓN DIGITAL O INDUSTRIA 4.0.

La **Industria 4.0** hace referencia al apego a la automatización e intercambio de datos en la tecnología de la fabricación, lo que da lugar a la **Cuarta Revolución Industrial**, llamada también **Industria Inteligente**, **Industria Interconectada**, entre otros. Este nuevo concepto surgió en Alemania.

Fomenta la informatización y **digitalización de los medios de producción y fabricación**, a través del intercambio y transformaciones de grandes cantidades de datos [Big Data], desarrollados y tratados conforme una serie de algoritmos que conducen a un control y organización óptimos de los medios de producción. Puede ser entendida como una **fusión del mundo virtual y el real**.

Permite lograr una conexión total entre **el usuario** y los medios de producción, pudiendo incidir e **intervenir en tiempo real** a lo largo de la cadena de producción, para llegar a un aprovechamiento eficaz de los recursos logrando satisfacer las necesidades requeridas por el cliente. Todo ello se consigue mediante la innovación e integración de diferentes tecnologías capaces de procesar datos, el uso de sensores y software inteligente que permitan controlar la producción añadiendo valor a la misma.

Se pretende **personalizar la producción** a la medida del proveedor y cliente con el fin de una **mayor satisfacción en el cliente**, todo ello rebajando a su vez el coste energético invertido en el proceso de fabricación. Es por ello que a través de una misma tecnología permite unir simultáneamente las dos líneas progresivas del ser humano: el sistema productivo y la puesta en práctica del conocimiento con un consiguiente giro tanto en la organización como en la gestión de la cadena de valor de las factorías.

Multitud de empresas de infinidad de sectores están fomentando su auge gracias a la inversión en instrumentos que contribuyan a la proliferación e implantación de la Industria 4.0, **a lo largo de la segunda década del s. XXI**. En resumidas cuentas, es útil para interconectar la totalidad de las partes que conforman una empresa para lograr una **automatización efectiva e inteligente y una alta personalización de los medios productivos**.

Este atractivo avance en la Industria tiene un impacto muy positivo en diversos sectores, tales como:

- La **monitorización de riesgos**: de esta forma se avala la seguridad en el transporte de materias peligrosas.

- **Mantenimiento predictivo:** ofrece la posibilidad de pronosticar el correcto funcionamiento de la maquinaria y útiles usados a lo largo de un sistema productivo, por medio de sensores para garantizar una mayor eficiencia y evitar posibles percances.

En un corto plazo de tiempo las empresas deberán implantar la transformación digital para garantizar la competencia en el sector industrial, si no quieren verse obsoletas con el paso del tiempo. La transformación digital trae consigo unas notables oportunidades de negocio, una posible mejora en la gestión de los posibles riesgos y un valor añadido a la producción, todas ellas como finalidades de la implantación de dicha técnica citada.

En conclusión, la Industria 4.0 consiste en la integración de tecnologías de la información en el sistema productivo de modo que se consiga una mejor asignación de recursos y una reducción en los costes.

### 2.3 CONCEPTOS CLAVES EN LA INDUSTRIA 4.0

**API [Application Programming Interface]** (Interfaz de programación de aplicaciones): grupo de comandos, funciones y protocolos informáticos que ofrece una determinada biblioteca para poder ser utilizada por otro programa informático, sin necesidad de programar desde cero. La facilidad de integración de la misma es un rasgo que la caracteriza, posibilitando la interacción entre aplicaciones para intercambiar datos, de modo que se puedan aprovechar los servicios que ofrezca una aplicación en otra.

La reducción de costes y tiempo invertidos en el desarrollo de nuevas aplicaciones junto a la interoperabilidad entre distintas aplicaciones, son algunas de las ventajas que ofrecen las API, gracias a que el código pueda ser reutilizado.

**Big Data:** cantidad colosal de datos que se escapa su manejo a través de la tecnología y medios tradicionales y que debe ser abordada, manipulada, tratada y analizada por medio de herramientas específicas que permitan la detección, administración y procesamiento de estos datos con una baja latencia. El IoT (Internet of Things) y la IA (Inteligencia Artificial) son algunos de los fenómenos que han contribuido a la intensificación de la complejidad de abordar las innumerables fuentes y diversas formas de gestionar los datos

que proporcionan, que por lo general son generados en tiempo real y a gran escala.

No obstante, este término, no es únicamente utilizado para hacer referencia al gran volumen de datos, sino también se hace uso de él para referirse a la técnica de análisis y manejo de datos (procedimientos y herramientas), la cual se ha asentado hoy en día con gran aceptación ya que ayuda a mejorar las operaciones y la funcionalidad de una empresa, pudiendo tomar decisiones estratégicas de forma más rápida, eficiente e inteligente.

**Ciberseguridad:** también conocida como seguridad informática, es el área que se centra en ofrecer la garantía de seguridad y solidez de la red informática y lo vinculado con ella.

**Computación en la Nube (Cloud Computing):** se refiere al espacio accesible a los datos por parte del usuario a través de Internet.

La computación en la nube se fundamenta en compartir recursos y de este modo alcanzar grandes economías de escala de forma congruente.

La nube se ha implantado hoy en día como solución al análisis de cantidades voluminosas de datos, agilizando así la toma de decisiones dentro del marco que se esté analizando, bien un proceso productivo en el área industrial, entre otros. Es decir, con la disponibilidad de los datos en la nube, se posibilita el acceso inmediato a los datos, para así predecir posibles situaciones pudiendo a su vez aminorar los costes.

**Hiperconectividad:** término que hace alusión a la situación en que está inmersa la sociedad, en la que los seres humanos viven permanentemente conectados a través de diversos dispositivos, permitiendo así el acceso a la información actualizada en tiempo remoto.

**Impresión 3D:** es uno de los cimientos de la transformación industrial y su desarrollo está en pleno auge. Permite obtener, de forma rápida y a bajo coste, prototipos a partir del modelo programado y a continuación comprobar su efectividad, proporcionando un indiscutible servicio. Se obtienen objetos tridimensionales gracias a la fabricación por adición.

Los materiales más usados por las impresoras 3D suelen ser los polímeros termoplásticos **PLA** (Poliácido láctico) o **ABS** (Acrilonitrilo butadieno estireno) empleando para ello, generalmente, la técnica **FDM** (Fused Deposition

Modeling), "Modelo por Deposición Fundida", la cual consiste en hacer pasar un filamento de un rollo de material plástico por la boquilla del extrusor, donde se calienta y va siendo depositado capa a capa (por adición) y posteriormente solidifica el material fundido sobre la placa/cama de impresión obteniéndose el objeto que se desee.

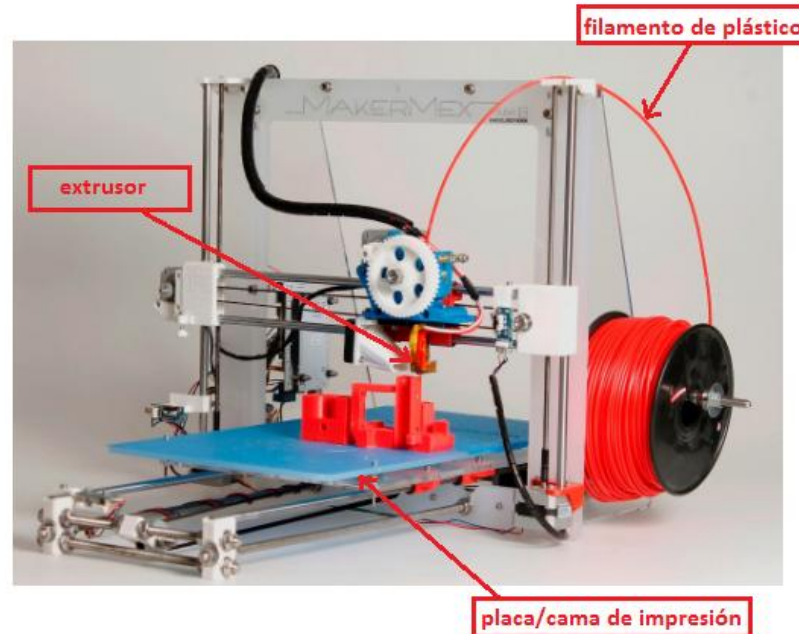


Figura 3. Impresión 3D mediante Modelado por Deposición Fundida (FDM). Modelo: PRUSA i3 & i3 XL MAKERMEX.

No obstante, existen otros métodos de impresión 3D, menos comúnmente empleados, principalmente por su alto precio y la limitación de colores, respecto del FDM que era bastante económico:

- La "Estereolitografía" (**SLA**): en ella se usa un haz de luz ultravioleta que se proyecta en una resina fotosensible, para que ésta vaya solidificándose.
- Selective Laser Sintering (**SLS**), "Sinterización Selectiva por Láser": en esta, lo que se proyecta es un haz de láser de mayor potencia que la anterior para que funda el material que ahora se trata de polvo cerámico, de cristal, nylon o poliestireno.

**Inteligencia Artificial (IA):** Este término apareció en 1956 y se refiere a la tecnología implantada a la empresa que sitúa al usuario en el centro del proceso productivo, confiriéndole una gran personalización, de forma remota. Los procesos industriales serán prácticamente automatizables en todo su dominio, adoptando así las máquinas facultades como captar, comprender y



emprender acciones ante estímulos con una mayor capacidad de procesamiento.

**Internet of Things (IoT):** es un concepto propio del área de la informática que explica la relación existente entre los objetos físicos cotidianos conectados a la red o bien con otros dispositivos. Permite gestionar y analizar una cantidad masiva de datos. Describe la fusión entre el mundo físico y el mundo virtual de forma inteligente. Se trata de la base que sustenta la Industria 4.0 así como la potencia que implanta la Revolución Industrial en la época actual.

**La Robótica Colaborativa, (Cobot):** se trata del último de los adelantos de la robótica y su función elemental es la de crear robots que interactúen físicamente con el ser humano en un mismo área de trabajo, contribuyendo a optimizar la productividad de una empresa. Se les confiere de este modo a los robots la posibilidad de percibir, almacenar, razonar y planificar, para así adaptarse a la variable demanda productiva de forma instantánea por medio de la sensorización lo que proporciona adaptabilidad a la automatización.

También se consigue mejorar la precisión, ya que una máquina programable puede alcanzar mucha mayor precisión que una tarea realizada por el ser humano. Una ventaja de los Cobots (Robots Colaborativos) frente a los Robots tradicionales, es que llevan incorporados sistemas de IA y normas que rigen su comportamiento frente a diversas actuaciones cuando se encuentran trabajando al lado de las personas. Estas normas son aplicables en función de las señales recibidas por los sensores que por medio de los sistemas inteligentes, obligarán que actúen de un modo u otro, reduciendo así la incertidumbre en que están embebidos y garantizando la seguridad en el entorno circundante.

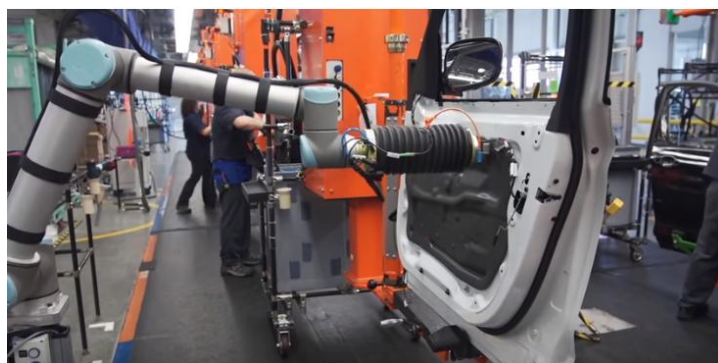


Figura 4. Robot colaborativo (Cobot) en la planta de BMW en Spartanburg (EE.UU.) trabajando junto al personal del taller de producción. [13]



**Realidad Aumentada y Realidad Virtual:** La Realidad Aumentada (RA) es una herramienta que posibilita al usuario visualizar parte de lo que ve en la realidad en un aparato, añadiendo información virtual a la realidad en tiempo real, con el fin de buscar la perfección del mundo real a gusto del usuario.

Por el contrario, la Realidad Virtual (RV) en vez de usar como soporte el mundo real, como ocurría en la realidad aumentada, lo que hace es sumergirte en un entorno completamente virtual, alejado de la realidad. También cabe señalar que la realidad virtual exige de un elemento externo para poder observarla, como bien pueden ser gafas, cascos u otro medio, sin embargo, la realidad aumentada únicamente requerirá de una aplicación en el móvil o bien de otro dispositivo tecnológico. Las gafas de realidad virtual dispondrán de unos sensores que acatan el movimiento de la cabeza, de forma que al realizar cualquier movimiento con la misma, se lleve a cabo el mismo movimiento dentro del mundo virtual en que estés inmerso en ese instante.

## 2.4 VENTAJAS E INCONVENIENTES DE LA INDUSTRIA 4.0.

La **Industria 4.0** reúne una serie de **VENTAJAS**:

- Una versatilidad inherente frente a la demanda.
- Personalización de los productos y servicios en función de las necesidades del usuario.
- Producto de valor añadido.
- Un análisis de información y explotación de la misma en tiempo real para una mejora continua y mayor confiabilidad. Mejora de los pronósticos y decisiones.
- Conectividad.
- Optimización de los niveles de calidad en la producción con una reducción notoria de los costes.
- Mayor seguridad al personal implicado en la cadena de valor.
- Reducción del tiempo de producción y tiempos inactivos.
- Reducción de los tiempos de reacción y toma de decisiones, gracias a la mayor eficiencia en el flujo de datos e información.
- Creación de fábricas predictivas, proactivas y competitivas.
- Trabajo seguro entre robots y humanos de forma síncrona.

Pese a ello, conlleva también una serie de **INCONVENIENTES**, citados a continuación:

- Riesgo de obsolescencia de compañías que no dispongan de suficientes recursos para hacer frente a la exigencia de avances tecnológicos.
- Mayor cualificación en los trabajadores y en algunos casos insuficiencia para abordar la remuneración requerida por los perfiles de las personas a llevar a cabo ciertas funciones dentro de la producción de la fábrica.
- Mayor coste de inversión, sobre todo en la etapa inicial, respecto a los medios tradicionales.
- Brecha de competencia por la gran proliferación tecnológica.
- Podría aparecer una barrera a la hora del mantenimiento de la integridad del proceso productivo ya que hay una menor inspección humana directa.

## **2.5 EJEMPLOS DE IMPLANTACIÓN DE LA INDUSTRIA 4.0.**

### **2.5.1. EJEMPLO 1. TESLA (AUTOMOCIÓN).**

Uno de los sectores en que se pueden encontrar grandes avances gracias a la implementación de la Industria 4.0, la cual engloba el IoT [Internet of Things], tecnología e impresión 3D, entre otras, es el sector automovilístico. Concretamente se menciona, a continuación, el modelo de coche Tesla, el cual ya en el año 2017 mostró indicios de cómo había sido descubierta la posibilidad de controlar de forma remota el funcionamiento de este coche. Se estimó a partir de determinados estudios que en 2021 el 82% de los coches vendidos habrían adoptado esta técnica del IoT en sus sistemas, aún teniendo presente el riesgo que la conectividad presenta en la seguridad.

La puesta en práctica del IoT en el modelo Tesla ha consolidado en la sociedad la posibilidad de transmitir información desde el propio auto hacia el exterior, donde el propio cliente puede manejar o bien tomar decisiones del propio automóvil a distancia. Este coche está provisto de una API [Application Programming Interface], es decir, un software que permite la interacción o comunicación con otro por medio de funciones predefinidas. Cuenta también con una serie de sensores que permiten la conducción monitorizada, rastreando el entorno más cercano y detectando posibles obstáculos, algunos tan pequeños como por ejemplo un llavero que estuviese situado a una distancia próxima, activando así su sistema de frenado como medida de seguridad.



Figura 5. Conducción remota Tesla a través de una aplicación móvil (izqda.) y vista desde el interior del mismo mediante conducción autónoma (dcha.). [14]

El automóvil puede garantizar de forma instantánea información sobre su geolocalización, así como del acelerómetro, entre otros. A su vez, dispone de cámara para vista trasera y los ya mencionados sensores que informan de forma remota sobre la situación del vehículo. Los sistemas de digitalización del vehículo informan de otros aspectos como pueden ser la presión de las ruedas, configuración de los espejos, nivel del aceite, el estado de la batería, el estado de carga ,... , en resumen, aborda la práctica totalidad de la funcionalidad del automóvil. Es a través de aplicaciones de móvil desde las que se puede dirigir el control del mismo.

Lleva instalado una consola de unas 17" [15] para controlar todas las funciones que conforman el coche, lo que supone un gran avance sobre el coche convencional. El sistema de carga inteligente que lleva integrado puede recibir corriente de cualquier estación eléctrica. Y, por medio de la telemetría se puede mejorar su mantenimiento lo cual le confiere idoneidad y confiabilidad.

El mantenimiento de éste está muy restringido, pues no hay que cambiarle el aceite, no obstante, los sistemas hidráulicos para la dirección, el frenado, el aire acondicionado y la suspensión pueden precisar de mantenimiento como ocurre con el automóvil convencional, pero a diferencia de éste, dispone de un sistema de frenado regenerativo que reduce la necesidad de reemplazar los frenos. El apoyo de la configuración inteligente hace posible sistemas de reducción de energía, así como poder acceder al consumo que va a tener todo trayecto que se ejecute. Toda esta nube de información que proporciona es para optimizar el rendimiento del mismo, con un buen manejo sobre él. La batería puede ser reemplazada en minuto y medio por una completamente cargada, y, por otro lado, las actualizaciones del software tienen lugar gracias a una conexión inalámbrica.

La versión más robusta de este modelo de coches, es la P100D con una batería de 100 kWh que permite acelerar de 0-100 km/h en tan solo unos



2.4 s. Tesla ha garantizado la circulación autónoma, es decir la capacitancia de sus vehículos para viajar sin conductor con una alta confiabilidad. Sin embargo, la legislación en España, aún no ha aprobado su uso. Deberán incorporar ciertos programas que se adelanten o anticipen a cualquier situación de riesgo para ser evitada y la reacción se producirá en un rango de tiempo inferior al que la capacidad humana permitiría en la misma situación.

### **2.5.2. EJEMPLO 2. WEARLUMB.**

Se trata de una camiseta inteligente en la que se integra el **IoT** [Internet of Things] para poder realizar un análisis de los posibles riesgos posturales en el ámbito laboral y poder hacerlos frente tomando ciertas acciones en función de los datos recabados. Lleva incorporada para la medición de los datos de movimientos, una serie de sensores impresos, en la espalda, que recogen la información y posteriormente ésta es enviada mediante Bluetooth a una aplicación, en la cual se podrá llevar a cabo el análisis en tiempo real, para paliar estos riesgos de fatiga, en especial la lumbalgia, ocasionados como consecuencia de posturas incorrectas en el propio trabajo.

En la aplicación se puede examinar la postura y, la misma ofrece una serie de recomendaciones a adoptar frente a tal situación mostrada.

Es posible obtener lecturas de la elongación y tensión de los músculos, así como de la velocidad con la que se efectúan los movimientos.



Figura 6. Ejemplar de uno de los posibles informes que puede ofrecer la camiseta "smart = inteligente". [17]

Se propone como una medida correctora con las consiguientes mejoras de la ergonomía en el trabajo y es diseñada no para llevarla consigo durante toda la jornada laboral, sino en ciertos períodos, los suficientes y necesarios como para obtener una muestra de datos que garantice un posible estudio exhaustivo y conlleve a una toma de decisiones correctoras lo más satisfactorias posibles.



Figura 7. Camiseta Wearlumb junto con una tablet en la que se muestra la aplicación donde se registran y visualizan los datos. [16]

Como se ha podido ver se trata de un ejemplo claro del concepto "Wearables", que hace referencia a una serie de dispositivos electrónicos que se incorporan a alguna de las partes del cuerpo humano, y son los encargados de la digitalización de aspectos físicos para su posterior tratamiento y análisis.

### 2.5.3. EJEMPLO 3. INTRACE (SECTOR INDUSTRIAL).

Con este nuevo avance basado en la Industria 4.0 se consigue un seguimiento inteligente en entornos industriales para los que ha sido diseñado. La función para la cual fue puesto en práctica fue la de monitorear el rendimiento de una planta industrial aprovechando la tecnología del IoT, a un coste relativamente bajo.

Este invento trajo consigo la posibilidad de llevar a cabo un seguimiento minucioso del rendimiento de la planta, personal de trabajo, máquinas, transporte en la misma, herramientas, etc., para así poder tomar decisiones más satisfactorias pudiendo comprobar el estado del funcionamiento a partir de la base de datos en tiempo real de la situación de la planta, contribuyendo de este modo a una proliferación de oportunidades para la mejora de la misma. Con todo ello se permite una producción más eficiente.



Figura 8. Fotografías de la planta junto a los parámetros de configuración para su correcta monitorización en tiempo real en un dispositivo electrónico. [18]

Unas de las acciones que permite realizar son las siguientes: determinación de las oportunidades de mejora y formación de cuellos de botella con una adecuada trazabilidad. Es a través de los Dashboards donde se puede visualizar gráficamente la información recibida por el dispositivo. Cuenta, a mayores, con una serie de alarmas personalizables y programables que informan de ciertas incidencias en la cadena productiva y el medio en que se encuentra inmersa. Pero, el papel fundamental lo desempeñan los sensores que permiten detectar localización, vibraciones, movimientos, es decir, toda y cuanta información se proporcionará al dispositivo para su posterior análisis.

Por último, señalar aunque era de pensar que la planta deberá contar con una interconexión tanto vertical como horizontal para la recogida de los datos.

#### **2.5.4. EJEMPLO 4. MONITORIZACIÓN DE CULTIVOS (SECTOR AGRÍCOLA).**

Se ha desarrollado una plataforma para el control, análisis y recogida de datos de los cultivos. Ésta incluye como funciones la monitorización de la condición en que se encuentre el fruto permitiendo que la cosecha sea en el momento más idóneo para obtener una mejor producción, predecir la productividad desde las primeras etapas de su curso y ver qué situaciones de plagas o enfermedades puedan tener lugar.

Es por medio de sensores, imágenes hiperespectrales y el Internet de las cosas (IoT) junto a la técnica del Big Data lo que hace posible la manipulación de los datos en la nube. Conforme a una serie de parámetros como son el nivel del suelo, características físico-químicas y otras se incluirá en uno de los modelos de los que se incluyen en la plataforma y se tomarán las decisiones oportunas según lo que se haya establecido en la plataforma para el modelo en cuestión.

Una de las ventajas que dará lugar será el uso de menos productos fitosanitarios, como pesticidas y herbicidas, contribuyendo a su vez a mantener la calidad del producto sin que se vea menoscabado y una mejora de la productividad agraria.



Figura 9. Digitalización de los cultivos a través de una plataforma para su gestión. [19]

## **2.6 METROLOGÍA INDUSTRIAL.**

Como se citó al comienzo de esta memoria, se trataba de exponer un caso particularizado de la Industria 4.0 aplicado al ámbito de la Metrología 4.0.

### **2.6.1 LA METROLOGÍA -> METROLOGÍA INDUSTRIAL.**

La **metrología** es la ciencia de las medidas, entendidas éstas junto con la evaluación de sus correspondientes incertidumbres. Se trata de uno de los

pilares que sustentan en el país el sistema de calidad y competitividad en el sector industrial, junto a los restantes, normalización, acreditación y certificación.



Figura 10. Los pilares fundamentales en que se sustenta la infraestructura de la calidad. Fuente: Elaboración propia.

- **NORMALIZACIÓN**: Incorporación de la normativa vigente en materia de medidas, en España AENOR [Asociación Española de Normalización y Certificación]. Las normas no han de ser de estricto cumplimiento, pero si se introducen en el Ordenamiento jurídico sí adquirirán carácter de obligatoriedad.
- **ACREDITACIÓN**: Se audita el trabajo laboral del operador en el momento que lo realiza, es decir, que lo que haga éste lo haga adecuadamente, proporcionando así competencia técnica.
- **CERTIFICACIÓN**: Hace alusión a la garantía que le proporciona la realización de auditorías en la empresa, para que se le autentique que cumple ciertamente la norma.

Según el profesor D. Carlos Granados sería:

"La ciencia que tiene por objeto el estudio de las propiedades medibles, las escalas de medida, los sistemas de unidades, los métodos y técnicas de medición, así como la evaluación de lo anterior, la valoración de la calidad de las mediciones y su mejora constante, facilitando el progreso científico, el desarrollo tecnológico, el bienestar social y la calidad de vida".

Hay tres tipos de Metrología: **Científica, Legal e Industrial.**



### - Metrología Científica:

Es el campo que desempeña las labores de desarrollo y organización de los patrones de medida y de su continuo mantenimiento. Esta tarea será realizada por el Centro Español de Metrología y sus Laboratorios asociados para garantizar una infraestructura metrológica que garantice la vida cotidiana de la sociedad aportando medidas con gran solidez y confiabilidad, véase la Figura 11.

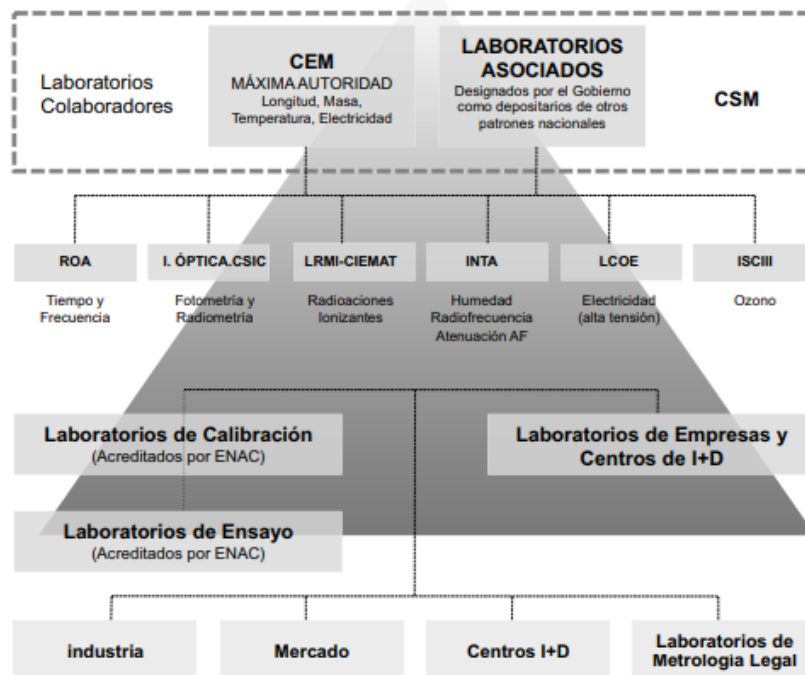


Figura 11. Infraestructura Metrológica Española. Fuente: Centro Español de Metrología.

Los laboratorios asociados son: **ROA** (Real Observatorio de la Armada), **IO-CSIC** (Instituto de Óptica), **CIEMAT** (Centro de Investigaciones Energéticas, Medioambientales y Tecnológicas) **INTA** (Instituto Nacional de Técnica Aeroespacial), **LCOE** (Laboratorio Central Oficial de Electrotecnia) e **ISCIII**: Instituto de Salud Carlos III. Habría que sumar a estos laboratorios asociados unos Laboratorios Colaboradores, como aparece reflejado en la cúspide piramidal de la metrología española.

### - Metrología Legal:

Es la rama de la Metrología encargada de cotejar y examinar el reglamento legal por el que se rigen los instrumentos, unidades, medidas, métodos y procedimientos de medición para conseguir una clara transparencia en las

transacciones comerciales, en el terreno de la salud o en el ámbito de la seguridad, tanto para los usuarios como para el medio ambiente.

El artículo 149.1.12 de la Constitución Española otorga competencia exclusiva al Estado de legislar sobre pesas y medidas y determinación de la hora oficial; siendo el organismo responsable de este cometido el CEM [Centro Español de Metrología].

Sin embargo, la ejecución del control metrológico del Estado mora en las Comunidades Autónomas.

Por otro lado, las administraciones públicas son las responsables en materia de regularidad legal, sustentándose en la acreditación de ENAC [Entidad Nacional de Acreditación] para avalar la competencia técnica.

#### - **Metrología Industrial:**

La Metrología Industrial o Metrología Técnica, es la más afín a la labor del ingeniero. Se ocupa del uso de las medidas dentro del control y verificación de calidad y de la fabricación para conferir confiabilidad a las medidas que se usan en la industria centrándose principalmente en los sistemas de gestión de calidad aplicados a la fabricación (normas ISO 9000).

Sin mediciones exactas y precisas no hay calidad. Es por ello que las empresas deben poseer una infraestructura metrológica para conocer y probar el grado de conformidad que tienen sus productos para ratificar que los productos que comercializan se ajustan a las especificaciones, consiguiendo con ello el fin último de satisfacer las necesidades del cliente.

La Metrología Industrial es voluntaria, la gestión del control de calidad de lo que se está haciendo, en cambio la Metrología Legal sí es obligatoria, de no realizarse no se podría comercializar el producto.



## **CAPÍTULO 3.**

# **IMPLEMENTACIÓN INDUSTRIA 4.0: ARDUINO Y PLATAFORMA THINGER.IO.**

### **3.1 HARDWARE: ARDUINO.**

#### **3.1.1. CONCEPTUALIZACIÓN DE ARDUINO.**

Se trata de una plataforma de **código abierto** [Open Source] de creación de prototipos basada en **hardware y software libres**, de público dominio. Su entorno de aplicación es multiplataforma, pudiéndose así instalar y ejecutar en los siguientes sistemas operativos: Windows, Mac OS y Linux.

La **plataforma de ARDUINO** ejecuta una serie de datos que se le han introducido con anterioridad, para ello dispone del software o de un programa para poder insertar estos datos a la placa a utilizar. El software de Arduino es un programa informático formado por un conjunto de herramientas de programación y se le conoce como **IDE** [Integrated Development Environment], o lo que es lo mismo, "Entorno de Desarrollo Integrado".

El IDE de Arduino consiste en un editor de código, un compilador, un depurador y un constructor de interfaz gráfica (GUI). El IDE se encuentra instalado en el PC y en él se escribe el programa que se quiere que Arduino ejecute. Se configurará, a continuación, el IDE para que se establezca la conexión con la placa a través del cable USB. No obstante, dado que Arduino presenta la limitación de estar conectado mediante USB, en este proyecto se va a utilizar una placa NodeMCU que permitirá que se realice sin precisar este medio de conexión, posibilitando la comunicación, envío y captura de datos a través de la red WiFi.

El uso de una **placa NodeMCU**, basada en el popular **SoC** [System on Chip], describe la tendencia cada vez más habitual de utilizar tecnologías de fabricación que integran todos o gran parte de los módulos que componen un computador o cualquier otro sistema informático o electrónico en un único circuito integrado o chip. La placa que se ha utilizado en este proyecto es **NodeMCU 1.0 (ESP-12E Module)**. NodeMCU es una placa de desarrollo en el ESP12E, el cual es el módulo más común que integra el SoC ESP8266.

Por otra parte, comprende un sistema embebido, lo que quiere decir que está diseñado para cubrir necesidades específicas, a diferencia del PC que cubre un amplio abanico de necesidades. De esta forma se puede ordenar a la placa lo que debe de hacer con solo escribir un código y subirlo al microcontrolador usando su lenguaje propio de programación (basado en Wiring) y el software de Arduino [IDE] con el que programar la placa empleada, basado en Processing.

En cuanto al hardware, por lo general, éste consiste en un microcontrolador re-programable Atmel AVR y una serie de pines, conectado sobre una placa de circuito impreso a la que se le pueden conectar placas de expansión [**shields**] a través de la disposición de los puertos de entrada y salida presentes en la placa seleccionada. Los shields complementan la funcionalidad de la placa empleada, añadiendo circuitería, sensores y módulos de comunicación externos a la placa original. También, se pueden comunicar con la plataforma de Arduino, bien por los pines o bien por algún bus como el SPI, I<sub>2</sub>C o puerto serie.

El **microcontrolador** es un circuito integrado programable, capaz de ejecutar las órdenes registradas en su memoria. Se encuentra formado por diversos bloques funcionales, los cuales desempeñan funciones específicas. Un microcontrolador integra tres unidades esenciales: memoria, periféricos de E/S y CPU (unidad central de procesamiento).

La principal característica del software y del lenguaje de programación es su **simplicidad** de uso.

### **3.1.2. HISTORIA.**

La fundación de Arduino tuvo lugar en el año 2005, de manos de un estudiante del instituto de Diseño Interactivo IVRAE (Italia) como una herramienta para el estudio en la rama relativa a la computación y electrónica que permitiese interactuar con el entorno a través de sensores y actuadores a un bajo costo y que ofreciese una amplia diversidad. Su fin no era lucrativo, sino que nació por la necesidad de subsistencia en que se veían inmersos en aquella época ante la inminente clausura del instituto.

No obstante, anteriormente, en el año 2003 un estudiante colombiano, Hernando Barragán, había llevado a cabo su tesis la cual constaba de una plataforma de desarrollo Wiring, que se componía de una placa que incorporaba un circuito impreso junto con un microcontrolador ATmega168 y un Entorno de Desarrollo Integrado (IDE), éste último incluía una serie de librerías y funciones que permitían una fácil programación del microcontrolador. Debido a su gran costo, surgió la idea de crear placas que fueran más económicas y de menor tamaño, a las que denominaron Arduino.

El nombre hace alusión a un bar llamado Bar di Re Arduino (antiguo Rey Europeo en torno al año 1002), donde Massimo Banzi, uno de los miembros del grupo de creadores de esta placa, pasaba gran parte de su tiempo.

Entonces, fue pocos años después cuando Massimo Banzi y David Cuartielles consiguieron que el proyecto inicializado por el pionero Barragán derivara en una plataforma de código abierto, accesible a una amplia población, fue ahí cuando crearon el hardware de Arduino y apoyándose en otro joven estudiante, David Mellis, consiguieron desarrollar el software de Arduino. También contribuyeron al desarrollo del mismo otras dos personas, Tom Igoe como asesor y Gianluca Martino quien era el responsable de la creación de las placas.

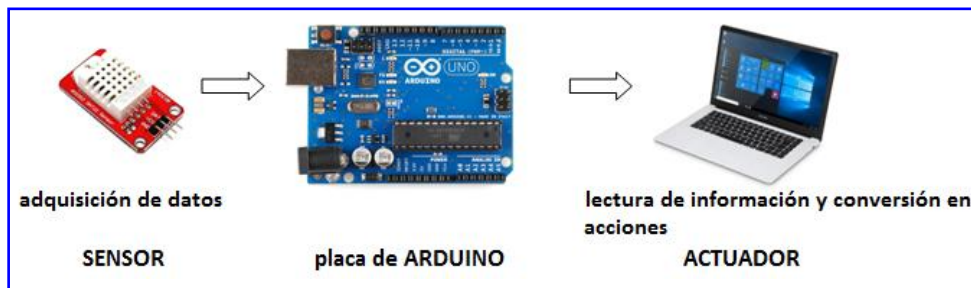


Figura 12. Esquema entorno de Arduino. Fuente: Elaboración propia.

### **3.1.3. VENTAJAS.**

- Barato en comparación con otras plataformas de microcontroladores.
- Software y hardware de código abierto y extensible.
- **Multiplataforma:** El software de Arduino se ejecuta en diversos sistemas operativos, como son Windows, Mac OS X y GNU/Linux. La mayor parte de los sistemas de microcontroladores se limitan a Windows.
- Entorno de trabajo fácil y accesible tanto a perfiles profesionales como a principiantes en el mundo de la electrónica.

En cualquier caso, con un compilador estándar de C++ podría existir la posibilidad de que no reconociese un programa de Arduino, es por ello que, cuando se pulsa el botón "Subir", se realiza una copia del código en un archivo temporal y el Sketch que se suba a Arduino deberá ir encabezado por la instrucción `#include` y la función `main()`, transformándolo de este modo en un programa perfectamente ejecutable en el lenguaje C++.

### **3.1.4. ESTRUCTURA DE UN PROGRAMA.**

Se compone al menos de dos partes, o funciones, que encierran bloques con declaraciones o instrucciones. Estos bloques se encuentran encabezados por las funciones **setup()** y **loop()**.

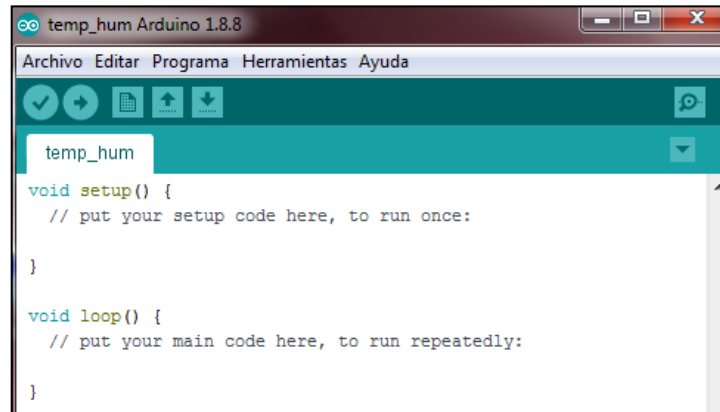


Figura 13. Las dos partes fundamentales de un programa de Arduino. Fuente: Elaboración propia.

La función **setup()** se ejecuta la primera y una única vez, se ocupa de recabar la información y debe contener la declaración de las variables del programa y **loop()** es para ejecutar el programa cíclicamente, esta última función contiene el código, es el centro del programa.

Si la función no devuelve ningún valor, se colocará delante el término **void**, el cual significa (función vacía).

### **3.1.5. FUNCIONES FUNDAMENTALES DE UN PROGRAMA DE ARDUINO.**

- **delay(ms)**: Se utiliza para incluir en el programa una pausa de una duración determinada, la cual se coloca en milisegundos. Ej.: `delay(1000)` //equivale a una pausa de 1 s (1 segundo = 1000 milisegundos).

- **Serial.begin(speed)**: pone en marcha el puerto serie y recoge la velocidad de transmisión de datos en serie expresada en bits por segundo (**baudios**). La velocidad más empleada para la comunicación con el ordenador es de 9600 bits por segundo.

```
void setup()
```

```
{
  Serial.begin(9600); //apertura del puerto serie fijando la velocidad en 9600
  bits por segundo
}
```

- **Serial.println(data)**: muestra los datos por el monitor serie seguidamente de un retorno de línea automático, facilitando así la lectura de los datos.

- **Serial.println(data,data type)**: Tiene un uso similar al comando Serial.print(). Los parámetros que lo forman responden a lo siguiente:

- data: número o cadena de caracteres que es devuelta.
- data type: especifica el formato de salida por pantalla (decimal, octal, binario, ...) (DEC, OCT, BIN, ...).

- **Serial.print(data,data type)**: envía un número o una cadena de caracteres al puerto serie.

- **millis()**: esta función da el número de milisegundos desde que se ejecutó el programa hasta el momento actual.

`time=millis(); // se recogen los milisegundos que lleva en funcionamiento el programa desde que se cargó la placa o desde que se reinició la misma mediante el Reset (RST) en time.`

Para la visualización por pantalla del programa ejecutado se irá a Herramientas/Monitor serie.

### **3.1.6. ALTERNATIVAS.**

Cada vez cobra una mayor importancia **el ESP8266** en el cosmos del IoT [Internet of Things]. Con un pequeño procesador se consigue una **conexión WiFi** compatible con el software de Arduino, lo que supuso una gran ventaja para la innovación tecnológica. El ESP8266 ocupa un papel crucial hoy en día en el mundo del IoT, pues es un microprocesador completo de bajo coste que lleva WiFi incorporado. Es por consiguiente, uno de los mayores competidores a los que tiene que hacer frente Arduino.

La novedad que situó por delante al ESP8266 de Arduino, fue que este último precisa de conexión USB, mientras que por el ESP8266 la conexión se establece por medio de WiFi.

#### **3.1.6.1. ¿QUÉ ES EL ESP8266?**

El ESP8266 es un **SoC** [System on Chip] confeccionado por la compañía china Espressif. Cuenta con diversos componentes, todos ellos integrados en uno mismo, principalmente un procesador de 32 bits y un chip WiFi.



El procesador integrado en el ESP8266 es un Tensilica L106 de 32 bits que funciona a una velocidad de 80 Mhz, pudiendo alcanzar como máximo unos 160 Mhz.

### 3.1.6.2. CARACTERÍSTICAS DEL ESP8266.

- Procesador de 32 bits de bajo consumo.
- Una velocidad de 80 Mhz, pudiendo alcanzar como máximo 160 Mhz.
- 32 kiB RAM instrucciones, 32 kiB RAM cache.
- Pila de **TCP/IP** integrada.
- 16 pins GPIO.
- Conversor analógico digital de 10 bits.
- Voltaje de operación (3,0-3,6) V.
- Consumo medio 80 mA.



Figura 14. Procesador ESP8266. [22] ESP8266, La alternativa a Arduino con WiFi. Marzo, 2018.

El ESP12E está resultando ser el modelo preferido por el ESP8266.

- Pila de **TCP/IP** integrada. Se denomina así debido a los protocolos más importantes que lo componen: Protocolo de Control de Transmisión (TCP) y Protocolo de Internet (IP), los cuales fueron los primeros en definirse.

La alimentación del ESP8266 es de 3,3 V, no pudiendo exceder de 3,6 V lo cual provocaría la destrucción del SoC. Pese a esta observación, hay algunas placas que contienen reguladores de voltaje, lo cual las permite una alimentación de hasta 5 V.

Por otra parte, un **GPIO** es un pin genérico en un chip, cuyo comportamiento puede ser controlado (programado) por el usuario en tiempo de ejecución. Una de sus aplicaciones son los sistemas embebidos como por ejemplo Arduino, que hace un uso intensivo del mismo para poder obtener lecturas de variables climáticas como T, P y HR a través de sensores.

La corriente que pueden absorber o proporcionar los pines digitales como máximo es de 12 mA.

El convertor analógico digital (ADC) puede registrar una tensión de valor máximo 0-1 V. Sin embargo, algunas placas incorporan divisores para poder ser ampliada esta tensión en el siguiente rango: 0-3,3 V.

Las funcionalidades de algunos de los pines se recogen seguidamente:

- El pin **EN** permite examinar el encendido o apagado del SoC.
- El pin **RST** examina el RESET del ESP8266.

### **3.1.6.3. MÉTODOS DE AHORRO DE ENERGÍA DE LOS PROCESADORES ESP8266.**

Los procesadores ESP8266 conllevan un gasto enorme de energía debido a la conexión WiFi que permiten establecer, no obstante, es por esta conexión WiFi por la que resultan especialmente útiles. Para contrarrestar esta desventaja que ofrecen, existen unos modos de implementar en la programación que permiten suavizar este gasto de energía.

Los **modos de ahorro de energía** que pueden utilizarse en estos procesadores son los siguientes:

- **Modem-Sleep**: con este modo lo que se logra es mantener en suspense la conexión del NodeMCU con el punto de acceso [el router] de la red WiFi cuando esta no sea necesaria y ser conectada de nuevo en el momento solicitado.
- **Light-Sleep**: a diferencia del modo anterior, permite mantener activa la conexión, aminorando el consumo de energía en los períodos en que no se estén enviando datos.
- **Deep-Sleep**: se trata del modo con menor consumo consecuencia de que la placa se comporte en ciertos intervalos de tiempos, definidos por el programador, de manera ociosa, o lo que es lo mismo permaneciendo dormida. Lo único que estará en pleno funcionamiento será el reloj en tiempo real para posibilitar el inicio de la toma de datos en el tiempo preestablecido.

#### **3.1.6.3.1 MODO DE AHORRO DE ENERGÍA DEEP-SLEEP.**

Es uno de los tres modos que ofrecen los procesadores ESP8266, como bien se ha citado anteriormente. La posibilidad que ofrece de permanecer el WiFi, el reloj del sistema (System Clock) y la unidad central del proceso (CPU)

desactivados cuando no hay toma de datos, lo sitúa por delante de los otros modos, siendo el ahorro de energía mucho más notorio.

Únicamente, se encuentra activo el reloj en tiempo real (Real Time Clock, RTC) que será quien se ocupe de enviar la señal para inicializar el procesador, al final del tiempo de suspenso que se le haya especificado. Para que esto suceda, debe de conectarse una vez subido el Sketch al programa, un cable que por un extremo se encuentre conectado al GPIO16(D0) y por el otro lado al pin de RESET (RST). De no ser así, no se produciría el reinicio de la placa en los períodos marcados.

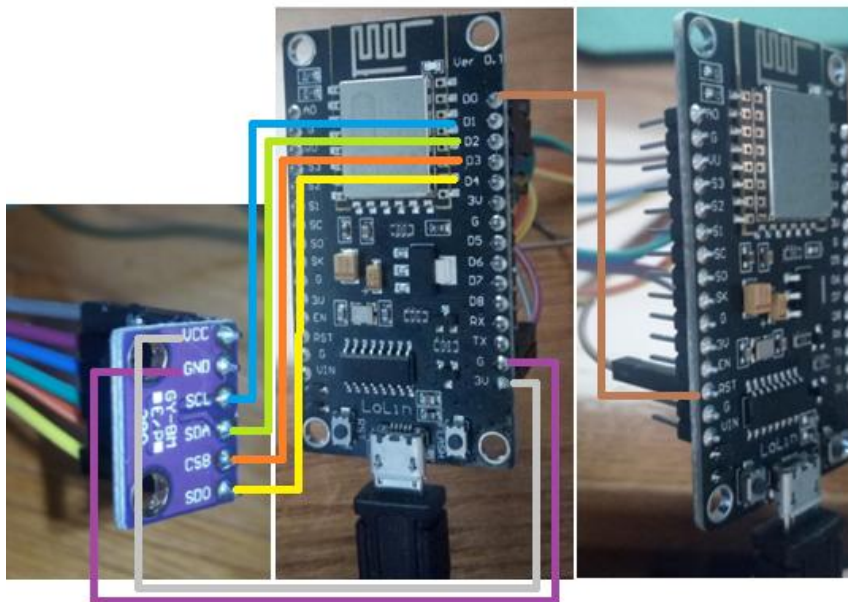


Figura 15. Circuito configuración modo Deep-Sleep en placa con sensor BME280. Fuente: Elaboración propia.

### 3.1.6.3.2. FUNCIÓN DEEP-SLEEP Y PROGRAMACIÓN.

Para poder establecer el modo Deep-Sleep, se procederá a conocer la función a la que habrá que llamar, la sintaxis de la misma es la siguiente: **ESP.deepSleep()** la cual contiene dos variables, por lo que su llamada a la función de forma explícita es la siguiente: **ESP.deepSleep(tiempo\_en\_us, modo\_reinicio).**

Los dos parámetros de los que consta se definen a continuación:

- **tiempo\_en\_us(uint32\_t):** tiempo expresado en microsegundos,  $\mu$ s, que debe permanecer el microprocesador en suspense, pudiendo ser como máximo de unos 71 minutos y 35 segundos.

El razonamiento de esta estimación se muestra a continuación:

Los procesadores ESP8266 trabajan a 32 bits, por lo que en microsegundos el tiempo máximo será de unos  $2^{32} = 4.294.967.296\mu\text{s}$ , o lo que es lo mismo decir, unos  $4.294,97 \text{ s} = 71' 35''$ .

- **modo\_reinicio**: hace alusión al modo de reinicio del microprocesador junto con la calibración de radio para el WiFi. Existen cuatro tipos:

- **WAKE\_RF\_DEFAULT**: una vez reiniciado el microprocesador, solamente se calibra la señal de radio en el caso que el chequeo de error.

- **SLEEP\_TIME, WAKE\_RFCAL**: siempre se calibra la señal de radio al reiniciarse el microprocesador, lo que genera un consumo superior.

- **SLEEP\_TIME, WAKE\_NO\_RFCAL**: es la antítesis al modo precedente, pues no se calibra la señal de radio, viéndose el consumo reducido.

- **WAKE\_RF\_DISABLED**: cuando se reinicia el microprocesador se deshabilita la señal de radio, con un consumo ínfimo, aunque no permite el envío ni la toma de datos a través de la red WiFi.

Continuando con la plataforma empleada a lo largo de este TFG, Thingier.io, se podrá determinar cuándo se quiere que la información sea enviada del procesador a la API, bien para ser almacenada o representada para su posterior análisis y visualización, a través de los recursos de transmisión [**streaming resources**]. Con el modo **Deep-Sleep** que se empleará en el siguiente código, se ilustrará a continuación, cómo se verá prolongada la vida útil de una batería.

En cambio, una de las limitaciones que ofrece esta plataforma es que la admisión de los datos debe ser en períodos superiores a los 55 segundos, para que puedan ser recogidos en los **Data Buckets** correspondientes. Sin embargo, la visualización de los datos puede ser realizada de forma casi instantánea en los **Dashboards**.

El modo a proceder es similar al de la lectura de datos de temperatura y humedad para el sensor DHT22. Será necesario crear un dispositivo, como se especifica en el apartado 3.2 y hacer coincidir los datos de éste con el correspondiente Sketch que se subirá a Arduino. Una vez realizados todos estos pasos, se subirá el Sketch de la Figura 16, en el cual se pueden ver los datos que se desean enviar a la API que irán entre paréntesis en la función **thing.stream()**. Transcurridos 10 minutos desde que se produjo el inicio de la



placa, ésta entrará en modo Deep-Sleep. El tiempo que debe permanecer en suspense la placa se definirá en la línea en que se detalla el parámetro `uint32_t` de la función `ESP.deepSleep()`, y transcurrido el mismo, volverá a inicializarse, tomando datos cada 10 minutos y pasados estos volverá a entrar en modo "sueño", y así cíclicamente. El tiempo que se ha programado para que la placa esté en suspense es de 10 minutos, que expresado en  $\mu$ s será de 600e6. El tiempo en que se producirá la toma de datos se explicita a través de la función `millis()` donde se establece un tiempo de 60 segundos que irá en milisegundos, por lo tanto se pondrá como 60.000 milisegundos.

```
#define _DISABLE_TLS_
#include <ESP8266WiFi.h> //Librería de establecimiento de la conexión WiFi del módulo ESP8266
#include <ThingierESP8266.h> //Librería característica de la plataforma Thingier.io
#include "DHT.h" //Librería del sensor utilizado DHT22

// Parámetros del establecimiento de la conexión con la plataforma Thingier.io
#define usuario "nubethmcu2k19"
#define device_Id "IdNodeMCU1"
#define device_credentials "$UVQMOdYd!vk"

ThingierESP8266 thing(usuario, device_Id, device_credentials);

// Definición de los parámetros del sensor empleado DHT22
#define DHTPIN 4 //Pin para el establecimiento de la conexión - GPIO02
#define DHTTYPE DHT22 //Modelo del sensor utilizado
DHT dht(DHTPIN, DHTTYPE);

// Tiempo del modo Deep-Sleep del módulo ESP8266 (expresado en microsegundos)
uint32_t SLEEP_TIME = 600e6;

// Parámetros de conexión WiFi
const char WiFi_ssid[]="uva_IOT"; //Nombre de red
const char WiFi_password[]="www.uva.es"; //Clave de red
// Variables globales
float h; //Variable de humedad
float t; //Variable de temperatura

void setup() {

// Puesta en marcha del sensor DHT22
dht.begin();

// Inicialización de la WiFi para comunicarse con la API
thing.add_WiFi(WiFi_ssid, WiFi_password);

// Lectura y recogida de datos desde la API
thing["DHT22"] >> [(pson& out){
  out["Temperatura"] = t;
  out["Humedad"] = h;
};
}

void loop() {
  thing.handle();

  float t_NOverificada = dht.readTemperature();
  float h_NOverificada = dht.readHumidity();
```



```

if (isnan(t_NOverificada) || isnan(h_NOverificada)) { //Se comprueban que las medidas sean correctas
  return;
}
else{
  t = t_NOverificada;
  h = h_NOverificada;
  thing.stream(thing["DHT22"]); //Si las lecturas han sido favorables, son enviadas a la plataforma
Thingier.io
}

if (millis() >= 60000){
  ESP.deepSleep(SLEEP_TIME, WAKE_RF_DEFAULT); //Se usa este modo de reinicio
(WAKE_RF_DEFAULT) considerándolo como el más acertado (ver arriba en qué consiste)
}
}

```

Figura 16. Código de programa para el establecimiento de la conexión de la placa con la plataforma Thingier.io. incorporando el modo Deep-Sleep.

En cuanto a la vida útil de las baterías, se hará una estimación de ella a partir del anterior Sketch junto con los siguientes datos:

- Consumo stand-by: 0,02 mA en el modo Deep-Sleep.
- Consumo en funcionamiento: 60 mA.
- Batería compuesta por dos pilas de 2100 mAh de carga.

Por otro lado, el ciclo que se ha programado consta de un período en uso de 60 s y en stand-by (en reposo) 600 s, de modo que el tiempo total del ciclo será de 660 s, por lo que el consumo total expresado en mAs/ciclo será de:

$$60 \text{ mA} \cdot 60 \text{ s} + 0,02 \text{ mA} \cdot 600 \text{ s} = 3.612 \text{ mAs/ciclo}$$

La batería consumirá:

$$2.100 \text{ mAh} \cdot \frac{3.600 \text{ s}}{1 \text{ h}} = 7.560.000 \text{ mAs}$$

El número de ciclos que estará el dispositivo en funcionamiento será de:

$$\frac{7.560.000 \text{ mAs}}{\frac{3.612 \text{ mAs}}{\text{ciclo}}} = 2.093,02 \text{ ciclos}$$

Como el tiempo del ciclo es de 660 s:

$$2.093,02 \text{ ciclos} \cdot \frac{660 \text{ s}}{1 \text{ ciclo}} = 1.381.395,35 \text{ s} = 383,72 \text{ h} = \mathbf{15,99 \text{ días}}$$

La vida útil del dispositivo será de 15,99 días.

En cambio, si no se emplease este modo "sueño", Deep-Sleep, la vida útil de la batería se vería notablemente reducida, en la Figura 17 se mostrará el resultado de días que duraría la batería.

Modo NORMAL			Modo Deep-Sleep		
Batería:	2.100	mAh	Batería:	2.100	mAh
Consumo:	60	mA	Consumo:	60	mA
Stand-by:	0,02	mA	Stand-by:	0,02	mA
Ciclo:			Ciclo:		
Consumo:	660	s	Consumo:	60	s
Stand-by:	0	s	Stand-by:	600	s
Tiempo ciclo:	660	s	Tiempo ciclo:	660	s
<b>Consumo ciclo:</b>	<b>39.600</b>	<b>mAs/ciclo</b>	<b>Consumo ciclo:</b>	<b>3.612</b>	<b>mAs/ciclo</b>
<b>Batería:</b>	<b>7.560.000</b>	<b>mAs</b>	<b>Batería:</b>	<b>7.560.000</b>	<b>mAs</b>
<b>Ciclos:</b>	<b>190,91</b>	<b>ciclos</b>	<b>Ciclos:</b>	<b>2.093,02</b>	<b>ciclos</b>
<b>Tiempo:</b>	126.000	s	<b>Tiempo:</b>	1.381.395	s
	35	horas		383,72	horas
	<b>1,46</b>	<b>días</b>		<b>15,99</b>	<b>días</b>

Figura 17. Comparativa modo funcionamiento normal (izqda.) con el modo de ahorro Deep-Sleep (dcha.). Fuente: Elaboración propia.

La duración de la batería sería de 15,99 días en modo Deep-Sleep respecto a los 1,46 días que duraría trabajando en modo normal, de modo que queda total evidencia de la mejora que conlleva en cuanto ahorro de energía (consumo de unas 11 veces inferior al modo normal) y prolongación de la vida útil, si lo que se quiere es recoger datos con una periodicidad determinada y no continuada.

### 3.1.7. BUSES DE COMUNICACIÓN.

Un **bus** no es más que un sistema cableado al que se encuentran conectados dispositivos de una manera determinada para la transferencia de datos entre los mismos.

Existen dos tipos de transferencia de datos:

- En **serie**: Se transfieren los datos bit a bit. Un único cable para la transición.
- En **paralelo**: Permite la transferencia de varios bits simultáneamente.

Pensar que la conexión en paralelo puede resultar más eficaz encierra algunas trabas, como son una reducción en la frecuencia del reloj y en la longitud de los cables por los posibles ruidos o demoras en la señal.

En la transferencia de los datos intervienen como mínimo dos elementos clave:

- **Maestro (MASTER):** dispositivo que inicia la transmisión en el bus, genera la señal de reloj (Clock) y dictamina a qué esclavo va dirigida la orden.
- **Esclavo (ESLAVE):** dispositivo conectado al bus que recibe señales del maestro y no puede generar pulsos de reloj.

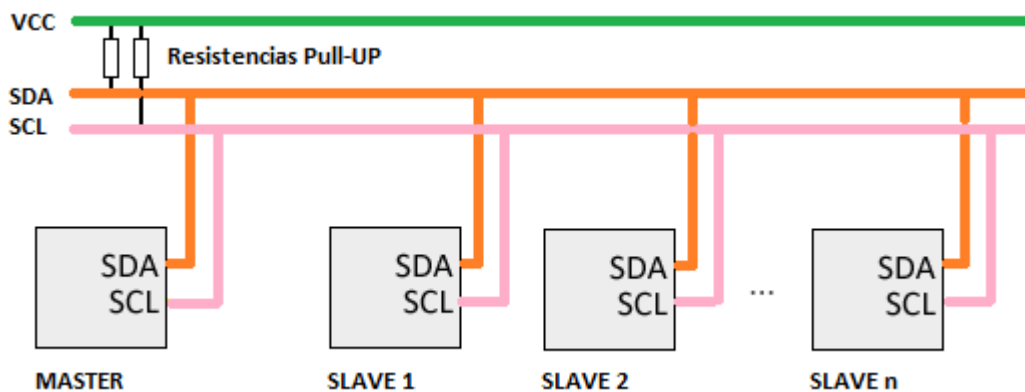


Figura 18. Componentes conectados en paralelo a las dos líneas del bus, SDA y SCL. Conexión multi-master. Fuente: Elaboración propia.

Para poder comunicarse Arduino con otros dispositivos a través del bus I<sup>2</sup>C, se debe tener la librería "Wire.h" instalada, ésta es proporcionada por el IDE de Arduino.

No obstante, el IDE dispone de otras librerías más avanzadas para este bus, como son: **I2Cdevlib** o **I2C library**.

En Arduino por lo general las resistencias internas Pull-UP se encuentran dentro de esta última librería citada. Estas resistencias hacen que la velocidad sea menor y las distancias de comunicación sean inferiores también. Para poder combatir estas carencias en situaciones que requieran de una mayor velocidad lo que debe hacerse es añadir físicamente unas resistencias Pull-UP. Comúnmente, el inconveniente que tiene este bus I<sup>2</sup>C es el de ofrecer una velocidad limitada.

### 3.1.7.1. BUS DE DATOS I<sup>2</sup>C.

Esta técnica de transmisión, **I2C** o **I<sup>2</sup>C**, responde a "Inter-Integrated Circuit". Este bus de datos en **serie** y síncrono, precisa únicamente de dos líneas de drenaje abiertas, una para los datos (SDA) y otra para la señal del reloj (SCL),



esto es una ventaja respecto del SPI. Sin embargo, su funcionamiento es más complejo.

Este tipo de bus fue diseñado por Philips en el año 1982. Es utilizado cada día más en la industria, para establecer la comunicación entre microcontroladores y sus periféricos en sistemas integrados [Embedded Systems] y, de forma más general, para la comunicación de circuitos integrados (ICs) entre sí que normalmente albergan en un mismo circuito impreso.

También se precisa de una tercera línea (conexión a tierra) en los casos en que los circuitos interconectados se encuentren en diferentes placas. En la mayor parte de los casos los circuitos van en una misma placa haciendo que no sea necesaria esta tercera línea adicional. Esta tercera línea es la que responde a la conexión a tierra (GND).

Las **tres líneas** de las que dispone un I<sup>2</sup>C son las siguientes:

- **SCL [System Clock | Serial Clock Line]** es la línea de pulsos de reloj que se encargan de sincronizar el sistema.
- **SDA [System Data | Serial Data Line]** es la línea por la que se transfieren los datos entre los diversos dispositivos.
- **GND [Ground=tierra]** toma de tierra común a la interconexión entre dos dispositivos diferentes acoplados al bus.

Los dispositivos conectados al bus I<sup>2</sup>C tienen **una dirección unívoca** para cada uno.

El bus I<sup>2</sup>C es **síncrono** puesto que el maestro manda una señal de reloj que mantiene sincronizados a todos los dispositivos que integran el bus, de modo que no se precise de un reloj propio para cada uno de los dispositivos.

Necesita un único conjunto de resistencias pasivas Pull-UP. Los dispositivos en el bus I<sup>2</sup>C pueden ser o maestros o esclavos. El maestro es el que controla la línea correspondiente al reloj. Los esclavos obedecen las órdenes del maestro. Un esclavo no puede iniciar una transferencia, a esta función responde el maestro. No obstante, tanto el maestro como el esclavo, pueden transferir datos a través del I<sup>2</sup>C.

Todas las direcciones del bus I<sup>2</sup>C son de 7 o de 10 bits. Cuando se envía una dirección de un dispositivo, de 7 bits, se están enviando en realidad secuencias de 8 bits. El bit plus que se incorpora, bit de lectura/escritura (R/W), se usa para registrar la información de si el maestro se encuentra leyendo al esclavo o escribiéndole, con valores 1 y 0, respectivamente.

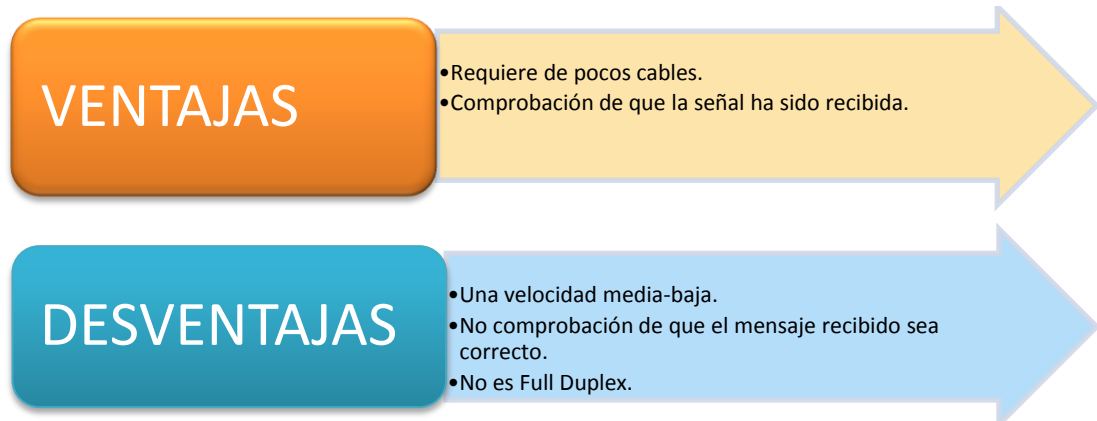


Figura 19. Ventajas e inconvenientes del bus I<sup>2</sup>C. Fuente: Elaboración propia.

### **3.1.7.1.1. CARACTERÍSTICAS DEL I<sup>2</sup>C EN ATMEGA328P:**

- Interfaz de comunicación sencilla, robusta, flexible y bidireccional.
- Compatibilidad de operaciones entre maestro y esclavos, pudiendo operar como emisor o receptor.
- El espacio de direcciones de 7 bits posibilita tener hasta  $2^7=128$  direcciones de diferentes esclavos.
- Puede alcanzar una velocidad de hasta 400 kbits/s en la transferencia de datos. Lo más común es que sea de unos 100 kbits/s.
- Disposición de controladores de salida de velocidad limitada y de que la señal se ha producido con éxito.
- Compatibilidad con el protocolo Philips I<sup>2</sup>C.
- En el Arduino UNO, los pines I<sup>2</sup>C están en los pines analógicos A4 (SDA) y A5 (SCL).
- En el Arduino MEGA y DUE, son el D2 (SDA) y el D1 (SCL).

#### **- Inconvenientes:**

- Baja-media velocidad.
- No verifica que el contenido del mensaje sea acertado.

No se requiere licencia para implementar el protocolo I<sup>2</sup>C y es conveniente para periféricos donde la simplicidad y el bajo coste de fabricación son más relevantes que la velocidad. Un hecho que le confiere gran importancia al I<sup>2</sup>C es que a través de un microcontrolador posibilita el control de una serie de chips que monitorizan varios dispositivos por medio de tan solo dos pines y un software de E/S.

Son muchas otras las tecnologías existentes de bus de datos, como puede ser el caso del bus **SPI [Serial Peripheral Interface Bus]**, pero éste último requiere de más pines para llevar a cabo la conexión entre diversos dispositivos.

### 3.1.7.2. BUS DE DATOS SPI.

El bus de datos **SPI [Serial Peripheral Interface]** fue creado por Motorola en los años 80. La correspondencia de datos responde a la comunicación maestro-esclavo que se explicó previamente. En este bus, la comunicación tiene lugar a través de dos líneas diferenciadas, una que va del maestro al esclavo, y otra que realiza el trayecto inverso, de modo que la comunicación es de tipo Full-duplex, o lo que es lo mismo, que el maestro pueda realizar a la vez la emisión o recepción de datos de manera simultánea. Se trata también de un bus síncrono.

Precisa de tres cableados como mínimo, a los que se deberá añadir una línea agregada SS [Slave Select = selección de esclavo], por cada uno de los dispositivos esclavos que intervengan en la comunicación, como se muestra en la Figura 20.

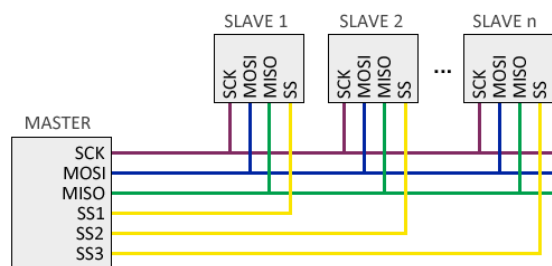


Figura 20. Líneas principales en un bus de datos SPI junto a las líneas adicionales (en amarillo). [22]

Las tres líneas principales son:

- **SCK**: hace referencia a la señal de reloj que ejecuta el maestro.
- **MOSI** (master-out, slave-in): comunicación maestro → esclavo.
- **MISO** (master-in, slave-out): comunicación esclavo → maestro.

Con esta configuración, se debe hacer frente a una desventaja, la cual pasa por disponer de una línea de cableado por cada uno de los esclavos, es por ello, que se admite una configuración en cascada en la que cada esclavo envía datos al siguiente, véase imagen inferior. Por consiguiente, se verá rebajada la velocidad de transferencia de datos.

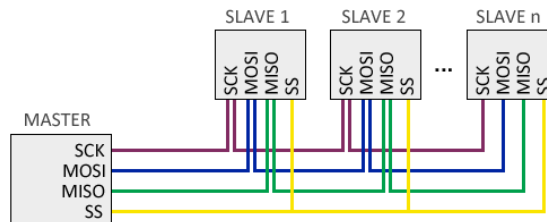


Figura 21. Configuración en cascada de los esclavos de un bus SPI. [22] Bus SPI en Arduino. Mayo, 2016.

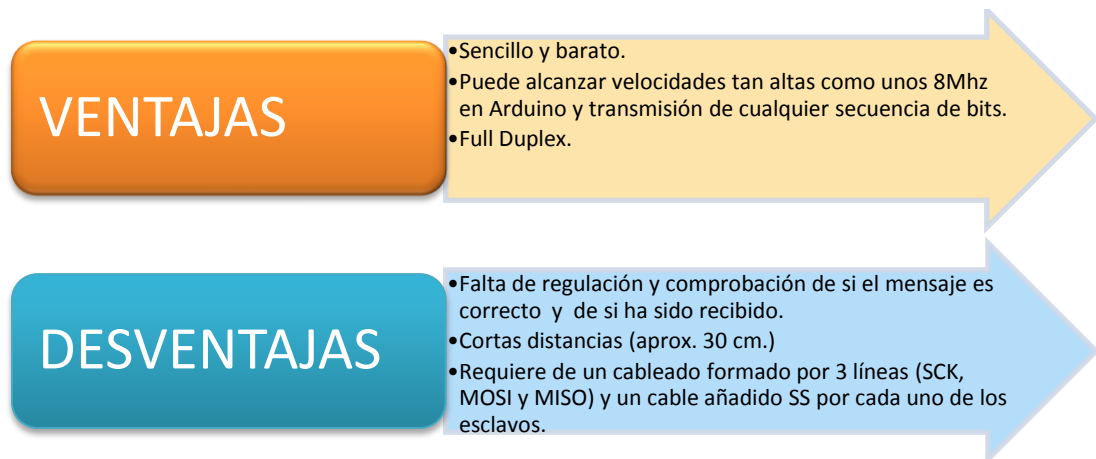


Figura 22. Ventajas e inconvenientes del bus SPI. Fuente: Elaboración propia.

### 3.2 SOFTWARE: PLATAFORMA THINGER.IO

**Thinger.io** es una plataforma que lleva incorporada una serie de librerías de código abierto (Open Source) que permite dirigir gran variedad de dispositivos, como ESP8266, Arduino, etc. a través de la red. Responde en buena parte al **IoT** (Internet of Things) permitiendo la interconexión de útiles comunes con Internet.

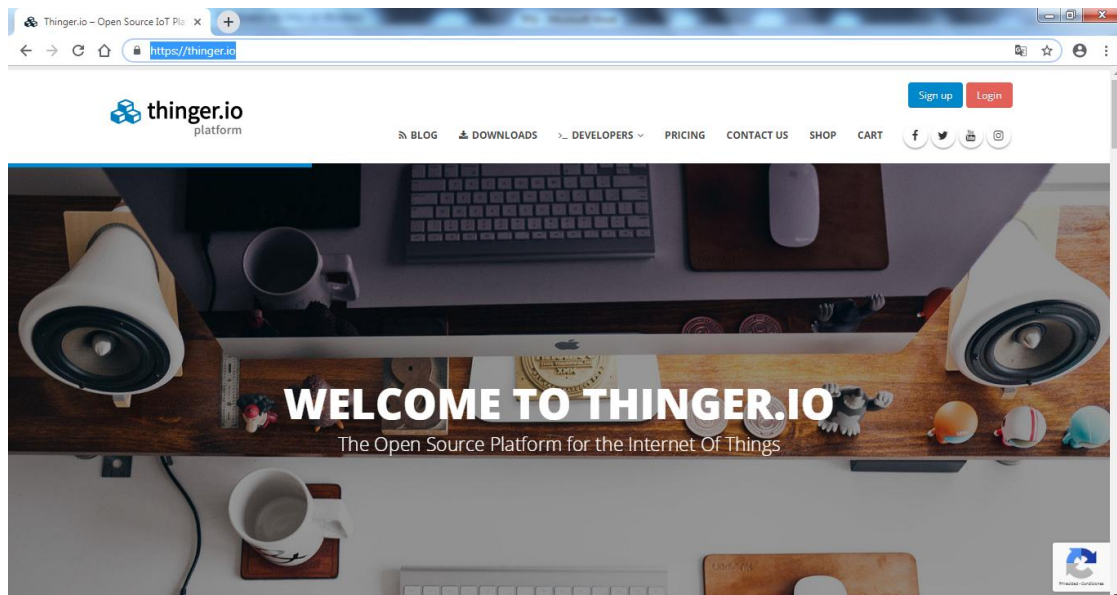


Figura 23. Vista principal de acceso a la plataforma Thinger.io.

Esta plataforma ofrece lo siguiente:

- registrar y mostrar por pantalla la información procedente de los sensores, a través de gráficas o bien pudiendo exportar esos datos para poder ser analizados por otro programa a posteriori.
- enviar órdenes e información a dispositivos (solicitar que una bomba se pare cuando se alcance cierto nivel de agua, establecer una temperatura a la que se tenga que activar un climatizador, etc.)

No obstante, tiene una serie de limitaciones:

- Permite como máximo la conexión de 3 dispositivos.
- No hay limitación en cuanto a recursos de los dispositivos, ni a la hora de recibirlos ni a la de enviarlos a los mismos.
- Número máximo de campos [Data Buckets] a registrar 10, pudiendo registrar cada Data Bucket datos de diversos dispositivos con una frecuencia máxima de 1 minuto (1 dato por cada minuto).
- Número máximo de gráficas de datos [Dashboards] 10.

Para recibir y **subir datos** a la plataforma de Thinger.io **a partir de** un sensor **DHT22** conectado a una placa NodeMCU y poder gestionarlos se procede según la siguiente **secuencia**:

Se procede al **registro** en la plataforma; una vez dentro de la misma, en el panel de control [Console Dashboard] se puede ver lo siguiente:

- estado de la cuenta [Account Stats]
- localización de los dispositivos que estén conectados [Connected Devices Locations]
- registro de la cuantía de transmisión de datos que se reciben y emiten [Data Transmission]

Se configura, a continuación, el dispositivo a utilizar. Una vez instalado éste, se va a la pestaña Statistics, donde aparecerá que el dispositivo está registrado (1/3 devices). En la Figura 24 aparecerá 1/2 devices pues el estudio se hace con dos dispositivos, uno para cada uno de los sensores empleados.

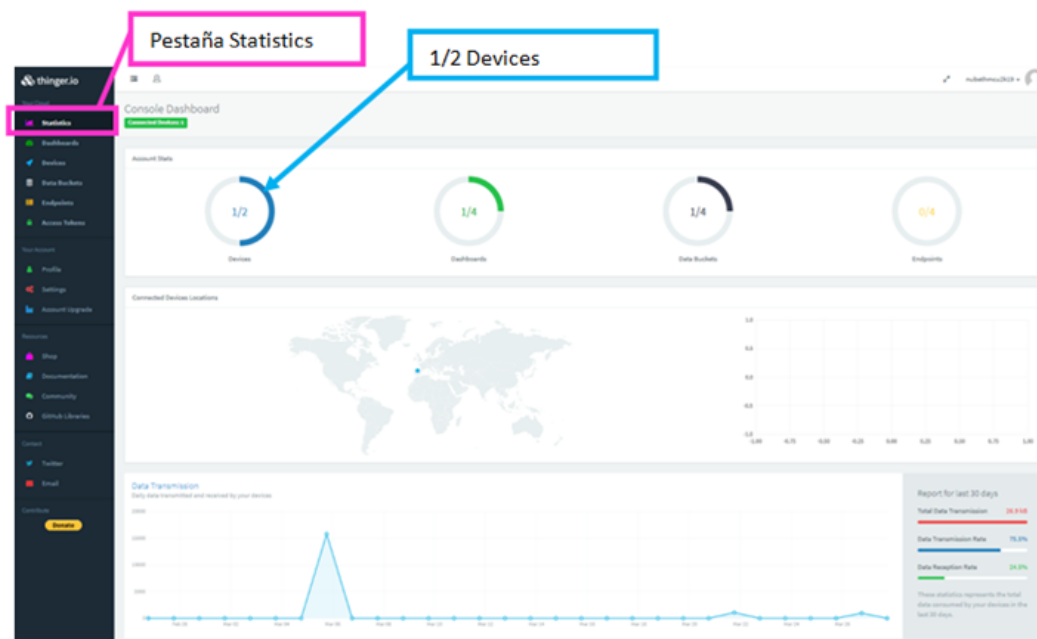


Figura 24. Pantalla principal de Thinger.io una vez funcionando uno de los dos dispositivos registrados.

El circuito de la placa será el mostrado en la Figura 25.

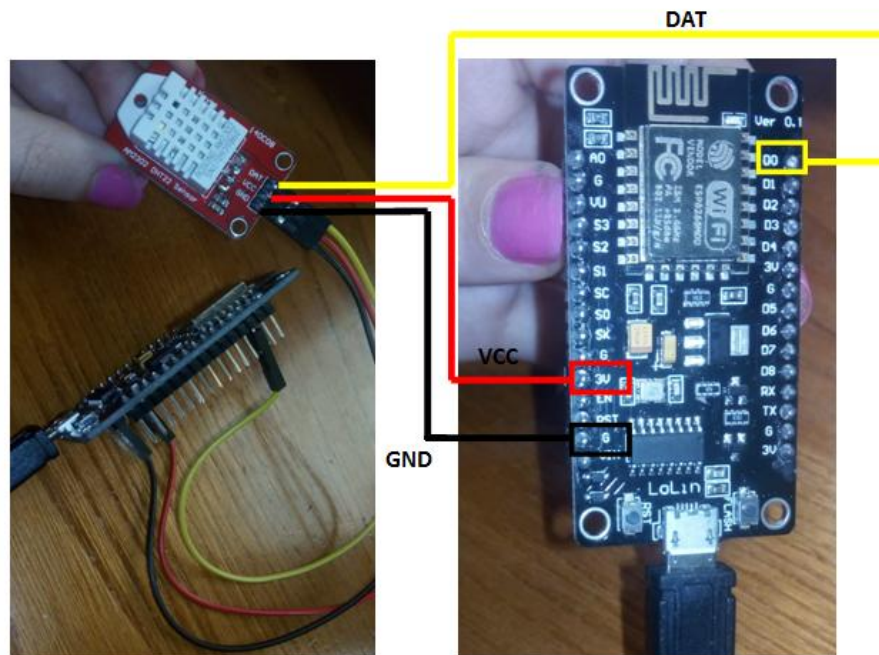


Figura 25. Circuito placa con sensor DHT22. Fuente: Elaboración propia.

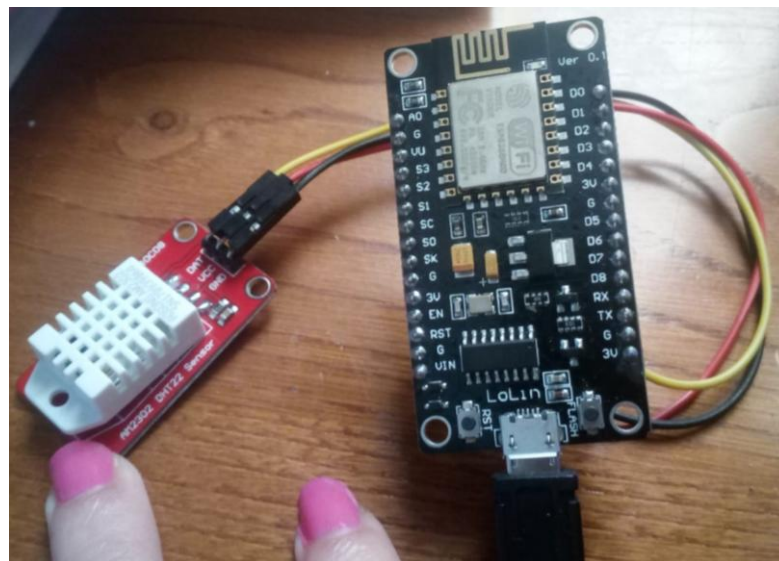


Figura 26. Conexiones placa (dcha.) junto al sensor AM2302 DHT22 (izqda.) con sus pines correspondientes. Fuente: Elaboración propia.

Para cargar el sketch de establecimiento de conexión entre la placa y la plataforma Thinger.io, se instalará la librería **thinger.io** by **Alvaro Luis Bustamante** en el Gestor de librerías de Arduino. También se precisa de la librería del sensor DHT22, de esta última se coge la versión última de la librería DHT sensor library by Adafruit.

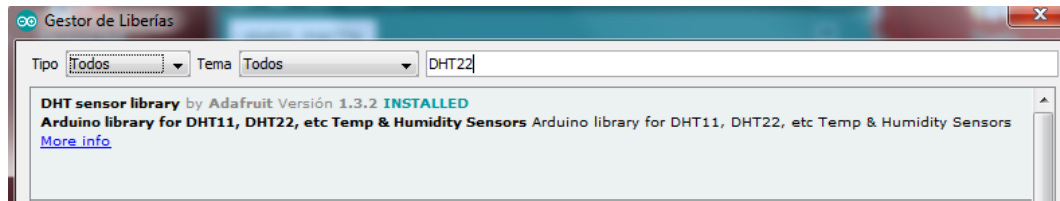


Figura 27. Instalación de la librería DHT, última versión.

Los parámetros de conexión con Tinger.io deben de coincidir con los del dispositivo instalado en la plataforma como se muestra en la Figura 28.

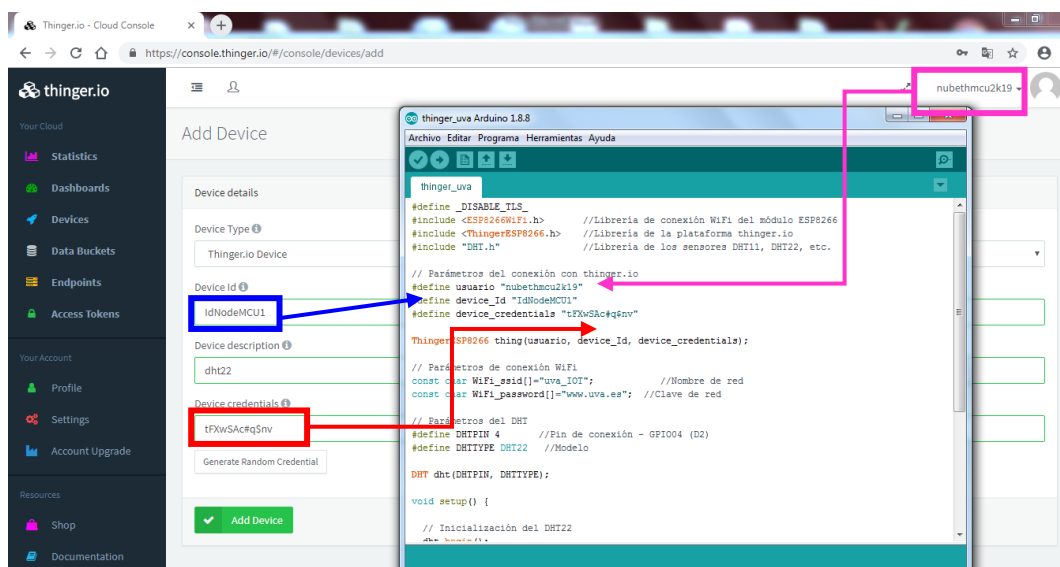


Figura 28. Datos al añadir un nuevo dispositivo para hacer esclava a la placa con la plataforma Tinger.io.

Una vez que se han hecho coincidir estos datos, junto a los de establecimiento de la conexión WiFi (nombre de la red = WiFi\_ssid y clave de la red = WiFi\_password) se procede a Subir el Sketch, con ello la placa se convierte en esclava de la plataforma Tinger.io, debiendo así enviar datos de temperatura y/o humedad a la misma. Pero, a pesar de convertirse en esclava se pueden realizar funciones con placas externas a tal conexión, sin contrariedad alguna.

Una vez que el Sketch está Subido, en la plataforma Tinger.io yendo al apartado de **Devices**, se puede comprobar el estado del dispositivo, deberá poner **Connected**, en caso contrario, en que ponga **Disconnected** se deberá proceder a revisar los pasos anteriores. Una vez que se han añadido los dos dispositivos a utilizar, en la pestaña Devices aparecerá lo siguiente:



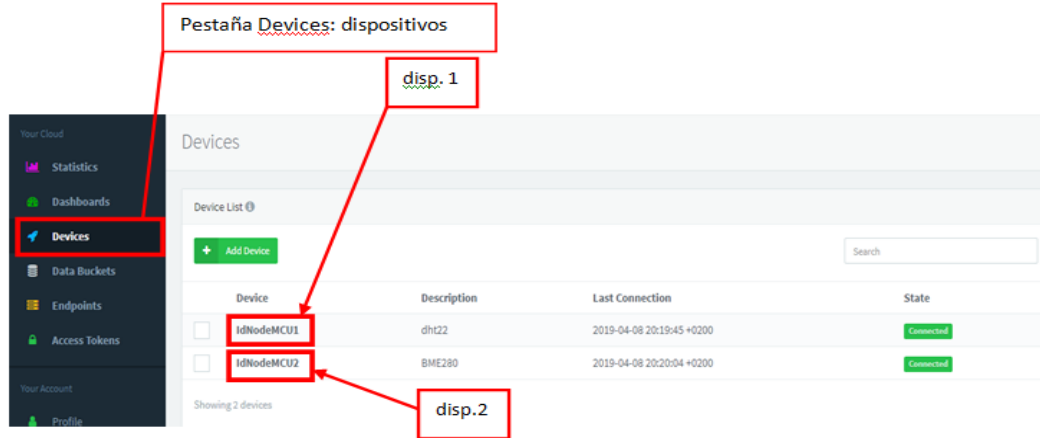


Figura 29. Dispositivos conectados.

Si se pulsa sobre el dispositivo, una vez conectado éste, se podrá ver por pantalla más información relativa a éste:

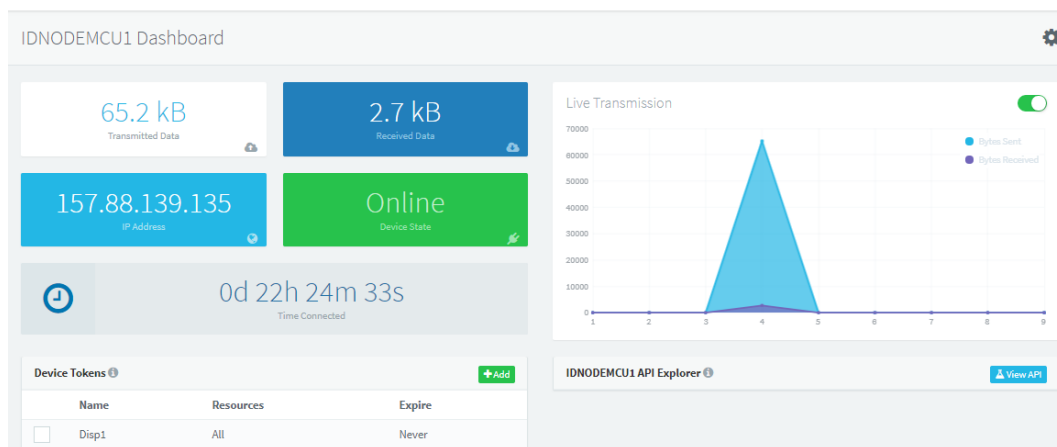


Figura 30. Información relativa al dispositivo 1: IdNODEMCU1.

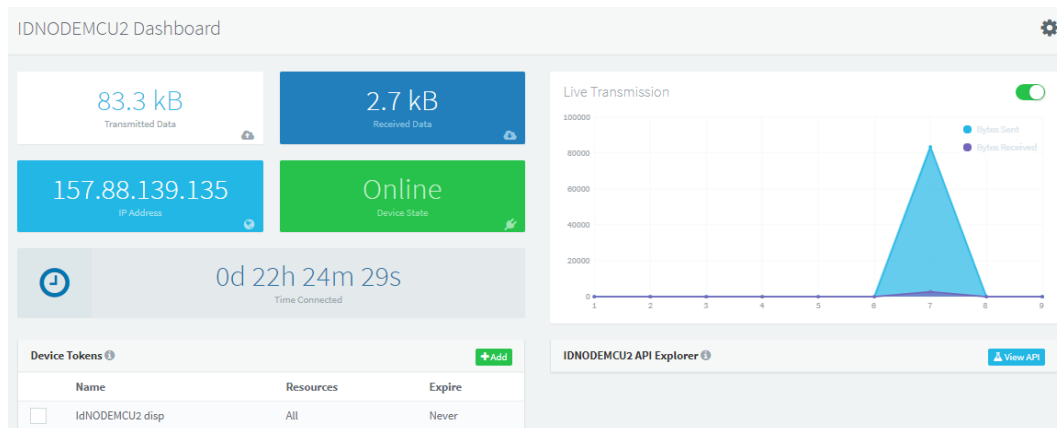


Figura 31. Información relativa al dispositivo 2: IdNODEMCU2.

Con el dispositivo conectado, para almacenar los datos se irá a la pestaña de Data Buckets y se registrará el Data Bucket con un nombre que no incluya espaciado, de no ser así incurriría en error. Es desde el propio Data Bucket desde el que los datos de los gráficos se pueden exportar, bien todos [Export all data in bucket] o especificando un rango de datos personalizado [Specify a custom data range for export the data] a otro programa para su posterior análisis, en esta memoria se han exportado a Excel.

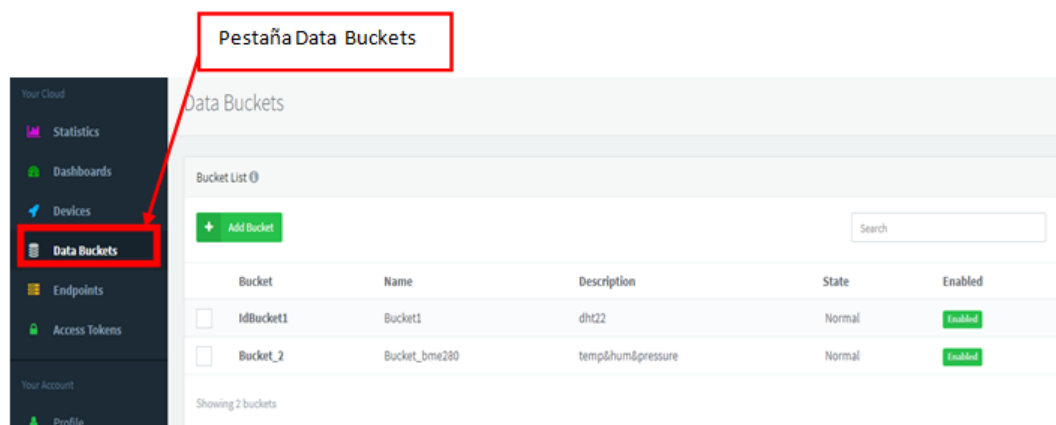


Figura 32. Cuadro de trabajo de los Data Buckets disponibles en el momento (máx. 10).

Date	Humedad	Temperatura
2019-03-26T18:37:39.086+0100	24,5	21,8
2019-03-26T18:36:39.094+0100	24,4	21,8
2019-03-26T18:35:39.084+0100	24,5	21,8
2019-03-26T18:34:39.089+0100	24,5	21,8
2019-03-26T18:33:39.084+0100	24,6	21,8
2019-03-26T18:32:39.083+0100	24,4	21,8
2019-03-26T18:31:39.084+0100	24,7	21,9
2019-03-26T18:30:39.084+0100	24,6	21,9
2019-03-26T18:29:39.096+0100	24,6	21,9
2019-03-26T18:28:39.084+0100	24,6	21,9

Figura 33. Exportación de datos a otro programa.

Por otro lado para la visualización gráfica de los datos, se irá a la pestaña Dashboards, registrando 1 de los 10 Dashboard posibles, dándole nombre sin incluir espaciado alguno.

Dentro de la pantalla gráfica (Dashboards), los diferentes elementos de representación gráfica que se muestran a continuación pueden ser manipulados a través de **Widget Settings**:

- **Time Series Chart**: gráfico que une con un spline los datos tomados en un tiempo determinado.
- **Donut Chart**: gráfico de donut para representar un solo dato registrado.
- **Progressbar**: gráfico de barras para representar un solo dato registrado.



- **Google Map:** mapa de Google Maps en que aparece la localización del dispositivo.
- **Image/MJPEG:** imagen obtenida desde una URL (vía Internet).
- **Text/Value:** un único valor de texto o numérico.
- **Clock:** reloj.

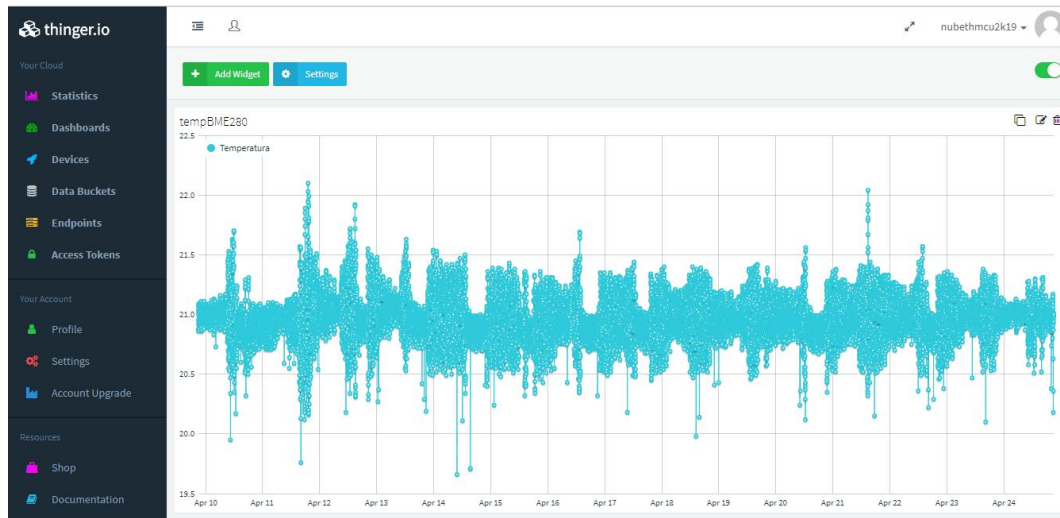


Figura 34. Dashboard tempBME280 activado el botón de la parte superior derecha [en verde].

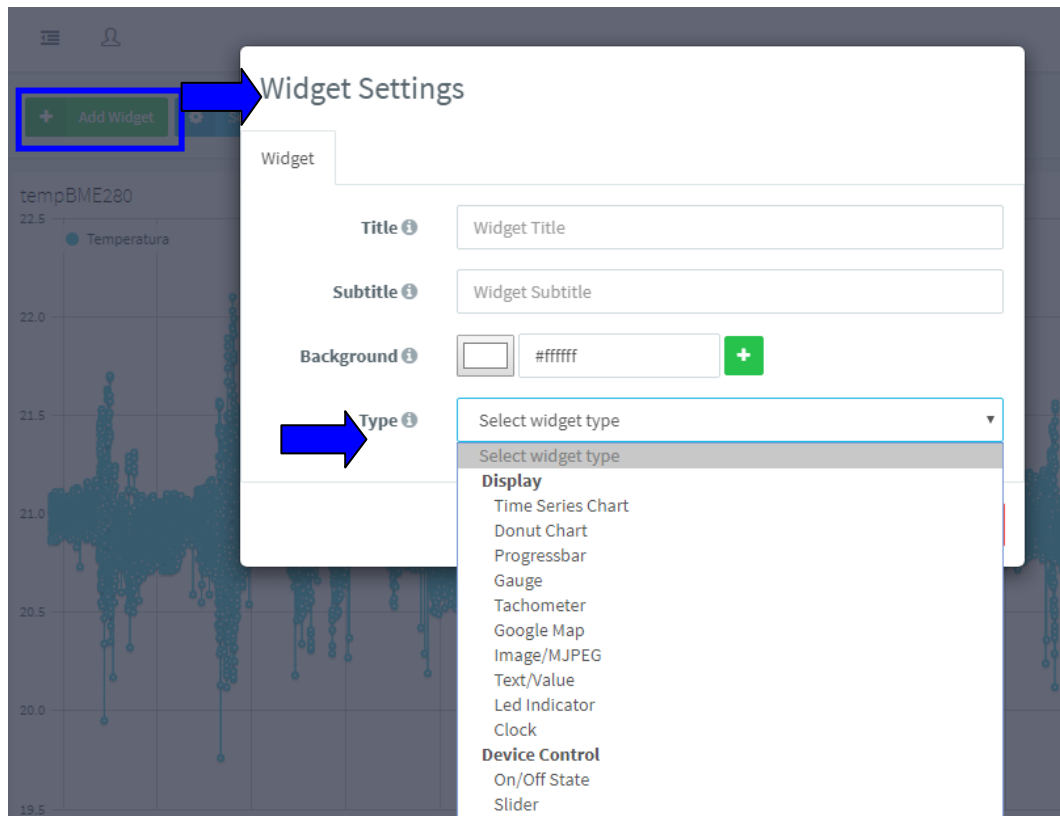


Figura 35. Operaciones que se pueden insertar dentro de la pestaña Add Widget, WidgetSettings, Type.

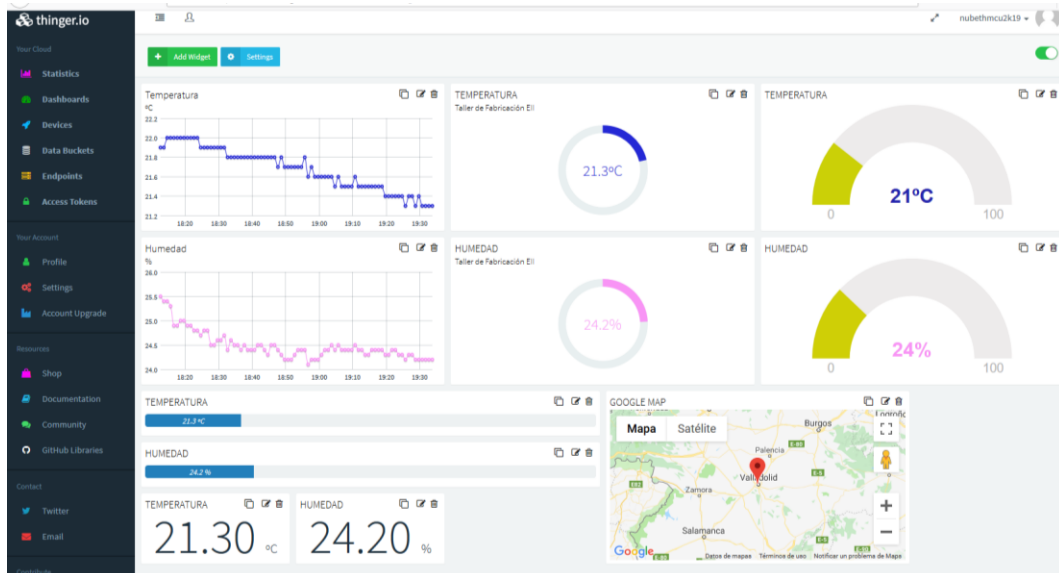


Figura 36. Diferentes gráficos o formas de visualizar los datos por pantalla en el área de trabajo: Dashboards.

Esta plataforma también ofrece la posibilidad de disponer en tiempo real de la información en el móvil, a través de la aplicación *thinger.io* que se puede descargar desde Play Store y es totalmente gratuita, el icono de la aplicación responde a la Figura 37.



Thinger.io

Figura 37. Insignia de la aplicación *thinger.io*.

Desde la plataforma en la pestaña *Devices*, señalando el dispositivo del que se quiera obtener su información, en el apartado *Device Tokens* dentro de su ventana de comandos, clicando sobre *Add*, se podrá crear el *Token* del dispositivo que permitirá establecer la transmisión de datos con la aplicación del móvil.

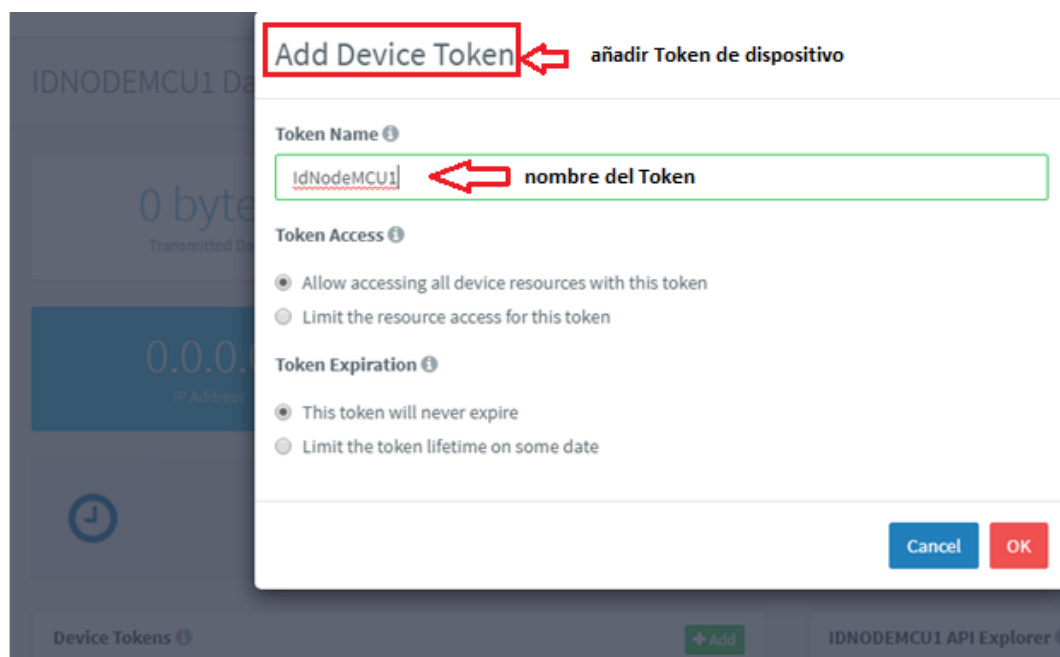


Figura 38. Creación del *Token* de dispositivo.

Ofrece, como puede verse en la Figura 38, la posibilidad de restringir qué recursos se quiere que sean compartidos y la caducidad o establecimiento de la expiración del *Token*.

Una vez creado el dispositivo, el acceso al mismo es inmediato, con solamente escanear con el móvil el código QR que se muestra al acceder al *Token* creado en la página web de *Thinger.io*.

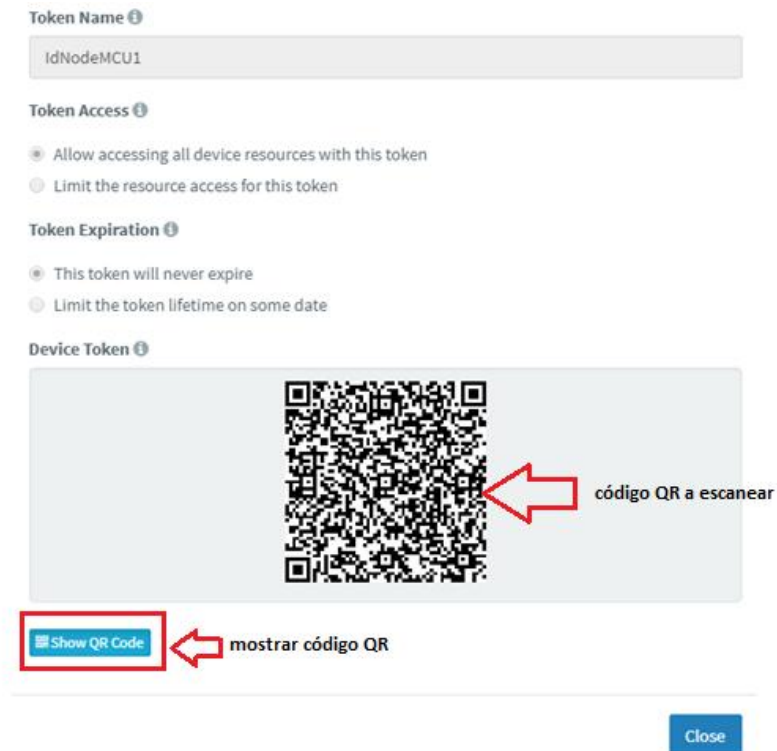
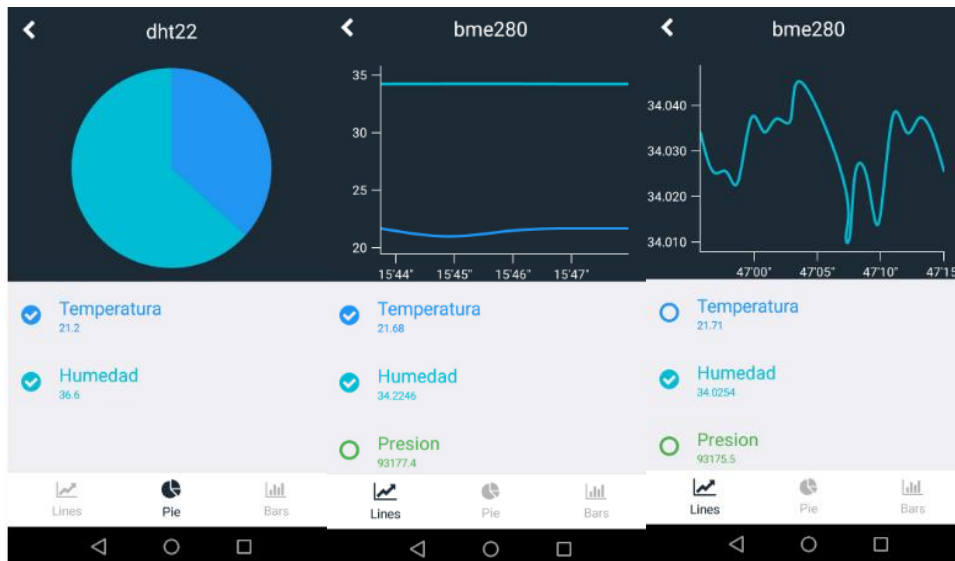


Figura 39. Código QR que permite el acceso al dispositivo desde la aplicación de Android.

Y, es a continuación cuando se procede a la lectura de los datos registrados por los dispositivos en tiempo real en la nube, siempre y cuando se quiera, desde cualquier dispositivo móvil y desde cualquier distancia. Esta aplicación permite visualizar de diferente forma, a través de gráficas y/o numéricamente, la información recibida del dispositivo en cuestión.



Settings		Settings		IdNodeMCU1	
Name	IdNodeMCU1	Name	IdNodeMCU2	<b>dht22</b>	
Device	IdNodeMCU1	Device	IdNodeMCU2	Temperatura	20.3
User	nubethmcu2k19	User	nubethmcu2k19	Humedad	38.5
Server	https://api.thinger.io	Server	https://api.thinger.io	Update Charts	
Token QR	→	Token QR	→		
Token creation date	Tue Apr 2 16:56:04 2019	Token creation date	Tue Apr 2 16:58:49 2019		
Token expiration date	Never	Token expiration date	Never		
Remove		Remove			



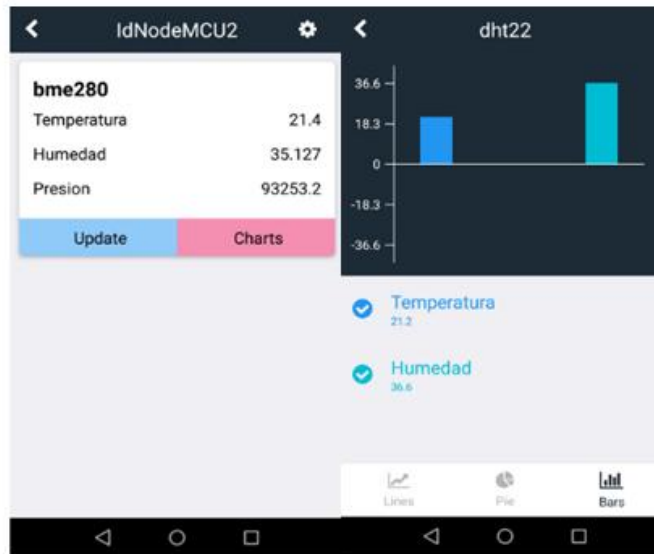


Figura 40. Diferentes formas de lectura de los dispositivos en la app. tinker.io desde el smartphone. Fuente: Elaboración propia.



Universidad de Valladolid

IMPLANTACIÓN, ESTUDIO Y CONTROL DE CALIDAD DE VARIABLES  
CLIMÁTICAS EN EL ÁMBITO DE LA METROLOGÍA 4.0.

Capítulo 3. Implementación Industria 4.0: Arduino y Plataforma  
Thingier.io.



ESCUELA DE INGENIERÍAS  
INDUSTRIALES



Universidad de Valladolid

IMPLANTACIÓN, ESTUDIO Y CONTROL DE CALIDAD DE VARIABLES  
CLIMÁTICAS EN EL ÁMBITO DE LA METROLOGÍA 4.0.

Capítulo 4. Instrumentación.



ESCUELA DE INGENIERÍAS  
INDUSTRIALES

# **CAPÍTULO 4.**

# **INSTRUMENTACIÓN.**



## 4.1 PLACA NODEMCU.

Remontándose a los inicios del desarrollo de NodeMCU, se refería a un firmware de código abierto (Open Source) basado en hardware y software sencillo y simple de usar. Pero, actualmente, el término de **NodeMCU** se le atribuye a la **placa de desarrollo**. El firmware NodeMCU apareció poco después de surgir el ESP8266, el 30 de Diciembre de 2013. Posteriormente, en Octubre de 2014 fue publicada la primera versión de la placa NodeMCU, llamada devkit v0.9, la cual era también de hardware abierto.

Comenzaron a surgir alternativas de programación para el ESP8266, entre ellas la plataforma de Arduino a través del lenguaje C++. Pero, conforme fue pasando el tiempo, el desinterés ascendente en cuanto al firmware es lo que hizo que la placa de desarrollo adquiriese en su totalidad el concepto de NodeMCU; no obstante, debe tenerse presente que tanto el firmware como la placa han contribuido al apogeo de la implantación del ESP8266.

Centrándose en la placa NodeMCU, está se basa en el ESP12E, al cual le confiere una serie de ventajas a mayores de las que este tiene:

- Puerto micro USB y conversor Serie-USB.
- Programación sencilla a través del MICRO-USB.
- USB para la alimentación.
- Terminales [pines] para facilitar la conexión.
- LED y botón de RESET integrados.

Las placas NodeMCU dependiendo del tipo de fabricante pueden tener alguna discrepancia en lo que a designación de los pines se refiere. Se tienen 3 versiones:

- **PRIMERA GENERACIÓN V0.9**

Es la versión original del NodeMCU y se denominó devkit 0.9. Contaba con un ESP12, el cual es similar al ESP12E, pero éste último cuenta a mayores con una hilera de pines, por lo que dispone de más GPIO.

- **SEGUNDA GENERACIÓN V1.0/V2**

La principal diferencia respecto de la anterior, es la inclusión del ESP12E, en vez del ya comentado ESP12, por lo que se dispone de más pines y en cuanto a diseño se refiere, es más estrecha.

▪ **TERCERA GENERACIÓN V1.0/V3**

Se creó para introducir ciertas mejoras respecto de la anterior. La principal novedad es que integra un conversor serial CH340G, confiriendo mayor robustez al USB. Como inconveniente a la hora del montaje es que vuelve a ser más ancho que la placa anterior.

En la Tabla 1 pueden verse resumidas estas tres versiones.

GENERACIÓN	VERSIÓN		MÓDULO	Observaciones
Primera	0.9		ESP12	La original no está a la venta.
Segunda	1.0	V2	ESP12E	Es la versión "oficial"
Tercera	1.0	V3	ESP12E	Conversor CH340G

Tabla 1. Las tres versiones de la placa NodeMCU.

Para diferenciar una V2 de una V3, hay que fijarse en el conversor serial que dispone la placa, el CH340G es más alargado que el CP2102, como puede verse en la Figura 41, el estudio experimental de este TFG es llevado a cabo con la V3.

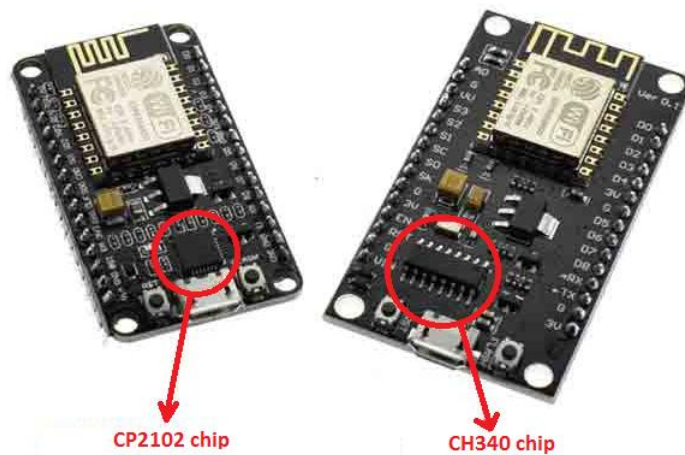


Figura 41. Identificación de las versiones V2 y V3 conforme a los chips, de izquierda a derecha respectivamente. [22] NODEMCU, La popular placa de desarrollo con ESP8266. Junio, 2018.

**4.1.1. PLACA EMPLEADA EN LA EXPERIMENTACIÓN DEL TFG.**

La placa que se ha utilizado es la siguiente: **Placa NodeMCU v3 – ESP8226**, véase Figura 42.

Se trata de una pequeña placa con conexión WiFi integrada, para proyectos de IoT fundamentada en el ESP8266 / CH340G y con una serie de pines dispuestos en los laterales de la misma. Lleva además consigo un regulador de tensión a la par que un puerto USB y puede ser programada bien por el IDE de Arduino o por LUA.

Gracias a llevar un convertor CH340G, por lo general no se precisa de la instalación de ningún driver para su instalación, realizándose ésta de forma automática.

Características:

- 9 pines GPIO con I<sup>2</sup>C y SPI.
- Función auto-reset y pulsador manual de RESET (RST).
- Procesador: ESP8266, 80 MHz, 3,3V (ESP-12E).
- Regulador de 3,3 V (500 mA).
- Convertor USB-Serial CH340G y conexión WiFi de 802.11 b/g/n.
- Una entrada de alimentación externa, VIN (20V máximo).
- 4 MB de memoria FLASH (32 MBit).
- En las esquinas lleva perforados 4 agujeros de montaje de unos 3 mm de diámetro.

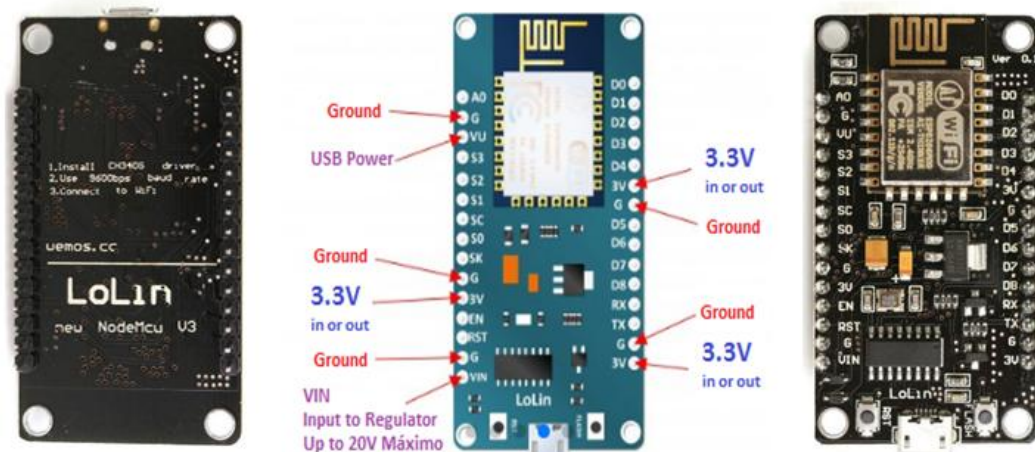


Figura 42. Placa NodeMCU v3- ESP8226 empleada para el desarrollo experimental, vista anterior (izqda.) y vista frontal (dcha). Fuente: Elaboración propia.



## 4.2 SENSORES.

Los sensores empleados para el estudio serán el sensor DHT22 y el BME280.

Se citará a continuación el marco en el que cada uno se sitúa, sus características y entorno en la programación.

En primer lugar, los sensores de la familia DHT, son capaces de suministrar de forma digital la temperatura y la humedad relativa. Se pueden encontrar dos tipos: DHT11 y DHT22.

### - Características del DHT11:

- Muy barato.
- Tensión de alimentación de 3-5 V.
- Corriente máxima de 2,5 mA en la adquisición de datos.
- El rango de temperatura es de 0-50 °C  $\pm$  2 °C de precisión.
- El rango de humedad es de 20-80 % con un 5 % de precisión.
- Frecuencia de muestreo máxima 1 Hz. (Capaz de leer 1 dato por segundo).
- Dimensiones: 15,5x12x5,5 mm.
- Bajo consumo.
- 4 pines con 0.1" de espacio.

### - Características del DHT22:

- Bajo coste.
- Voltaje de 3,3-5,5 V DC.
- Corriente máxima de 2,5 mA en la adquisición de datos.
- Óptimo para lecturas de humedad en un rango de 0-100 % con precisión de 2-5 % y temperaturas de -40-80 °C con una precisión de  $\pm$  0,5 °C.
- Frecuencia de muestreo máxima 0,5 Hz. (Capaz de leer 1 dato cada 2 segundos).
- Dimensiones: 15,1x25x7,7 mm.
- Bajo consumo de energía.
- 4 pines con 0.1" de espacio.

#### 4.2.1. SENSOR DHT11.

En el DHT11 la versión con PCB dispone de 3 pines, en cambio la de sin PCB tiene 4 pines.

Los pines de la versión sin PCB son: VCC (alimentación), I/O (transmisión de datos), NC (no conecta, pin al aire) y GND (conexión a tierra) y se muestran en la Figura 43.

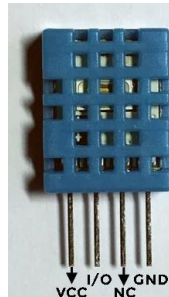


Figura 43. Pines del sensor DHT11 sin la placa de circuito impreso, PCB. [23]

La versión con PCB tiene 3 pines: VCC (alimentación), GND (conexión a tierra) y DATA (transmisión de datos) y se muestran a continuación en la Figura 44.



Figura 44. Pines del sensor DHT11 con la PCB. [23]

Se trata de un dispositivo analógico, aunque el pin sea digital, pues se produce en el propio dispositivo la conversión de analógico a digital. Esa señal digital es la que se emite al microcontrolador.

La resistencia Pull-UP de 5 k $\Omega$  es la aconsejable para un cable de longitud de hasta unos 20 m. No obstante, se deberá tener en cuenta la siguiente limitación, si la tensión de alimentación del DHT11 es superior a 3,5 V, el cable de alimentación no debe exceder de 20 cm. Y, además, es recomendable tomar medidas cada 5 s, ya que si se toman en menor tiempo puede verse disminuida la precisión. Es por ello que la muestra sobre la que se realiza el análisis comparativo entre los dos sensores, DHT 22 y BME280, se realizará en intervalos de 10 segundos, supliendo tal efecto.



En la conexión del DHT11 con PCB ya viene la resistencia Pull-UP integrada. No haría falta conectarla a la salida digital, como sí ocurriría si no se dispusiese de PCB.

#### **4.2.2. SENSOR DHT22.**

En particular el sensor que se utilizó fue un **sensor AM2302 DHT22**, el criterio que se sostuvo es que ofrecía un mayor rango en la recogida de datos y el grado de precisión que ofrece era mayor con un rango de tolerancia más reducido.

Se trata de un sensor de temperatura y humedad de bajo costo, con las características anteriormente citadas. Es muy simple. Su registro es lento, pero es muy accesible a nivel usuario, para poder registrar muestras de datos básicos. Dispone de dos partes, un sensor de humedad capacitivo y un termistor. En el interior dispone de un chip para la conversión de analógico a digital y emite esa señal digital de temperatura y humedad, que es fácilmente legible a partir de un microcontrolador.

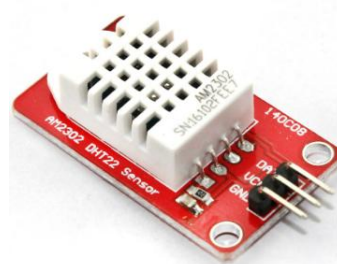


Figura 45. Sensor DHT22 empleado para la medida de temperatura y humedad relativa. Modelo: AM2302 DHT22.

Como se pudo comprobar el DHT22 es un sensor más preciso y ofrece un mayor rango de lectura de datos, no obstante ambos sensores, el DHT11 como el DHT22 son lentos, no pudiendo disponer de más de un dato por segundo o cada dos segundos, respectivamente.

Estos sensores disponen de un pin digital para emitir la información, con lo cual aminorarán el ruido, lo cual conlleva una gran ventaja a la hora de ser utilizado por parte del usuario.

Estos sensores se pueden encontrar de forma aislada, únicamente el sensor, o acompañado de una PCB [Printed Circuit Board], "Placa de Circuito Impreso". En cuanto al precio no existe una diferencia notable. La versión del

sensor insertado en una PCB aporta una resistencia Pull-UP y un LED. Más adelante se explicará en qué consiste esta resistencia Pull-UP.

#### **4.2.3. SENSOR BME280.**

El sensor **BMP280** permite medir la presión barométrica absoluta, y la temperatura. Existe, además, una versión del chip que integra la medición de humedad, a mayores de la presión y la temperatura, este chip es el **BME280** con el que se realizará el estudio comparativo respecto del sensor DHT22. Este sensor se conecta a un microcontrolador a través de I<sup>2</sup>C o bien por SPI.

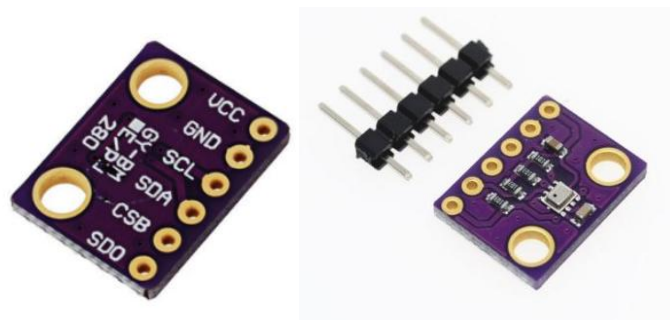


Figura 46. Sensor BME280 vista posterior (izqda.) y anterior (dcha). Modelo: GY-BM E/P 280.

#### **Características técnicas del BME 280:**

- Rango de presión: 300 – 1100 hPa.
- Rango de temperatura: -40 – + 85 °C.
- Rango de humedad relativa: 0-100 %.
- Precisión absoluta de la presión del aire:  $\pm 1$  hPa.
- Precisión absoluta de la temperatura:  $\pm 1,0$  °C.
- Precisión de humedad relativa:  $\pm 3$  %.
- Tensión de alimentación VDD: 1,71 – 3,6 V.
- Consumo de corriente: 2,8  $\mu$ A para medir presión y temperatura; 3,6  $\mu$ A para medir humedad, presión y temperatura y 0,1  $\mu$ A en modo de reposo.
- Interfaz de comunicación: I<sup>2</sup>C [hasta 3,4 MHz] y SPI [hasta 10 MHz].
- Muy bajo nivel de ruido, alta resolución y bajo consumo de energía.

En el programa para la lectura de datos, a través de la función **begin()** se comprobará si se encontró el sensor, devolviendo True si se ha encontrado, o por el contrario, False si no se encontró. En este último caso ha de revisarse el cableado para detectar el posible error de funcionamiento.



La temperatura estará en Grados Celsius y la presión en Pascales. También, existe la posibilidad de transformar el BME280 en un altímetro, si se conoce la presión al nivel del mar, pudiéndose así obtener la altitud a partir del dato de la presión barométrica.

#### **4.2.4. RESISTENCIAS.**

##### **4.2.4.1. INTRODUCCIÓN Y CARACTERIZACIÓN DE LAS RESISTENCIAS.**

Las resistencias que se pueden encontrar son resistencias comunes dispuestas en un circuito de una forma determinada. Dependiendo de la configuración se estará frente a una resistencia de tipo **Pull UP** ó **Pull DOWN**.

Se encargan de implantar un estado lógico en un pin o entrada de un circuito cuando el mismo está en pausa. La resistencia Pull-UP determina un estado high [alto], mientras que la resistencia Pull-DOWN establece un estado low [bajo] cuando el pin se encuentra parado. Todo ello contribuye a la **atenuación del ruido** existente en el propio circuito.

Mediante pulsadores se puede cambiar el valor de la entrada digital procedente del microcontrolador para obtener el voltaje que se desee, añadiendo al circuito una resistencia, bien en forma Pull-UP o Pull-Down.

Cuando se habla del término up ó down se refiere a la configuración, no son dos resistencias diferentes, sino los modos de actuación para la señal de respuesta que se prefiera conseguir.

Es en función del valor del voltaje en el estado de reposo, cuando el pulsador no se encuentra accionado, que se quiera tener para una lectura adecuada, lo que determina la selección de uno u otro tipo.

##### **4.2.4.2. RESISTENCIA PULL-UP.**

En esta configuración, cuando el pulsador se encuentra abierto (en reposo), véase imagen derecha (Figura 47), la entrada a Arduino será de +5 V. Mientras el pulsador esté accionado el voltaje a la entrada de Arduino es nulo, es decir, cuando se presiona el pulsador Arduino leerá un 0, véase imagen izquierda (Figura 47), y cuando el pulsador se encuentra en reposo leerá un 1.

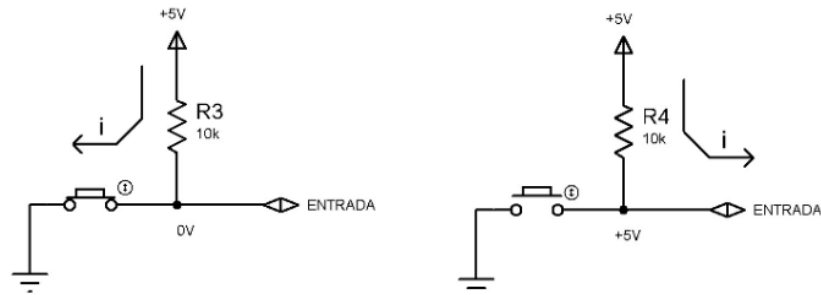


Figura 47. Configuración resistencias Pull-UP cuando el interruptor está accionado (izqda.) y cuando se encuentra abierto (dcha.). [24]

#### 4.2.4.3. RESISTENCIA PULL-DOWN.

De modo contrario a lo que ocurría con la resistencia Pull-UP, cuando el pulsador se encuentra abierto (en reposo), la entrada a Arduino será de 0 V, véase imagen izqda. (Figura 48); mientras que cuando el pulsador esté accionado el voltaje a la entrada de Arduino es de +5 V, véase imagen dcha. (Figura 48), o lo que sería lo mismo decir, cuando se presiona el pulsador Arduino leerá un 1 y cuando el pulsador se encuentra en reposo leerá un 0.

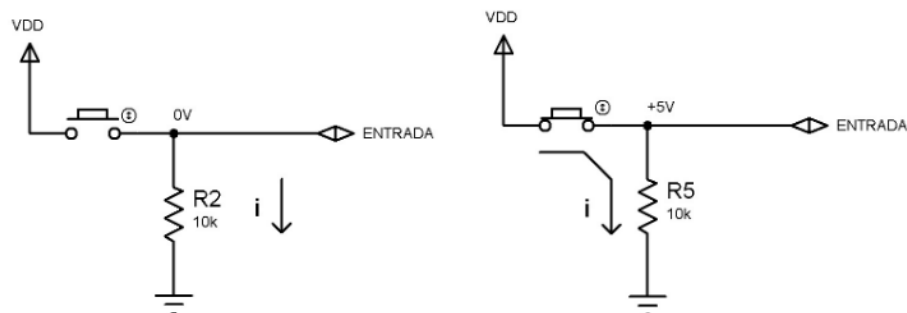


Figura 48. Configuración resistencias Pull-DOWN cuando el interruptor está accionado (dcha.) y cuando se encuentra abierto (izqda.). [24]

#### 4.2.4.4. ELECCIÓN DE RESISTENCIAS.

La configuración que más se suele usar es la de Pull-UP, no obstante, no existen diferencias técnicas apreciables entre una u otra. Muchos de los microcontroladores de Arduino integran resistencias Pull-UP, lo que hace que su uso sea más extendido. Estas se pueden activar mediante llamadas a través de funciones, como es la función **pinMode()**, en el propio IDE de Arduino, a través de la siguiente instrucción:

```
pinMode(pin, INPUT_PULLUP); // declaración de un PIN de entrada y
activación de la resistencia de configuración PULLUP para ese PIN
```

Las resistencias oscilan entre 1-10 K $\Omega$ , (corriente de 5-0,5 mA) siendo menor la velocidad de respuesta del pin a mayor resistencia y, además, disminuyendo su atenuación al ruido eléctrico.

Las resistencias Pull-UP se emplean para asegurar el nivel lógico tipo alto (High) en el caso de ausencia de señales digitales.

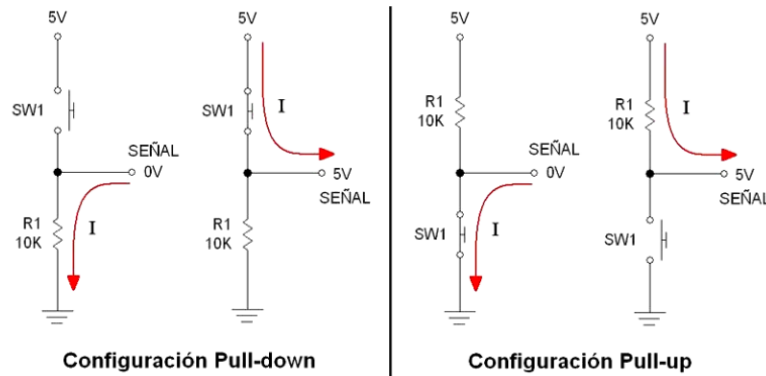


Figura 49. Resumen configuraciones resistencias Pull-DOWN y Pull-UP.

#### **4.2.5. LIBRERÍAS EN EL ENTORNO DE ARDUINO**

En proyectos multidisciplinares resulta muy útil hacer uso de bibliotecas que ensamblen una serie de archivos para simplificar el código en el entorno de programación, estos archivos contendrán especificaciones con diversas funcionalidades de modo que permitan a mayores de facilitar la programación, aumentar la modularidad y la reutilización de códigos en diferentes programas. Las librerías se insertan en la parte superior del programa. De los muchos ficheros que pueden conformarlas, al menos son dos los existentes en toda librería: un fichero con extensión .h (cabecera) y otro con extensión .cpp (con la implementación). El uso de estas librerías permitirá acortar sumamente las líneas de código, pues llevan implícitas código con funciones semejantes, de modo que permiten establecer la comunicación con sensores, placas, entre otros dispositivos.

En este apartado, se tratarán a continuación, las librerías empleadas para los sensores utilizados en este TFG.



#### 4.2.5.1. LIBRERÍAS SENSORES DHT.

Ciertas librerías en el entorno de Arduino permitirán obtener la información deseable de temperatura y humedad, se tratan de las que proporciona el fabricante Adafruit. Primeramente se importa la librería DHT.h:

```
#include <DHT.h> //Se incluye la librería correspondiente al sensor tipo DHT
```

Segundo se definen dos constantes, una donde se indica el PIN donde se ha conectado el DHT22 [pin digital] y otra, que hará referencia al tipo de sensor, DHT22. Esta librería es la misma tanto para este sensor como para los de la familia DHT. Y, por último, se declara el objeto DHT junto con los parámetros pin y tipo de DHT.

```
#define DHTPIN 4 //Se define el pin digital donde irá unido el sensor  
#define DHTTYPE DHT22 //Se explicita el tipo de sensor empleado, en este caso el DHT22
```

```
DHT dht(DHTPIN, DHTTYPE); //Se inicializa el sensor DHT22
```

Se utiliza la función **delay()**, llamándola con `delay(5000)` para esperar los 5 s aconsejables que se señalaron antes. La librería citada de Adafruit suministra datos, tanto en grados Celsius como en Fahrenheit. Para la obtención de los datos, en ambos casos, se utiliza la función, `readTemperature()`, devolviendo la temperatura en grados Celsius, pero si se quisiera obtener la temperatura en grados Fahrenheit se añade **true** dentro del paréntesis de la función, `readTemperature(true)`. Para el caso de la humedad se actúa de la misma manera, llamando a la función `readHumidity()`.

También, se debe realizar la comprobación de que la información no sea desacertada y que devuelva un número, para lo que se alude a la función **isnan()** la cual devuelve verdadero si no se trata de un número (`isnan`, Is Not a Number) y falso en caso contrario.

#### 4.2.5.2. LIBRERÍAS SENSOR BME280.

El sensor BME280 dispone de otra librería específica para su puesta en marcha; la descarga de la misma se puede realizar a través del Gestor de Librerías de Arduino, en que se seleccionará la versión más reciente del fabricante Adafruit, véase Figura 50.

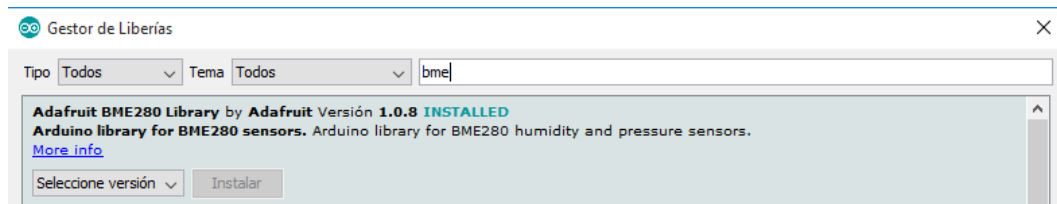


Figura 50. Descarga de la Librería Adafruit BME280 desde el Gestor de Librerías de Arduino.

Para visualizar su aplicación, se expondrá un ejemplo con esta librería, para ello se subirá a la placa primeramente el programa por medio del siguiente Sketch, Figura 51.

The image shows the Arduino IDE interface with a sketch loaded. The sketch file is named 'sketch\_BME280\_SCL\_20022019'. The code includes headers for SPI, BME280, and Adafruit\_BME280. It defines pins for SCK, MISO, and CS, and sets up the SPI interface. The setup function initializes the serial port and prints a test message. The loop function prints the sensor's values. The bottom of the window shows the serial monitor output with the text 'BME280 test' and some numerical data in brackets.

```
sketch_BME280_SCL_20022019
#include <SPI.h>
#include <BME280.h>
#include <Adafruit_BME280.h>
#include <Adafruit_BME280.h>

// assign the ESP8266 pins to arduino pins
#define D3 5
#define D2 4
#define D4 2
#define D0 0

// assign the SPI bus to pins
#define BME_SCK D1
#define BME_MISO D4
#define BME_MOSI D2
#define BME_CS D3

#define SEALEVELPRESSURE_HPA (1013.25)

Adafruit_BME280 bme(BME_CS, BME_MOSI, BME_MISO, BME_SCK); // software SPI

unsigned long delayTime;

void setup() {
  Serial.begin(9600);
  Serial.println("BME280 test");

  bool status;
  // default settings
  status = bme.begin();
  if (!status) {
    Serial.println("Could not find a valid BME280 sensor, check wiring!");
    while (1);
  }

  Serial.println("-- Default Test --");
  delayTime = 2000;
  Serial.println();
}

void loop() {
  printValues();
}
```

Figura 51. Carga del programa del sensor BME280 a la placa.

En este Sketch se puede ver cómo la llamada a la librería `Adafruit_BME280` se establece en la cabecera del programa a través de `#include <Adafruit_BME280.h>`.

Una vez se ejecuta el programa, se podrán visualizar los datos a través del Monitor Serie por pantalla:

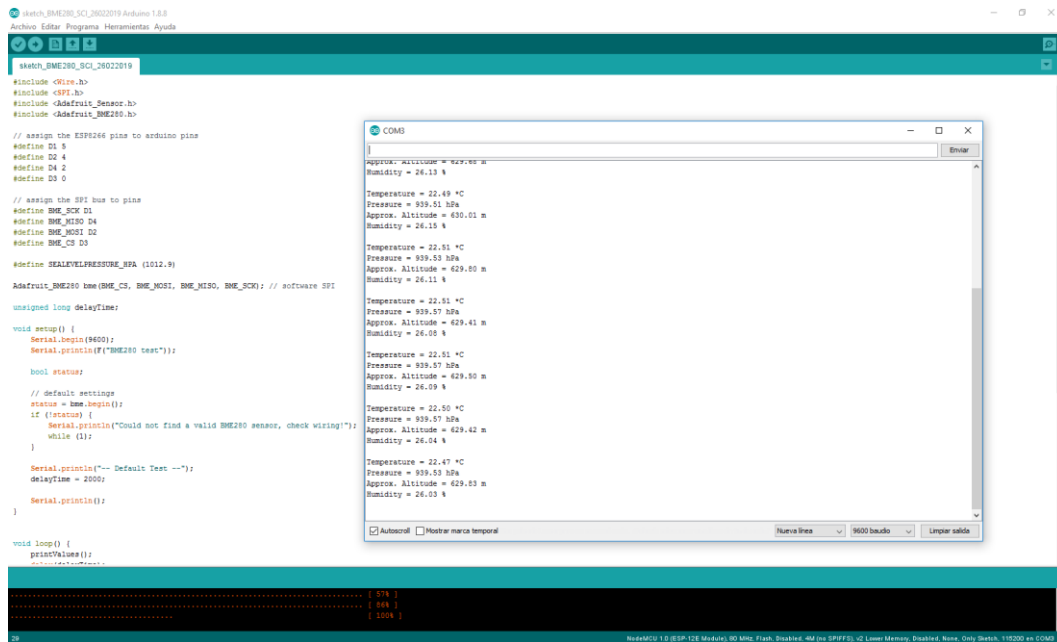
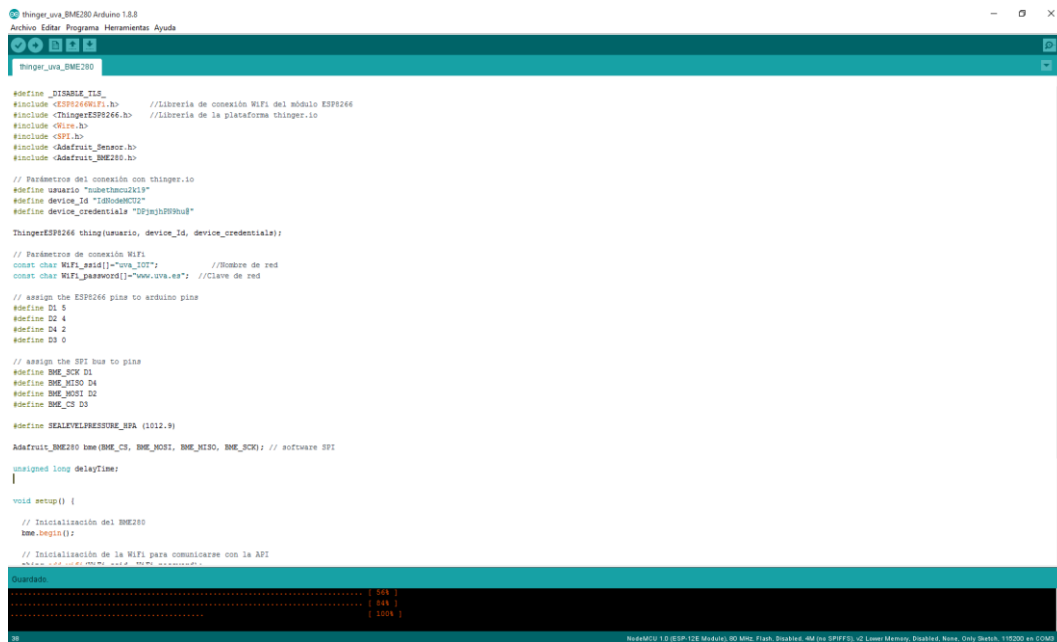


Figura 52. Ejecución programa subido a la placa y su visualización a través del monitor serie.

Se sube, a continuación, un nuevo Sketch, véase ANEXO [P3.], para establecer la conexión con Thinger.io de esta nueva placa, :





```
thingy_uva_BME280 Arduino 1.8.8
Archivo Editar Programa Herramientas Ayuda

thingy_uva_BME280
#define WEMOTE_UVA_ARDUINO_BME280
#define device_credenciales "Dfjmh299hu8"

ThingyESP8266 thing(usuario, device_id, device_credenciales);

// Parámetros de conexión WiFi
const char WIFI_SSID[]="uva_10T"; //Nombre de red
const char WIFI_PASSWORD[]="www.uva.es"; //Clave de red

// assign the ESP8266 pins to arduino pins
#define D1 5
#define D2 4
#define D4 2
#define D3 0

// assign the SPI bus to pins
#define BME_SCK D1
#define BME_MISO D4
#define BME_MOSI D2
#define BME_CS D3

#define SEALVELELPRESSION_HPA (1012.9)

Adafruit_BME280 bme(BME_CS, BME_MOSI, BME_MISO, BME_SCK); // software SPI

unsigned long delayTime;

void setup() {
  // Inicialización del BME280
  bme.begin();

  // Inicialización de la WiFi para comunicarse con la API
  thing->add_wifi(WIFI_SSID, WIFI_PASSWORD);

  // Inicialización de la lectura de datos desde la API
  thing["Bme280"] >> {}(bme);
  out["Temperature"] = bme.readTemperature();
  out["Humedad"] = bme.readHumidity();
}

void loop() {
  thing->send();
}
```

Figura 53. Programa establecimiento conexión placa con sensor BME280 a la plataforma Thinger.io.

Ya establecida la conexión, se procedería como se citó previamente para la generación de un nuevo dispositivo, para instalar el nuevo dispositivo IdNodeMCU2 y comprobar a continuación que se encuentre conectado, véase Figura 54.

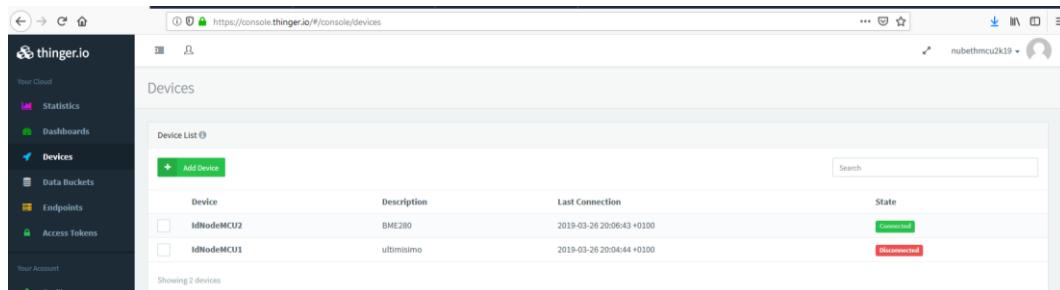


Figura 54. Instalación dispositivo IdNodeMCU2.



Universidad de Valladolid

# IMPLANTACIÓN, ESTUDIO Y CONTROL DE CALIDAD DE VARIABLES CLIMÁTICAS EN EL ÁMBITO DE LA METROLOGÍA 4.0.

## Capítulo 4. Instrumentación.



ESCUELA DE INGENIERÍAS INDUSTRIALES



## **CAPÍTULO 5.**

# **PARTE EXPERIMENTAL Y RESULTADOS.**



## 5.1 SISTEMA DE ADQUISICIÓN DE DATOS.

En este apartado se va a realizar una aplicación práctica de control y validación de variables dentro del ámbito industrial más concretamente enfocados en Metrología 4.0 que se desarrollará en el Laboratorio de Metrología de la Escuela de Ingenierías Industriales de la Universidad de Valladolid. Conocidos los datos de este local, se podrán caracterizar las medidas recogidas a través de sensores, en referencia al mismo.

Para ello se empleará Arduino y la plataforma Thingier.io que se explicó anteriormente, aunque podría haberse empleado otro sistema de programación como puede ser Python o LUA. No obstante, Arduino es una plataforma muy compacta que permite realizar grandes proyectos dentro del mundo de la electrónica, la programación y la robótica, basados en una sencilla placa programable, a través del lenguaje de programación Processing.

## 5.2 PLANTEAMIENTO.

Para el estudio se va a tomar una muestra de datos comprendida entre las 18:57 del día 08/04/2019 hasta las 10:57 del día 24/04/2019, más o menos unas dos semanas para poder extraer las conclusiones más precisas. El análisis que se va a detallar a continuación, comprende las variables de temperatura y humedad relativa obtenidas del sensor DHT22 y las mismas variables, añadiendo la presión, para el caso del sensor BME280.

Lo que se pretende es caracterizar el comportamiento de ambos sensores, aportando resultados concisos tras el estudio que se ha planteado, con el fin último de proporcionar una visibilidad certera del gran avance y utilidad que proporcionan dispositivos económicos y al alcance de todo ser humano que sin conocimiento de ellos no resultaría viable poder llegar a conclusiones de gran utilidad para procesos que requieren de esta información, bien para ayudarlos a optimizar su cadena productiva, las condiciones medioambientales en una sala o en un laboratorio donde se quieran estabilizar estos valores, dentro de un rango determinado, pudiendo tomar decisiones y actuar conforme los resultados que estemos observando por pantalla a través de un medio electrónico, entre otros muchos usos.

La muestra a la que se someterá al análisis, consta de unos 2.243 números. La periodicidad con la que se han seleccionado los datos ha sido de intervalos

de 10 minutos, resultando claramente viable no disponer de datos en un período más corto de tiempo, ya que el riesgo en este estudio comparativo no exige de una gran responsabilidad y su leve desviación sería efímera en el caso de haberla. En el caso de requerirse otros intervalos de tiempo será posible gracias a la programación disponible en la plataforma Thingier.io, donde los datos recogidos podrán programarse siendo 1 minuto el intervalo mínimo comprendido entre una lectura y otra de los datos o bien haciendo uso de los medios de ahorro energético que se trataron en el capítulo 3.

### 5.3 ESTADO INICIAL.

Para comenzar el análisis se procederá a la toma de los datos, suministrados éstos por los Data Buckets de la plataforma Thingier.io.

Previamente se crearon dos dispositivos, del modo en que se reseñó en el apartado 3.2, un primer dispositivo, NodeMCU1 para la placa con el sensor DHT22 y otro, llamado NodeMCU2 para la una segunda placa, unida la misma junto al sensor BME280. La exportación de datos se realizará a la hoja de cálculo Excel, en donde se grafican los valores de las variables frente al tiempo, obteniéndose las siguientes gráficas:

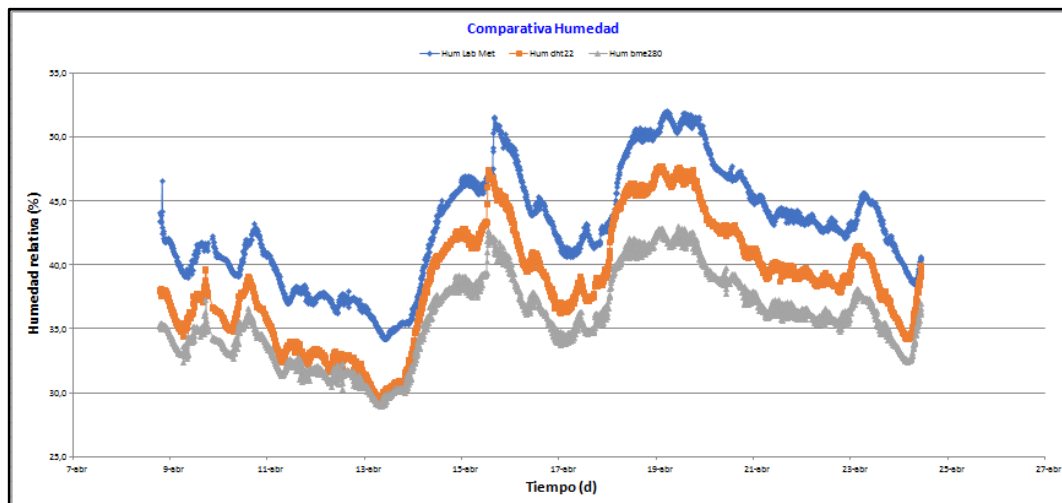


Figura 55. Gráfico comparativo de la humedad relativa medida por los sensores respecto de la referencia (curva azul).

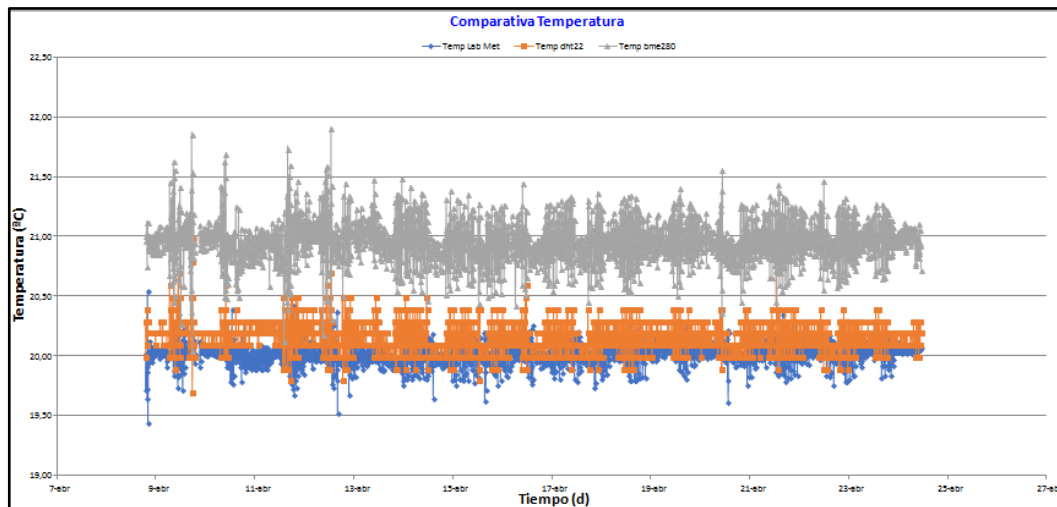


Figura 56. Gráfico comparativo de la temperatura medida por los sensores respecto de la referencia (curva azul).

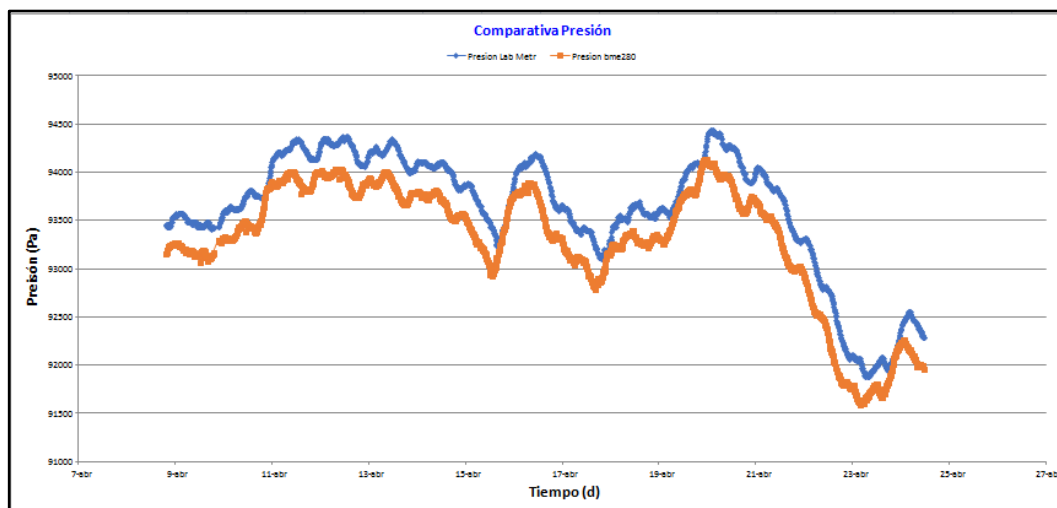


Figura 57. Gráfico comparativo de la presión medida por el sensor BME280 respecto de la referencia (curva azul).

En vistas a las series temporales representadas en las gráficas anteriores, puede apreciarse un bajo decalaje en cuanto a la magnitud medida entre los sensores y la referencia Laboratorio, a lo que podría sumarse un pequeño retardo temporal, si se realiza la comparativa en el eje de abscisas, este retardo como puede observarse podría ser debido a una mayor velocidad de respuesta del sensor ante variaciones del medio en que esté situado.

También las gráficas permiten observar que el sensor DHT22 (serie temporal en color naranja) se aproxima mejor a la referencia, el Laboratorio (serie temporal en color azul), lo cual lleva a admitir que en caso de optar por uno u otro sensor, se escogerá el DHT22, por ser más preciso, aunque



posteriormente se observará que esta afirmación optada a priori, se podría ver refutada si se introduce un factor de calibración que contrarreste el error sistemático.

Cuando se corrija a continuación el error sistemático en la operación de ajuste que se realizará sobre la muestra de datos, este ajuste será mayor en el caso del sensor BME280 como cabía esperar con la suposición primera planteada (mayor precisión en el otro sensor, en el DHT22). Pero, cuando se proceda a realizar la segunda operación, "calibración" nos encontraremos con el error entre los sensores y la referencia, ambos con sus respectivas correcciones, lo que arrojará resultados que en una primera instancia no eran los sospechados antes de operar sobre ambos instrumentos.

Una vez que se dispone de las gráficas para cada uno de los sensores junto a sus variables, temperatura y humedad, a la que se añadió la de presión al sensor BME280, se realizarán una serie de operaciones, en el siguiente orden para llegar a determinar la incertidumbre asociada al conjunto de datos registrados.

#### 5.4 ETAPA DE AJUSTE.

En esta primera fase se realizará un ajuste (estimación) del sensor, en el que mediante la Ecuación [1] se llegará a la representación del mismo, evidenciando la diferencia o ajuste del sensor, es decir, la diferencia que hay entre la referencia ya corregida con la lectura del sensor, procedente esta última de los datos recogidos directamente de la plataforma Thinger.io.

$$\text{ajuste} = T_{\text{sensor},i} - T_{\text{ref.Co},i} \quad \text{Ecuación [1]}$$

siendo la temperatura del sensor,  $T_{\text{sensor}}$  :

$$T_{\text{sensor},i} = T_{\text{lectura}} + C_{ti} \quad \text{Ecuación [2]}$$

en donde  $C_{ti}$  será la corrección por temperatura asociada al sensor  $i$ , es decir, al instrumento empleado para la medida del mensurando. Esta corrección es la que se introducirá en la programación del software a través de una variable (factor de corrección) que se añadirá a la lectura del instrumento, con lo que la lectura del sensor en la plataforma Thinger.io proporcionaría las gráficas más próximas a la realidad, con un rebaje del error sistemático implícito en el propio instrumento de medida, en los propios sensores en este caso.

Las correcciones por temperatura y humedad relativa de los instrumentos del laboratorio con que se llevará a cabo el estudio comparativo, se extraerán del Certificado de Calibración del instrumento que se ha utilizado como referencia



[Nº 26/LC 10.017 y número 145517]. En este caso, se trata del Termohigrómetro situado en el Laboratorio de Metrología y Calibración Dimensional (LCD) de la Universidad de Valladolid, sede Paseo del Cauce. En el rango especificado por la lectura del dispositivo, la calibración ha sido realizada con una sonda externa de temperatura y humedad: Código E1/108, conectada con el canal M5, conforme al procedimiento establecido en la Escala Internacional de Temperatura de 1990 (EIT-90), según la cual es realizada por comparación en el interior de un generador de temperatura y humedad.

La corrección se refiere a la cantidad que habría que sumar algebraicamente a la medida proporcionada por el instrumento, para de este modo poder obtener el verdadero valor convencional.

Los patrones e instrumentos empleados para tal calibración disponen de trazabilidad adecuada, verificada y constatada por ENAC.

Por otro lado, si no fuese este el organismo por el que se hubiese hecho la calibración, se debería ir comprobando para atrás con qué patrones se han ido calibrando hasta finalmente llegar a un certificado que estuviese avalado por ENAC u otra entidad de EA (European cooperation for Accreditation) para probar así la correcta trazabilidad del equipo de medida.



**Resultados de calibración de la sonda código E1/108, conectada en el canal M5**

*Results probe E1/108, connected to channel M5.*

TEMPERATURA TEMPERATURE			
Temperatura Patrón Standard Temperature ( $t_{90}$ ) °C	Medias Mensurando Measurand average °C	Corrección Correction (Patrón - Mensurando) (Reference-Measurand) °C	Incertidumbre Expandida Expanded Uncertainty °C
15,00	15,06	-0,06	0,20
19,97	20,04	-0,07	0,20
29,92	29,91	0,01	0,20
20,17	20,18	-0,01	0,20

Figura 58. Resultados calibración del Termohigrómetro empleado referentes a la variable temperatura.



HUMEDAD RELATIVA A 23 °C RELATIVE HUMIDITY AT 23 °C			
Humedad Patrón Standard Humidity	Medias Mensurando Measurand average	Corrección Correction (Patrón - Mensurando) (Reference-Measurand)	Incertidumbre Expandida Expanded Uncertainty
%hr	%hr	%hr	%hr
25,2	28,0	-2,8	1,6
49,7	50,8	-1,1	1,9
79,0	79,1	-0,1	2,3
50,0	51,3	-1,3	2,0

Figura 59. Resultados de calibración correspondientes a la variable humedad relativa.

En el caso de la HUMEDAD RELATIVA la corrección media que habría que sumar a las indicaciones de la referencia, al estar comprendida esta variable entre 30-50 %, será de -1,73 % como se indica a continuación, recogiendo para ello los datos que abarquen este rango:

$$\frac{(-2,8 - 1,1 - 1,3)}{3} = -1,73 \%$$

Tras corregir la referencia (indicación del Lab. De Metrología) la gráfica que resultó para la Humedad Relativa fue la mostrada en la **¡Error! No se encuentra el origen de la referencia.**

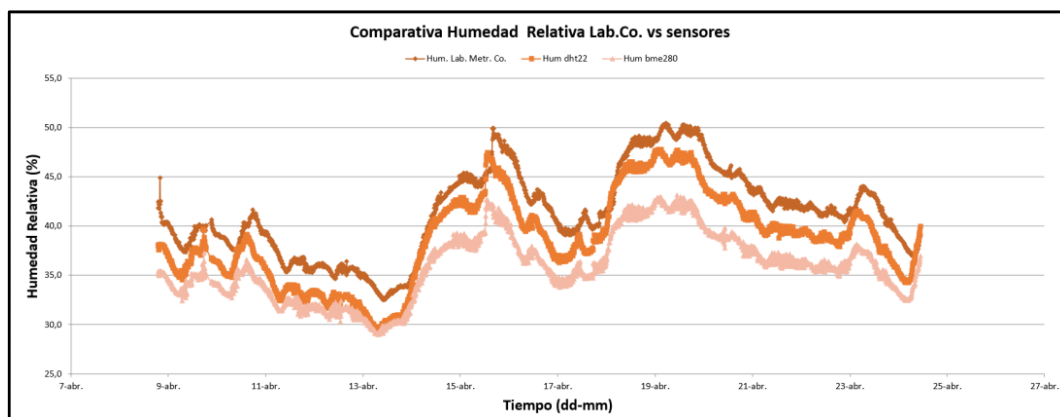


Figura 60. Gráfico comparativo de la humedad relativa medida por los sensores respecto de la humedad relativa corregida del Laboratorio.

Será a partir de las siguientes gráficas de ajuste, sobre las que se determinará el valor de la corrección que habrá que utilizar en la segunda etapa de este análisis.

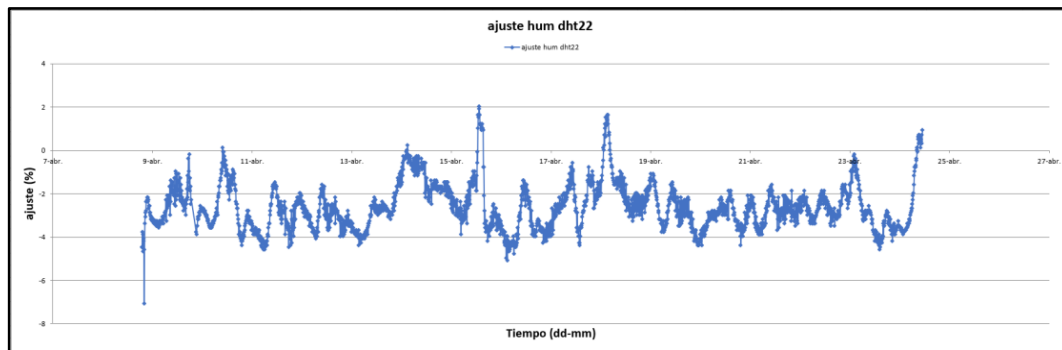


Figura 61. El ajuste existente entre la humedad relativa medida por el sensor DHT22 y la referencia corregida.

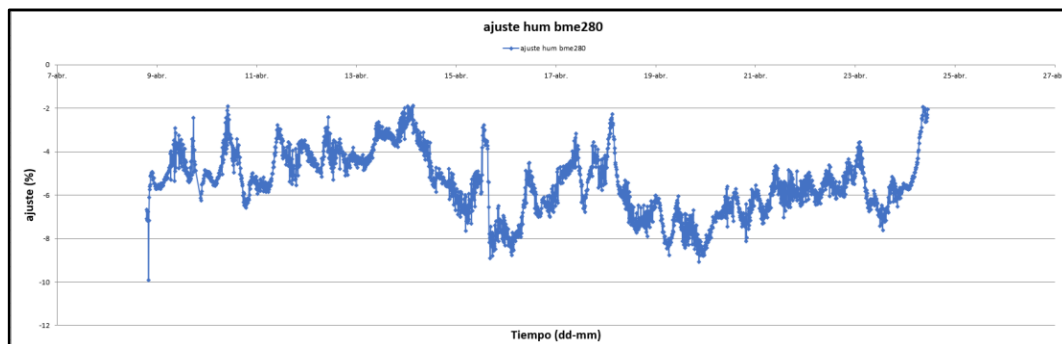


Figura 62. El ajuste existente entre la humedad relativa medida por el sensor BME280 y la referencia corregida.

Conforme a las gráficas del ajuste anteriores, la corrección para la humedad relativa de cada uno de los sensores será de +2,6 % para el sensor DHT22 y de +5,45 % para el sensor BME280 como se puede abstraer de forma cualitativa o bien realizando el ajuste promedio.

Para el ajuste de la TEMPERATURA, se procederá de la misma manera, corrigiéndose la curva correspondiente a la temperatura del Laboratorio conforme a la tabla de la Figura 58, según la cual la corrección media que habría que sumar sería:

$$\frac{(-0,06 + (-0,07) + 0,01 + (-0,01))}{4} = -0,0325 \text{ } ^\circ\text{C}$$

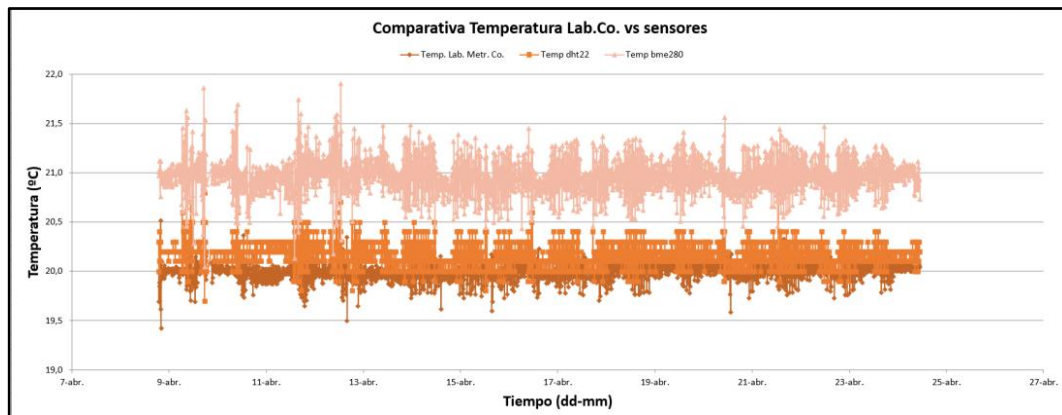


Figura 63. Gráfico comparativo de temperaturas, expresadas en ( $^{\circ}\text{C}$ ), una vez corregida la referencia, Laboratorio.

Una vez realizada la corrección de la temperatura a la referencia se procede a graficar ésta junto con las lecturas de la variable temperatura del sensor i para mostrar la curva de ajuste de cada uno de ellos.

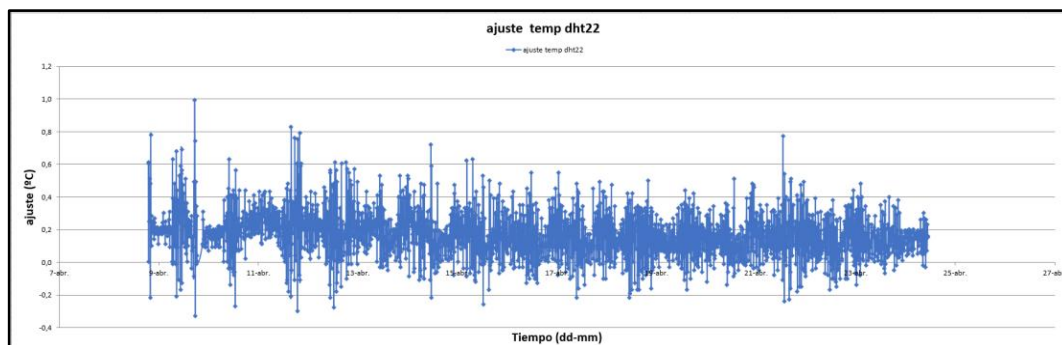


Figura 64. El ajuste existente entre la temperatura medida por el sensor DHT22 y la referencia corregida.

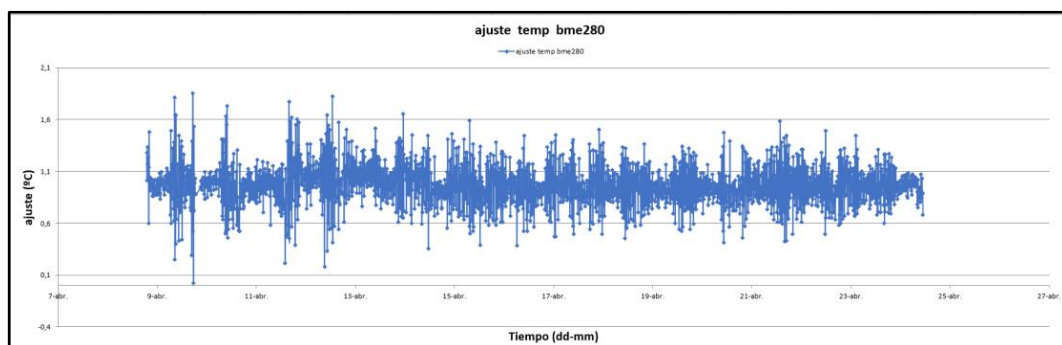


Figura 65. El ajuste existente entre la temperatura medida por el sensor BME280 y la referencia corregida.

En vistas a las gráficas del ajuste basadas en la Ecuación [1], se determinará a partir de ellas la corrección para el sensor DHT22, que será de  $-0,17\text{ }^{\circ}\text{C}$  y para el sensor BME280, de  $-0,97\text{ }^{\circ}\text{C}$  como se puede abstraer de forma

cualitativa o bien referenciándose en un estadístico como puede ser la media muestral, que al tener una cantidad tan numerosa de datos, ofrecerá una medida reseñable para poder tomar decisiones, ofreciendo un valor razonable del valor esperado. Se puede observar a mayores, que la corrección en cuanto a la temperatura es prácticamente nula, con lo cual podría omitirse, pero en este estudio se tendrá en cuenta para minimizar la incertidumbre en las conclusiones finales. No obstante, este valor de corrección aplicable, coincidirá con el del ajuste, pero cambiado de signo, pues la corrección o lo que habría que contrarrestar sería ese valor a los valores de lectura que se tenían de partida en la Hoja de cálculo de Excel.

Para la presión, la corrección del Medidor de Presión absoluta que se empleó como referencia, ubicado también el Laboratorio de Metrología y Calibración Dimensional (LCD) de la Universidad de Valladolid, sede Paseo del Cauce, se extraerá del Certificado de Calibración [Nº 26/LC 10.017 y número 145518].

Los patrones e instrumentos empleados para dicha calibración disponen de la trazabilidad adecuada, verificada y constatada por ENAC.

La corrección media para la presión se calculará a partir de los datos de la siguiente tabla extraída del certificado 145518.

**RESULTADOS**  
*Results*

Patrón <i>Standard Reference</i> (mbar)	Mensurando <i>Unit Under Test</i> (mbar)	Corrección <i>Correction</i> (mbar)	uef	k	Incertidumbre <i>Exp. Uncertainty</i> (mbar)
200,00	184,18	15,82	>100	2,00	0,44
600,00	597,80	2,20	>100	2,00	0,43
800,00	803,88	-3,88	>100	2,00	0,48
900,00	906,00	-6,00	>100	2,00	0,44
1050,00	1057,48	-7,48	>100	2,00	0,45

Figura 66. Resultados de calibración correspondientes a la variable presión.

El rango de presión se encuentra comprendido entre unos 900 y 1.050 mbar, por ello la corrección media para la PRESIÓN será de:

$$\frac{(-6 + (-7,48))}{2} = -6,74\text{mbar} \cdot \frac{1\text{bar}}{10^3\text{mbar}} \cdot \frac{10^5\text{Pa}}{1\text{bar}} = -674\text{ Pa}$$

De este modo se tendrá la siguiente gráfica, una vez corregida la presión del Laboratorio, que se empleará como referencia.

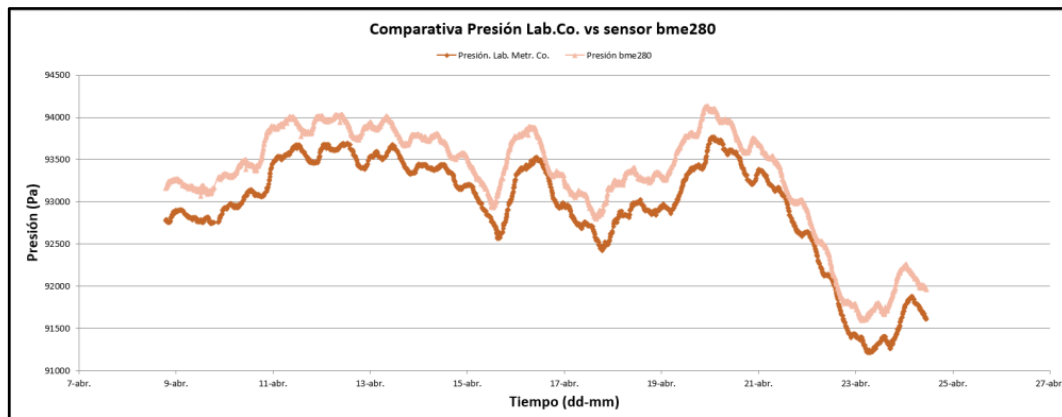


Figura 67. Gráfico comparativo de presión, expresada en (Pa), una vez corregida la referencia, Laboratorio.

En donde puede observarse que tras ser corregida la referencia, ésta estará ahora por debajo de la presión registrada por el sensor BME280, es por ello que habrá que restar  $-360$  Pa para que tanto la presión de referencia como la del sensor adquieran una forma lo más semejante posible, pudiéndose observar este valor en la gráfica siguiente del ajuste, Figura 68.

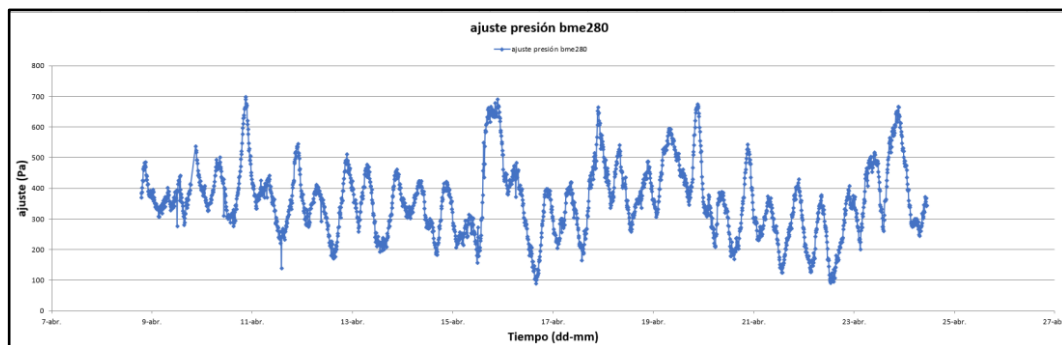


Figura 68. El ajuste existente entre la presión medida por el sensor BME280 y la referencia corregida.

## 5.5 CALIBRACIÓN.

Es en esta fase del estudio donde primeramente se presentarán las gráficas de las variables ya corregidas, tanto para la referencia como para los sensores, aplicando para estos últimos la corrección que se dedujo y explicitó en la etapa anterior de las gráficas del ajuste. Y, será a partir de las nuevas gráficas de error, con los valores corregidos, de donde se determinarán una serie de parámetros como son el error medio, la desviación típica del error, entre otros, útiles para la caracterización posterior de la incertidumbre y la función modelo que se desarrollarán en la última etapa.

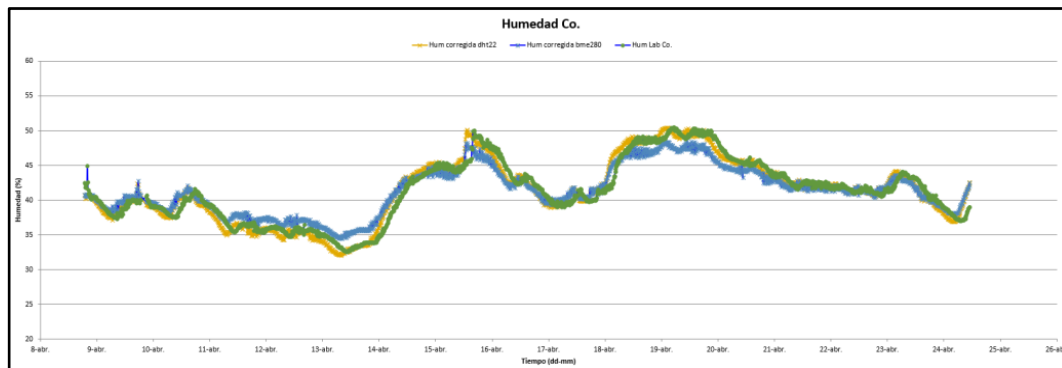


Figura 69. Gráfica comparativa de la evolución temporal de la humedad relativa corregida respecto de la de cada uno de los sensores con que se realiza el estudio.

En esta gráfica puede observarse cómo ambos sensores describen la misma tendencia temporal que la referencia patrón (línea verde), alejándose sensiblemente de la referencia en algunos puntos en los que podría no haberse estabilizado del todo el sensor a la hora de la lectura de los datos, no obstante deja clara evidencia de la tendencia que sigue la humedad relativa, pudiendo observarse que esta variable es mucho más sensible frente al tiempo que la temperatura, como podrá verse después. Se experimentan ciertos cambios en la tendencia de la humedad, debidos originalmente a la actividad física desarrollada dentro del aula, en días laborables, siendo el valor de la humedad relativa, expresada en %, ligeramente mayor los días de vacaciones (14-21 Abril, Semana Santa) en que no ha habido nadie en el laboratorio.

La humedad relativa se expresa en tanto por ciento, % e indica la relación existente entre la presión de vapor efectiva,  $e$ , y la presión de vapor máxima o saturante,  $E$ .

$$h_r = 100 \cdot \frac{e}{E} \quad \text{Ecuación [3]}$$

Una humedad relativa del 40% indica que faltaría un 60% de la masa de aire por saturar. Si disminuye la temperatura,  $e$  se mantiene, pero  $E$  disminuye, con lo que la humedad relativa,  $h_r$ , aumentaría. Por el contrario, si la temperatura aumentase, la  $h_r$  se aminoraría.

En definitiva, la humedad relativa, que se refiere a la cantidad máxima de vapor de agua que tiene el aire a una determinada temperatura, está estrechamente ligada a la variación de la temperatura y también a la presión. La humedad relativa es inversamente proporcional a la temperatura, un aumento de ella produce una disminución de la temperatura, y viceversa.

Siguiendo trabajando con todos los valores anteriores, se presentan a continuación las gráficas en las que se representará el error puntual o temporal,  $\varepsilon_i$ , conforme a la Ecuación [4].

$$\varepsilon_i \quad i = 1, \dots, n = [HR_{\text{sensor}} - HR_{\text{ref}}]_{\text{corregida}} \quad \text{Ecuación [4]}$$

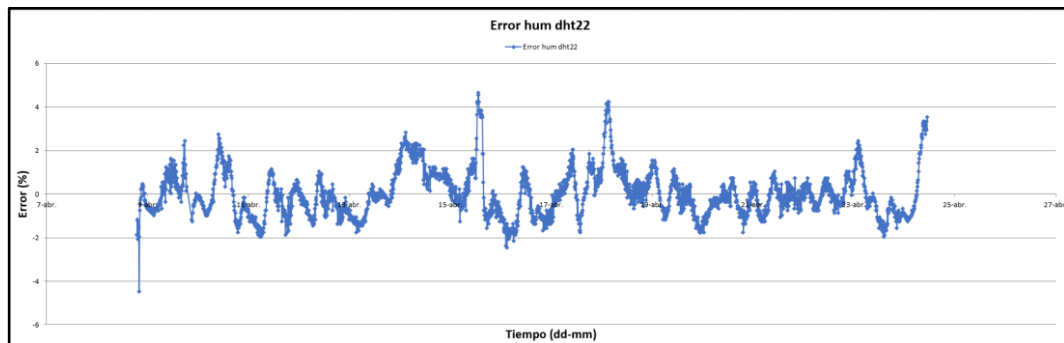


Figura 70. Error existente entre la humedad relativa medida por el sensor DHT22 y la referencia, ambos corregidos.

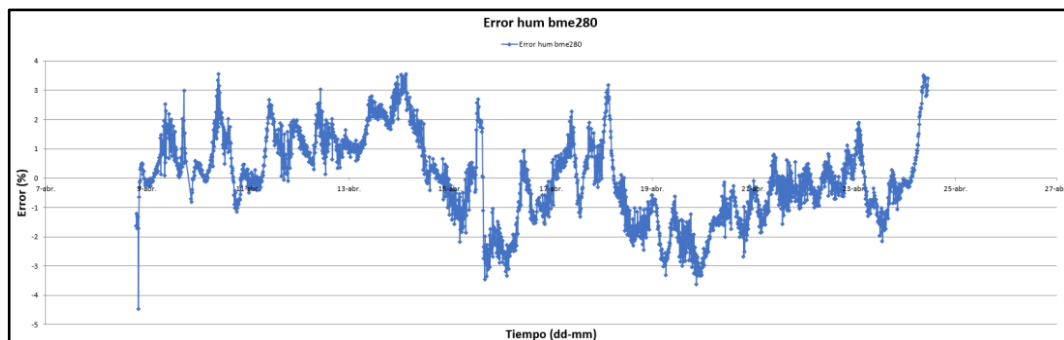


Figura 71. Error existente entre la humedad relativa medida por el sensor BME280 y la referencia, ambos corregidos.

Seguidamente se presenta la gráfica correspondiente a la lectura de temperatura a través de los sensores junto con la referencia, una vez que se han corregido, pudiéndose observar que prácticamente se superponen las tres series temporales representadas, de modo que habiendo seleccionado esas correcciones particularizadas para cada uno de los sensores, se estarán aproximando fielmente a la realidad (referencia patrón empleada para el análisis, en color amarillo, véase imagen Figura 72).

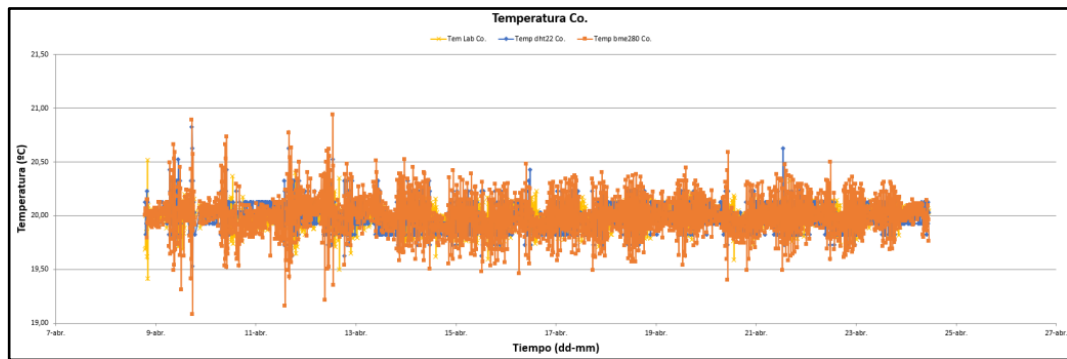


Figura 72. Gráfica comparativa de la evolución temporal de la temperatura corregida respecto de la de cada uno de los sensores con que se realiza el estudio.

Del mismo modo que se actuó con la variable anterior analizada, se graficará el error puntual,  $\varepsilon_i$ , para cada uno de los sensores, siguiendo la Ecuación [5].

$$\varepsilon_i \quad i = 1, \dots, n = [T_{\text{sensor}} - T_{\text{ref}}]_{\text{corregida}} \quad \text{Ecuación [5]}$$

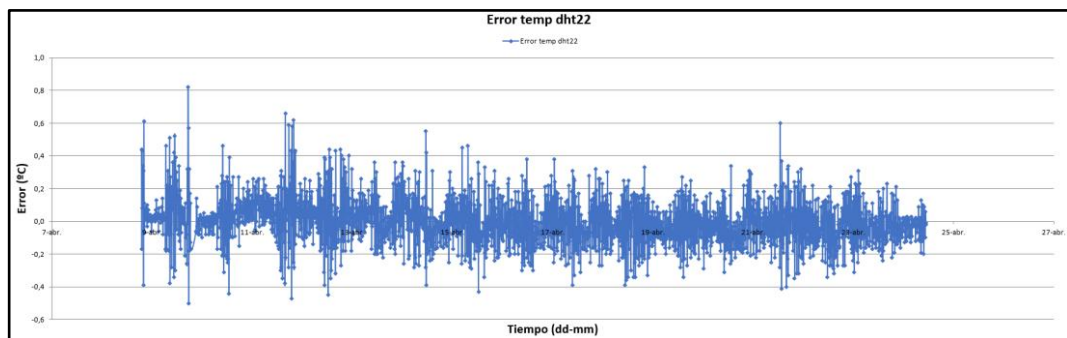


Figura 73. Error existente entre la temperatura medida por el sensor DHT22 y la referencia, ambos corregidos.

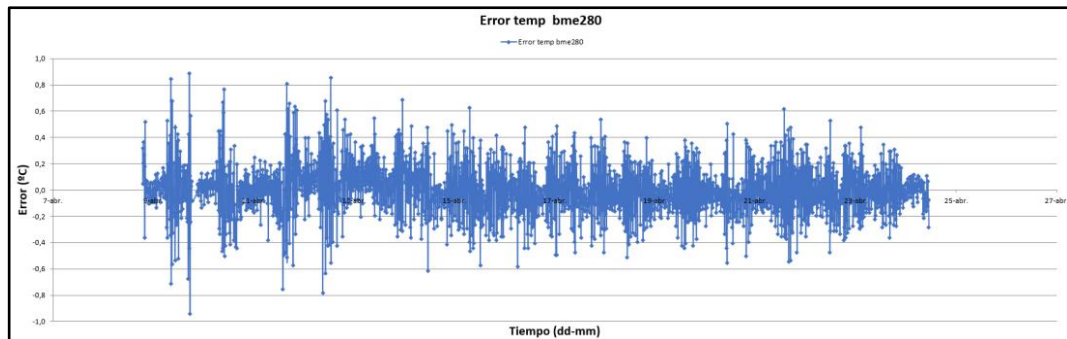


Figura 74. Error existente entre la temperatura medida por el sensor BME280 y la referencia, ambos corregidos.



Por último, se mostrará la presión corregida y su correspondiente error tras la corrección.

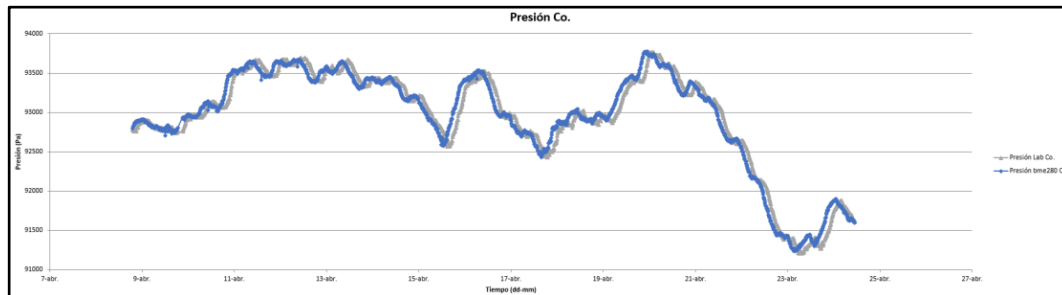


Figura 75. Gráfica comparativa de la evolución temporal de la presión corregida respecto de la del sensor BME280 corregido con que se realiza el estudio.

Con esta corrección se puede observar un posible retardo en las indicaciones de la medida de la presión del laboratorio, debido posiblemente a que la velocidad de respuesta del sensor fluctúe ante variaciones del entorno, con lo que el registro de las medidas del sensor sea recogido anteriormente al de la referencia patrón (el Laboratorio de Metrología).

La unidad básica en que se expresa la Presión en el SI es el Pascal (Pa), por lo que se aplicó el factor de conversión siguiente:

$$1 \text{ mbar} = 100 \text{ Pa}$$

Este valor de corrección de la presión de -360 Pa, puede determinarse a partir del gráfico del ajuste existente en la presión del sensor respecto a la referencia ya corregida, Figura 68, conforme a la Ecuación [6]:

$$\varepsilon_i \quad i = 1, \dots, n = [P_{\text{sensor,bme280}} - P_{\text{ref}}]_{\text{corregida}} \quad \text{Ecuación [6]}$$

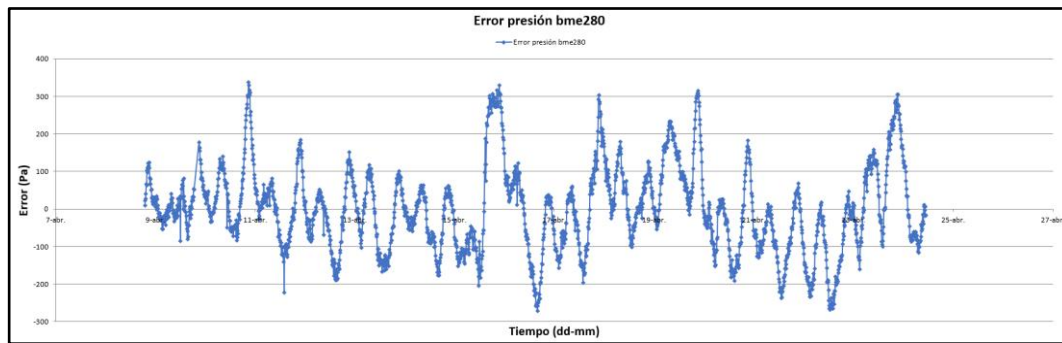


Figura 76. Error existente entre la presión medida por el sensor BME280 y la referencia, ambos corregidos.

Pese a tal observación, resultará muy útil el empleo del sensor BME280 utilizado para la medición de esta variable ya que se encuentra dentro del rango de incertidumbre que comporta el barómetro empleado como referencia.

Se muestra a continuación el valor del error medio o sistemático que se ha corregido, que será de igual módulo pero signo contrario a la corrección aplicada para tratar de anular el error sistemático, es decir, son iguales en valor absoluto corrección y error  $\epsilon_1$ , véase Ecuación [7].

$$\text{Corrección} = -\text{Error} \quad \text{Ecuación}[7]$$

	Corrección	Error
Temperatura dht22	-0,17 °C	+0,17 °C
Temperatura bme280	-0,97 °C	+0,97 °C
Humedad relativa dht22	+2,61 %	-2,61 %
Humedad relativa bme280	+5,45 %	-5,45 %
Presión bme820	-360 Pa	+360 Pa

Tabla 2. Correcciones y errores de las variables medidas por los sensores DHT22 y BME280.

Hum dht22 Co.	Hum bme280 Co.	Hum. Lab Co.	Temp Lab Co.	Temp dht22 Co.	Temp bme280 Co.	Presión Lab Co. (Pa)	Presión bme280 Co.
2,61	5,45	-1,73	-0,0325	-0,17	-0,97	-674	-360
40,6	40,8424	42,5	19,69	20,13	20,00	92786	92795
40,7	40,6568	41,9	20,00	19,83	20,04	92776	92799
40,2	40,5582	42,3	19,70	20,13	20,06	92786	92814
40,6	40,5797	41,9	19,85	19,93	20,05	92786	92826

Tabla 3. Hoja de cálculo Excel en que se calculan las correcciones de las lecturas de los sensores a partir de las correcciones establecidas.



No obstante, si se quisiese dar una repuesta más exhaustiva y explicativa numéricamente de las medidas empleadas de corrección previamente expuestas, se va a proceder a continuación a establecer una función modelo para cada variable registrada con sus correspondientes sensores, para lo cual se calculará para cerrar esta etapa, el error sistemático o medio existente tras aplicar la corrección, tanto a las indicaciones como a la referencia utilizada como patrón de medida en el análisis, junto con la desviación típica media asociada a este error, a través de las Ecuaciones [8] y [9].

$$\bar{\varepsilon} = \varepsilon_{i_{\text{medio}}} = \frac{1}{n} \sum \varepsilon_i \quad \text{Ecuación [8]}$$

$$S_{i_{\text{error}}} = \sqrt{\frac{1}{n_i} \sum |\bar{\varepsilon} - \varepsilon_i|} \quad \text{Ecuación [9]}$$

Siendo  $n_i$  el valor de números (indicaciones) que han sido recogidos por el mensurando, un número muy elevado que permite hallar resultados muy fiables y  $\bar{\varepsilon}$ , el error medio o sistemático.

	0,0	0,0	0,0	0,0	0,0
	115	0,1	0,2	1,1	1,4
<b>ERROR pres. bme Co.</b>	<b>ERROR temp. Dht. Co.</b>	<b>ERROR temp. Bme. Co.</b>	<b>ERROR hum. Dht. Co.</b>	<b>ERROR hum. Bme.Co.</b>	
9	0,4	0,3	-1,9	-1,6	
23	-0,2	0,0	-1,2	-1,2	
28	0,4	0,4	-2,1	-1,7	
40	0,1	0,2	-1,3	-1,3	

Tabla 4. Representación valores de los errores medios,  $\bar{\varepsilon}$ , (1ª fila) y desviaciones típicas,  $S_{i_{\text{error}}}$ , (2ª fila) una vez corregidas las muestras.

Se puede comprobar cómo el error sistemático se pudo corregir, no todas las veces es corregible del todo, pero sí se puede llegar a anular como ha sido el caso, mediante el método empleado. También, se puede ver como la desviación típica es distinta de cero, lo cual se debe a que el error comprende una componente, a mayores, que es aleatoria y deja expuesta la dispersión que hay en la medida ya corregida, siendo ligeramente mayor en el caso del sensor BME280 para las variables temperatura y humedad, que son las que se pueden comparar entre ambos sensores.

## 5.6 ÚLTIMA ETAPA: ESTIMACIÓN DEL MENSURANDO TRAS LA CALIBRACIÓN

Una forma de cuantificar la calidad es midiendo. Las medidas realizadas en este estudio son directas, ya que se requiere de un instrumento que de la medida directa del resultado sin solicitar ningún cálculo.



La variabilidad puede provenir de partes dispares, bien del entorno que incluye factores que se escapan del control, aquí en las gráficas anteriormente comentadas se ve reflejado por ejemplo en la variable temperatura en los períodos comprendidos entre las 08:00-20:00 hs de cada día debido a la condiciones climatológicas, que corresponden con la franja de día y son superiores a las de por la noche. También, el mensurando (T, HR o P) no es del todo homogéneo, lo cual menoscaba la exactitud (error) y precisión (dispersión) en la medida.

Son numerosas las fuentes posibles de incertidumbre en una medición, se podría añadir a las anteriores, la resolución finita del instrumento de medida o el umbral de discriminación, a lo que habría que adicionar los valores inexactos de los patrones de medida o de los materiales de referencia. En este TFG son mínimos estos últimos ya que las referencias patrón con las que se realizan las calibraciones de los equipos, disponen de garantía de calibración por ENAC.

En cuanto al error, éste tiene dos componentes, una parte sistemática y otra aleatoria. Lo que se puede corregir será la parte sistemática, que es siempre corregible, trasladando para ello la distribución, como se vio en la etapa anterior. No, obstante, este error sistemático no podrá eliminarse del todo, pero puede reducirse significativamente introduciendo un factor de corrección a fin de compensarlo, hasta poder decir que es prácticamente nulo.

En este estudio las distribuciones que se tienen son de tipo uniformes, véanse gráficas superiores correspondientes al error vs variable a medir (T, HR y P) de ambos sensores.

En cambio, la dispersión no se puede corregir debido a su naturaleza aleatoria, pues se debe a variables estocásticas o impredecibles que dan lugar a variaciones en las indicaciones repetidas del mensurando.

Una vez que se ha caracterizado el error se procede a su corrección y se asumirá que, tras aplicar la corrección, la esperanza matemática del mismo debido al efecto sistemático sea nula. Esto puede observarse en la Figura 77, donde aparece representado el Error para la variable temperatura del sensor DHT22, donde a mayores se mostrará cuál será el intervalo en que se encuentra la distribución asociada a este error del instrumento, dado por  $C_{inst}$ , que será la corrección del instrumento que se empleará dentro de la función modelo, como se describirá más adelante.

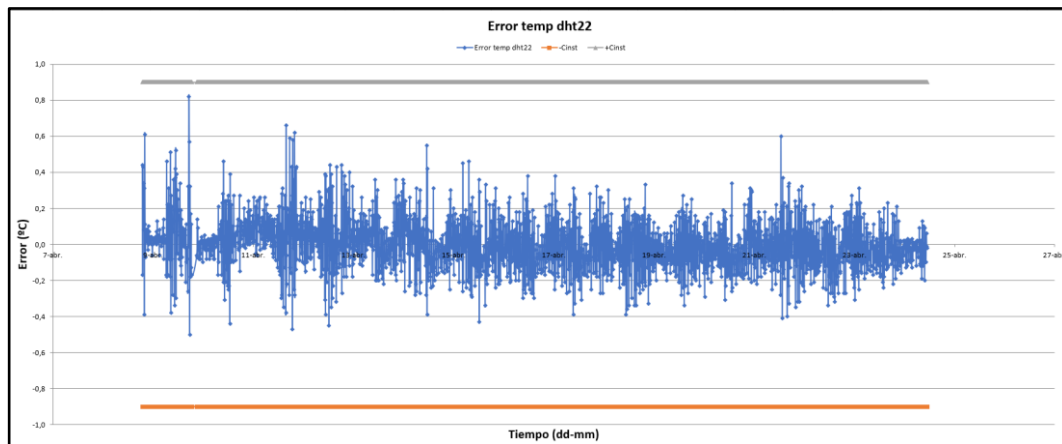


Figura 77. Error existente entre la temperatura del sensor DHT22 con respecto la referencia y sus delimitaciones.

$C_{inst}$  seguirá una distribución de tipo uniforme comprendida entre  $\pm 0,9$ .

$$C_{inst}: \text{v. a.} \rightarrow U[-0,9, 0,9]$$

El valor de la incertidumbre es el primer índice de calidad de una medida, que es tanto superior cuanto menor es aquella, siendo incertidumbre y calidad inversamente proporcionales. Esta incertidumbre manifestará la imposibilidad de saber el valor exacto del mensurando, ni siquiera corrigiendo la componente sistemática, pero sí permitirá acercarse a la medida real, expresada esta por su valor nominal acompañada de la incertidumbre.

Para establecer un proceso de evaluación de la incertidumbre, se operará conforme el esquema de la Figura 78.

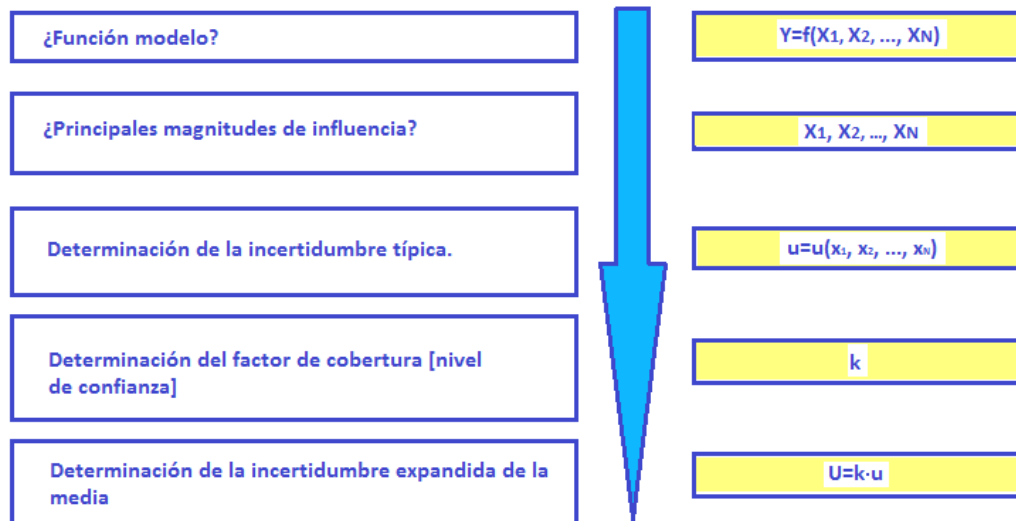


Figura 78. Esquema del proceso de evaluación o tratamiento de la incertidumbre de medida.  
Fuente: Elaboración propia.

Es por ello que para cada una de las variables medidas anteriormente, se constituirá a continuación una **función modelo**, que es una relación funcional entre las variables de "e" y de "s", siendo las variables de entrada las magnitudes de influencia.

La magnitud de salida o mensurando (lo que se desee medir), **y**, es la temperatura, la humedad relativa o en su caso la presión (lecturas del instrumento).

Y, la relación funcional que relaciona la magnitud de salida con las magnitudes de influencia será la siguiente:

$$Y = f(X_1, X_2, \dots, X_N) \quad \text{Ecuación [10]}$$

La función modelo de una medida directa tiene la siguiente expresión:

$$y = \bar{X}_i + \sum \text{Correcciones} \quad \text{Ecuación[11]}$$

El resultado de una medición es solamente una aproximación o estimación del valor del mensurando y únicamente se encontrará completo cuando vaya la medición asociada a una declaración de la incertidumbre.

Inicialmente, se propondrá la siguiente expresión para el cálculo de la estimación de la magnitud de salida,  $\hat{y}$ :

$$\hat{y} = E(\bar{X}_i + \sum C_i) = \frac{1}{n} \sum_{i=1}^n X_i + E(C_1) + E(C_2) + \dots \quad \text{Ecuación [12]}$$

siendo el operador lineal, E, la esperanza matemática.

Las componentes de incertidumbre se estructuran en dos categorías, en función del método de evaluación, véase Figura 79.

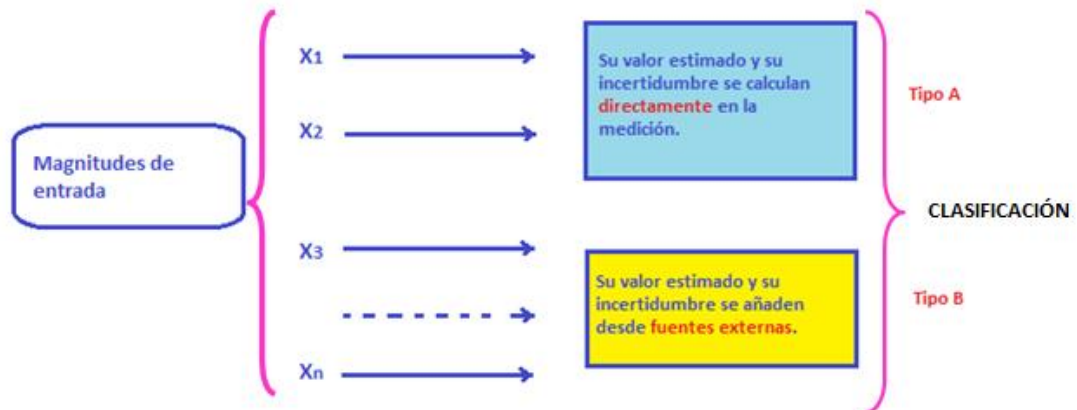


Figura 79. Categorías en que se agrupa la incertidumbre, según el método de evaluación. [11]

- **Tipo A:** Estudios estadísticos de observaciones de medición.
- **Tipo B:** Se obtiene de forma externa (el resto).

Las magnitudes de **entrada** ( $X_1, X_2, \dots, X_N$ ), de las que dependerá la magnitud de salida irán acompañadas con sus correcciones y factores de corrección de los efectos que contribuyan a la parte sistemática del error.

Se calculará la desviación típica estimada correspondiente a la estimación de salida, o lo que es lo mismo la **incertidumbre típica combinada**,  $u_c(y)$ , que vendrá fijada por el conjunto de desviaciones típicas estimadas, asociadas a cada una de las estimaciones de entrada  $x_i$ , llamadas también **incertidumbres típicas**,  $u(x_i)$ .

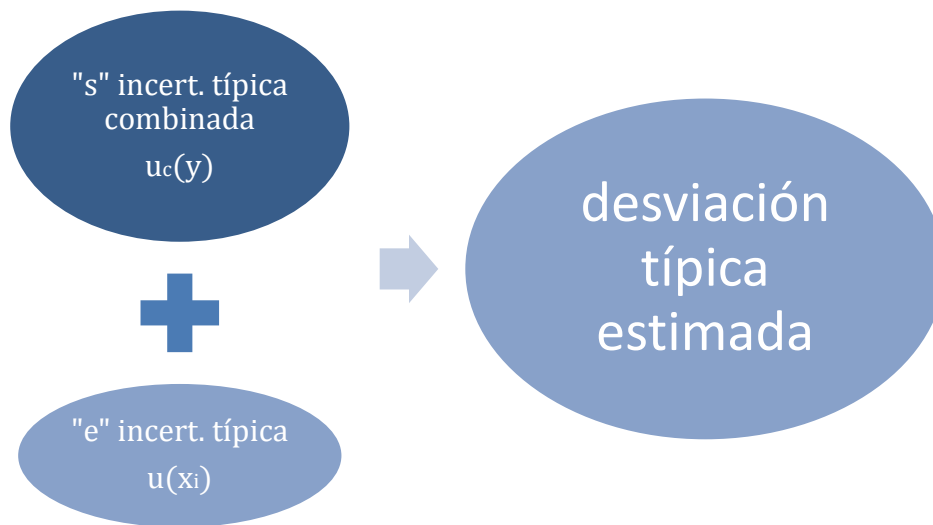


Figura 80. Esquema desviación típica estimada. Fuente: Elaboración propia.

Para el **análisis de la incertidumbre típica** asociada a una componente de **tipo A**, se procederá del siguiente modo:

Se emplea como **estimación** de la esperanza matemática  $\mu_q$  de una magnitud  $q$  aleatoria, de la que se han tomado  $n$  observaciones todas ellas independientes  $q_k$ , en las mismas condiciones de medida, **la media aritmética  $\bar{q}$** , véase Ecuación [13]

$$\bar{q} = \frac{1}{n} \sum_{j=1}^n q_j \quad \text{Ecuación [13]}$$

La **varianza experimental** de las  $n$  observaciones que se utiliza para estimar la varianza  $\sigma^2$  de la distribución de probabilidad de la magnitud  $q$  es:

$$s^2(q) = \frac{1}{n-1} \sum_{j=1}^n (q_j - \bar{q})^2 \quad \text{Ecuación [14]}$$

De la anterior ecuación, se extrae la **desviación típica experimental**,  $s(q_k)$ , que como se comentó anteriormente reflejará la dispersión de los valores de las  $n$  observaciones alrededor de la media,  $\bar{q}$ . Y, es a partir de esa misma ecuación de la que se dedujo que la mejor estimación de la **varianza de la media** vendrá dada por la Ecuación [15].



$$s^2(\bar{q}) = \frac{s^2(q_k)}{n} \quad \text{Ecuación [15]}$$

De esta manera, si se tiene una magnitud de entrada  $X_i$  a partir de  $n$  observaciones independientes y medidas en las mismas condiciones, la incertidumbre típica de tipo A,  $u(x_i)$  de su estimación  $x_i$  es la desviación típica de la media, véase Ecuación [16].

$$u(x_i) = s(\bar{q}) = \frac{s(q)}{\sqrt{n}} \quad \text{Ecuación [16]}$$

Una vez se han calculado las distintas contribuciones de las magnitudes de entrada a la incertidumbre, será necesario combinarlas, para ello si las magnitudes de entrada no se encuentran correlacionadas y la función modelo que se tiene es lineal, la incertidumbre típica combinada se calculará según la Ecuación [17].

$$u^2(y) = \sum_{i=1}^N u_i^2(y) \quad \text{Ecuación [17]}$$

Siendo  $u_i(y)$  la contribución de la incertidumbre típica correspondiente a la estimación de salida,  $y$ , determinada ésta última de la incertidumbre típica correspondiente a la estimación de entrada,  $x_i$ .

$$u_i(y) = c_i \cdot u(x_i) \quad \text{Ecuación [18]}$$

En esta última expresión,  $c_i$ , es el coeficiente de sensibilidad correspondiente a la estimación de entrada,  $x_i$  y se encarga de explicar cómo las variaciones en la estimación de entrada,  $x_i$  afectan a la estimación de salida, siendo el valor del mismo positivo o negativo, véase Ecuación [19].

$$c_i = \frac{\partial f}{\partial x_i} = \left. \frac{\partial f}{\partial X_i} \right|_{X_1 = x_1 \dots X_N = x_N} \quad \text{Ecuación [19]}$$

Para determinar por último el valor de la **incertidumbre expandida**,  $U$ , una vez calculada la incertidumbre típica combinada  $u(y)$  asociada a cada magnitud de salida se multiplicará a esta por un **factor de cobertura** ( $k$ ).

$$U = k \cdot u(y)$$

La muestra que se viene estudiando abarca un número muy elevado de indicaciones y, además, no se encuentran en una cámara aislada ni exentas



de las variaciones del entorno, por lo que para dar un resultado válido de incertidumbre expandida a cada una de las variables leídas por sus respectivos sensores se procederá a coger las indicaciones medidas en una hora con un período de 10 minutos, es decir n será 6 observaciones, de manera que aunque se disponga de 3 componentes o magnitudes de influencia del tipo B con infinitos gdl que explique perfectamente lo que ocurre y con ello se cumpla la hipótesis de normalidad, no se podrá afirmar que el factor de corrección k sea 2.

Esto es debido a que el criterio de fiabilidad que viene dado por los grados efectivos de libertad,  $\nu_{\text{eff}}$ , no se cumple ya que las contribuciones a la incertidumbre de tipo A no tienen un número  $n \geq 10$ . Conforme a esto, al no cumplirse una de las dos hipótesis, que conllevaría a aceptar como  $k=2$  y por tanto tener  $\nu_{\text{eff}} = \infty$ , no se puede suponer de entrada que los grados efectivos de libertad sean infinitos.

El número de los grados de libertad de la incertidumbre de tipo A correspondiente a la magnitud de influencia,  $X_i$ , que será la referida a los valores obtenidos directamente de la medición y que proporciona el sensor i,  $\nu_i$ , es  $n-1$ .

De acuerdo con ello, para la deducción del factor de cobertura, se hará uso de la fórmula de Welch-Satterthwaite:

$$\nu_{\text{ef}} = \frac{u^4(y)}{\sum_{i=1}^N \frac{u_i^4(y)}{\nu_i}} \quad \text{Ecuación [20]}$$

Una vez calculados los grados efectivos de libertad, si resultase ser un número superior a 50, se muestra en la Figura 81 cómo apoyándose en la tabla t-Student con una cobertura del 95,45 % se entraría con este dato y el factor de cobertura k sería el correspondiente, en este ejemplo, k sería 2. Es más adelante, para cada uno de los valores donde se verá aplicada la misma, pues ya se vio que al no cumplirse una de las dos hipótesis, en este caso la de fiabilidad, por ser  $n < 10$  no se podría afirmar directamente que k fuese 2 a priori.

$\nu_{\text{ef}}$	1	2	3	4	5	6	7	8	10	20	50	$\infty$
k	13.97	4.53	3.31	2.87	2.65	2.52	2.43	2.37	2.28	2.13	2.05	2.00

Figura 81. Tabla t-Student 95,45 % con los factores de seguridad o cobertura, k, correspondientes a sus grados de libertad  $\nu_{\text{ef}}$ .



Para las de **tipo B** se obtiene de forma externa, disponiendo del equipo y procediéndose a analizar el sesgo en base a su caracterización por medio de una referencia.

En la estimación de una componente de incertidumbre de esta clase se debe evitar la subestimación, cubriendo los límites superior e inferior toda la probabilidad, de modo que se pueda concluir que los  $g_{dl}$  sean infinitos,  $u_{eff} \rightarrow \infty g_{dl}$ .

La muestra a analizar será tomada en unas condiciones de medida lo más estables posibles y constará de  $m=6$  observaciones, medidas a las 11 horas del sábado 13 de Abril, siendo los datos que la componen los que aparecen en la Tabla 5.

DHT22			Lab Metr			BME280		
Hora hh:mm	Humedad %	Temperatura °C	Temperatura °C	Humedad %	Presión Pa	Temperatura °C	Humedad %	Presión Pa
11:02	30,0	20,3	19,99	34,5	94350	21,20	29,42	93901
11:12	30,3	20,0	20,01	34,5	94330	20,93	29,88	93909
11:22	30,4	20,0	20,06	34,4	94310	20,89	29,86	93890
11:32	30,0	20,4	19,96	34,6	94320	21,07	29,63	93880
11:42	30,2	20,1	20,02	34,6	94330	20,98	29,76	93866
11:52	30,2	20,3	20,00	34,7	94330	21,18	29,64	93869

Tabla 5. Muestra de datos con que se realizan las funciones modelo.

En primer lugar como se dijo se presentará la función modelo de las medidas directas, siendo la primera variable a estudiar la temperatura, y a continuación se procederá a calcular las estimaciones citadas anteriormente, junto con el valor de incertidumbre típica combinada, a lo que se añadirá el error de cobertura para determinar la incertidumbre expandida y poder proporcionar un resultado claro y conciso de la variable estudiada.



- FUNCIÓN MODELO PARA LA TEMPERATURA DEL SENSOR DHT22:

$$L_{T^{\circ}\text{dht22}} = \bar{X}_i + C_{\text{inst}} + C_E + C_{\text{deriva}}$$

$\bar{X}_i$  = promedio de las medidas (lecturas de temperatura).

$C_{\text{inst}}$  = corrección del instrumento de medida.

$C_E$  = hace alusión a la precisión del instrumento de medida significando la corrección por redondeo o de escala. (Ésta debe incluirse siempre).

Una vez se tiene la función modelo, se debe cuantificar, calculándose la estimación de la lectura, para lo cual se aplica la esperanza matemática a la misma:

$$\begin{aligned}\hat{L}_{T^{\circ}\text{dht22}} &= E(\bar{L}_i + C_{\text{inst}} + C_E + C_{\text{deriva}}) \\ &= E(\bar{L}_i) + E(C_{\text{inst}}) + E(C_E) + E(C_{\text{deriva}})\end{aligned}$$

$$E(\bar{L}_i) = \frac{1}{m} \sum L_i = \frac{20,3 + 20,0 + 20,0 + 20,4 + 20,1 + 20,3}{6} = 20,183 \text{ }^{\circ}\text{C}$$

La corrección por escala es una variable aleatoria que presenta una distribución de tipo uniforme o rectangular en el intervalo  $[-E/2, +E/2]$  °C con esperanza matemática nula.

$$E = 0,1 \text{ }^{\circ}\text{C}; \text{ v. a. } \rightarrow U[-0,05, 0,05] \begin{cases} E(C_E) = \frac{-0,05 + 0,05}{2} = 0 \text{ }^{\circ}\text{C} \\ S(C_E) = \frac{0,05 - (-0,05)}{\sqrt{12}} = \frac{0,1}{\sqrt{12}} \text{ }^{\circ}\text{C} \end{cases}$$

$$C_{\text{inst}}: \text{ v. a. } \rightarrow U[-0,9, 0,9] \begin{cases} E(C_{\text{inst}}) = \frac{-0,9 + 0,9}{2} = 0 \text{ }^{\circ}\text{C} \\ S(C_{\text{inst}}) = \frac{0,9 - (-0,9)}{\sqrt{12}} = \frac{1,8}{\sqrt{12}} \text{ }^{\circ}\text{C} \end{cases}$$

$$\begin{aligned}E(C_{\text{inst}}) &= C_{\text{inst}}: \text{ v. a. } \rightarrow U[-C_{\text{máxadmisible}}, +C_{\text{máxadmisible}}], \text{ siendo } E(C_{\text{inst}}) \\ &= 0 \text{ ya que corresponde con el centro del intervalo.}\end{aligned}$$

Sigue una distribución de tipo uniforme o rectangular, también, puesto que de las gráficas únicamente se dispone de los valores de los límites superior e inferior para caracterizarlo, siendo esta distribución por ello la que resulta más idónea.

En cuanto a la corrección por deriva, se dará un valor razonable estimativo, aunque sería necesario el histórico de las medidas para poder corroborar esta cifra.

$E(C_{\text{deriva}}) = C_{\text{deriva}}$ : v. a.  $\rightarrow U[-\text{der}_{\text{máx.adm.}}, +\text{der}_{\text{máx.adm.}}]$ , siendo  $E(C_{\text{deriva}}) = 0$

$\text{der}_{\text{máx.adm.}} = \pm 0,05 \text{ } ^\circ\text{C}$ , siendo inferior al valor de la corrección.

Esta última componente, también sigue una distribución de tipo uniforme o rectangular, a continuación aparece graficada una **distribución** de tipo **uniforme** en el intervalo  $[a,b]$  con sus parámetros principales.

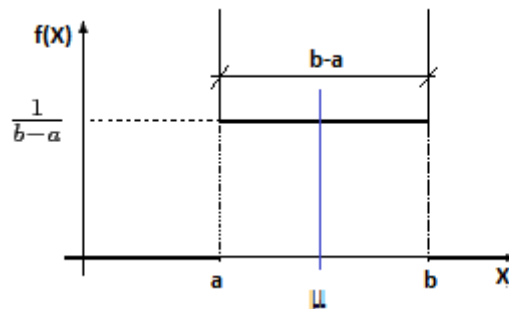


Figura 82. Función de densidad de distribución Uniforme continua.

En la que:

$$f(x) = \begin{cases} \frac{1}{b-a} & \text{si } x \in (a, b) \\ 0 & \text{si } x \notin (a, b) \end{cases} \quad \mu = \frac{a+b}{2}$$

$$\sigma^2 = \int_{-\infty}^{+\infty} (x - \mu)^2 f(x) dx = \frac{(b-a)^2}{12} \rightarrow \sigma = \frac{(b-a)}{\sqrt{12}}$$

En este punto seremos capaces de dar un valor de la estimación, que será la suma de las cuatro contribuciones que tenemos en la función modelo, siendo:

$$\hat{L}_{T^a_{\text{dht22}}} = 20,183 \text{ } ^\circ\text{C} \pm U(\hat{L}_{T^a_{\text{dht22}}})$$

Toda medida como ya se dijo va asociada a una incertidumbre de la estimación, que se procede a calcular.

La incertidumbre típica combinada se calcula del siguiente modo:



$$\begin{aligned}
 u(\hat{L}_{T^a dht22}) &= \text{Según la LPV (Ley de Propagación de varianzas)} \\
 &= \sqrt{C_1^2 \cdot u^2(\bar{L}_i) + C_2^2 \cdot u^2(C_{inst}) + C_3^2 \cdot u^2(C_E) + C_4^2 \cdot u^2(C_{deriva})} \\
 &= \sqrt{u^2(\bar{L}_i) + u^2(C_{inst}) + u^2(C_E) + u^2(C_{deriva})}
 \end{aligned}$$

Donde los coeficientes de sensibilidad,  $C_i$  son:

$$C_1 = \frac{\partial \hat{L}_{T^a dht22}}{\partial \bar{L}_i} = 1$$

$$C_2 = \frac{\partial \hat{L}_{T^a dht22}}{\partial C_{inst}} = 1$$

$$C_3 = \frac{\partial \hat{L}_{T^a dht22}}{\partial C_E} = 1$$

$$C_4 = \frac{\partial \hat{L}_{T^a dht22}}{\partial C_{deriva}} = 1$$

De donde  $u_i = 1 \cdot u(x_i)$  pues los coeficientes de sensibilidad son unitarios.

Se establece una tabla para el cálculo de las incertidumbres, véase Tabla 6.



Magnitud de entrada ( $X_i$ )	Distribución de probabilidad	Incertidumbre típica $u(x_i)$	Contribución (incertidumbre típica) $u_i(y) = c_i \cdot u(x_i)$	Tipo	gdl
$\bar{L}_i$	Normal	$\frac{S_i}{\sqrt{m}} = \frac{0,1722}{\sqrt{6}} = 0,0703$	<b>0,0703</b>	A	$(m-1) = 6-1 = 5$
$C_{inst}$	Uniforme	$\frac{2C_{m\acute{a}x adm}}{\sqrt{12}} = \frac{C_{m\acute{a}x adm}}{\sqrt{3}} = \frac{1,8}{\sqrt{12}} \text{ °C} = 0,5196$	<b>0,5196</b>	B	$\infty$
$C_E$	Uniforme	$\frac{E}{\sqrt{12}} = \frac{0,1}{\sqrt{12}} \text{ °C} = 0,0289$	<b>0,0289</b>	B	$\infty$
$C_{deriva}$	Uniforme	$\frac{2C_{der m\acute{a}x.adm.}}{\sqrt{12}} = \frac{C_{der m\acute{a}x.adm.}}{\sqrt{3}} = \frac{0,1}{\sqrt{12}} \text{ °C} = 0,0289$	<b>0,0289</b>	B	$\infty$
Incertidumbre típica combinada		$u(\hat{L}_{T^a dht22}) = \sqrt{u^2(\bar{L}_i) + u^2(C_{inst}) + u^2(C_E) + u^2(C_{deriva})} = 0,5259 \text{ °C}$			
$v_{ef}$		<b>15.659</b>			

Tabla 6. Determinación de la incertidumbre típica combinada para la variable temperatura medida por el sensor DHT22 y sus gdl.

La última columna representa los grados de libertad que indicarán cómo de fiable es la muestra.

Una vez calculada la incertidumbre típica combinada, se procede a hallar el factor de cobertura, como no se cumple la hipótesis de normalidad, no se puede concluir que  $k$  sea 2, es por ello que se utiliza la Ecuación [20] para hallar los grados efectivos de libertad:

$$v_{ef} = \frac{u^4(y)}{\sum_{i=1}^N \frac{u_i^4(y)}{v_i}} = \frac{u(\hat{L}_{T^a dht22})^4}{\frac{u\bar{L}_i^4}{m-1} + \frac{u_{C_{inst}}^4}{\infty} + \frac{u_{C_E}^4}{\infty} + \frac{u_{C_{deriva}}^4}{\infty}} \rightarrow$$



$$v_{ef} = (m - 1) \left( \frac{u(\hat{L}_{T^a_{dht22}})}{u\bar{L}_i} \right)^4 = 5 \cdot \left( \frac{0,5259 \text{ °C}}{0,0703 \text{ °C}} \right)^4 = 15.658,9049 = \mathbf{15.659}$$

Se irá a la tabla de la Figura 81 y entrando con  $v_{ef} = \infty$  se tendrá un factor de cobertura,  $k=2$ .

La incertidumbre expandida será:  $U=k \cdot u = 2 \cdot 0,5259 \text{ °C} = 1,0518 \text{ °C}$

Para la expresión final del resultado se tiene que redondear esta cifra a 2 cifras significativas, para ajustar la incertidumbre expandida, U, evitando una subestimación superior al 5 %, no estando permitido superar este valor.

Es por ello que el resultado de la incertidumbre expandida, U, será de 1,1 °C.

Por último, antes de expresar la medida, se debe ajustar de forma que tenga la misma resolución que la incertidumbre ajustada.

Entonces el resultado de la medida sería de:

$$\hat{L}_{T^a_{dht22}} = 20,2 \pm 1,1 \text{ °C}$$

Para reducir la incertidumbre tendría que haber seleccionado más medidas.





- FUNCIÓN MODELO PARA LA TEMPERATURA DEL SENSOR BME280:

Si se realizase la misma comprobación pero cambiando el sensor, en este caso se utilizará el BME280 para medir la variable temperatura, los resultados obtenidos se obtendrán a continuación.

$$L_{T^a_{bme280}} = \bar{X}_i + C_{inst} + C_E + C_{deriva}$$

$$E(\bar{L}_i) = \frac{1}{m} \sum L_i = \frac{21,20 + 20,93 + 20,89 + 21,07 + 20,98 + 21,18}{6}$$

$$= 21,0417 \text{ } ^\circ\text{C}$$

La corrección por escala es una variable aleatoria que presenta una distribución de tipo uniforme o rectangular en el intervalo  $[-E/2, +E/2]$   $^\circ\text{C}$  con esperanza matemática nula.

$$E = 0,01 \text{ } ^\circ\text{C}: v. a. \rightarrow U[-0,005, 0,005] \left\{ \begin{array}{l} E(C_E) = \frac{-0,005 + 0,005}{2} = 0 \text{ } ^\circ\text{C} \\ S(C_E) = \frac{0,005 - (-0,005)}{\sqrt{12}} = \frac{0,01}{\sqrt{12}} \text{ } ^\circ\text{C} \end{array} \right.$$

$$C_{inst}: v. a. \rightarrow U[-1, 1] \left\{ \begin{array}{l} E(C_{inst}) = \frac{-1 + 1}{2} = 0 \text{ } ^\circ\text{C} \\ S(C_{inst}) = \frac{1 - (-1)}{\sqrt{12}} = \frac{2}{\sqrt{12}} \text{ } ^\circ\text{C} \end{array} \right.$$

$$E(C_{inst}) = C_{inst}: v. a. \rightarrow U[-C_{m\acute{a}x.admisible}, +C_{m\acute{a}x.admisible}], \text{ siendo } E(C_{inst}) = 0 \text{ ya que corresponde con el centro del intervalo.}$$

En cuanto a la corrección por deriva, se dará un valor razonable estimativo, aunque sería necesario el histórico de las medidas para poder corroborar esta cifra.

$$E(C_{deriva}) = C_{deriva}: v. a. \rightarrow U[-der_{m\acute{a}x.adm.}, +der_{m\acute{a}x.adm.}], \text{ siendo } E(C_{deriva}) = 0$$

$der_{m\acute{a}x.adm.} = \pm 0,05 \text{ } ^\circ\text{C}$ , siendo inferior al valor de la corrección.

En este punto seremos capaces de dar un valor de la estimación, que será la suma de las cuatro contribuciones que tenemos en la función modelo, siendo:

$$\hat{L}_{T^a_{bme280}} = 21,0417 \text{ } ^\circ\text{C} \pm U(\hat{L}_{T^a_{bme280}})$$



Los coeficientes de sensibilidad, son también unitarios, pudiéndose verificar a través de la función modelo. Para presentar los resultados de una forma clarificada y ordenada, se incorporarán a la Tabla 7.

Magnitud de entrada ( $X_i$ )	Distribución de probabilidad	Incertidumbre típica $u(x_i)$	Contribución (incertidumbre típica) $u_i(y) = c_i \cdot u(x_i)$	Tipo	Gdl
$\bar{L}_i$	Normal	$\frac{S_i}{\sqrt{m}} = \frac{0,1298}{\sqrt{6}} = 0,0530$	<b>0,0530</b>	A	(m-1)= 6-1=5
$C_{inst}$	Uniforme	$\frac{2C_{m\acute{a}x adm}}{\sqrt{12}} = \frac{C_{m\acute{a}x adm}}{\sqrt{3}} = \frac{2}{\sqrt{12}} \text{ } ^\circ\text{C} = 0,5774$	<b>0,5774</b>	B	$\infty$
$C_E$	Uniforme	$\frac{E}{\sqrt{12}} = \frac{0,01}{\sqrt{12}} \text{ } ^\circ\text{C} = 0,0029$	<b>0,0029</b>	B	$\infty$
$C_{deriva}$	Uniforme	$\frac{2C_{der.m\acute{a}x.adm.}}{\sqrt{12}} = \frac{C_{der.m\acute{a}x.adm.}}{\sqrt{3}} = \frac{0,1}{\sqrt{12}} \text{ } ^\circ\text{C} = 0,0289$	<b>0,0289</b>	B	$\infty$
<b>Incertidumbre típica combinada</b>		$u(\hat{L}_{T^{\circ}bme280}) = \sqrt{u^2(\bar{L}_i) + u^2(C_{inst}) + u^2(C_E) + u^2(C_{deriva})} = 0,5806 \text{ } ^\circ\text{C}$			
$u_{ef}$		<b>88.758</b>			

Tabla 7. Determinación de la incertidumbre típica combinada para la variable temperatura medida por el sensor BME280 y sus gdl.

Una vez calculada la incertidumbre típica combinada, se procede a hallar el factor de cobertura, como no se cumple la hipótesis de normalidad, no se puede concluir que k sea 2, es por ello que se utiliza la Ecuación [20] para hallar los grados efectivos de libertad:



$$v_{ef} = \frac{u^4(y)}{\sum_{i=1}^N \frac{u_i^4(y)}{v_i}} = \frac{u(\hat{L}_{T^{a}bme280})^4}{\frac{u\bar{L}_i^4}{m-1} + \frac{u_{C_{inst}}^4}{\infty} + \frac{u_{C_E}^4}{\infty} + \frac{u_{C_{deriva}}^4}{\infty}} \rightarrow$$

$$v_{ef} = (m - 1) \left( \frac{u(\hat{L}_{T^{a}bme280})}{u\bar{L}_i} \right)^4 = 5 \cdot \left( \frac{0,5806 \text{ } ^\circ\text{C}}{0,0503 \text{ } ^\circ\text{C}} \right)^4 = 88.757,73 = \mathbf{88.758}$$

Se irá a la tabla de la Figura 81 y entrando con  $v_{ef} = \infty$  se tendrá un factor de cobertura,  $k=2$ .

La incertidumbre expandida será:  $U = k \cdot u = 2 \cdot 0,5806 \text{ } ^\circ\text{C} = 1,1612 \text{ } ^\circ\text{C}$

Para la expresión final del resultado se tiene que redondear esta cifra a 2 cifras significativas, para ajustar la incertidumbre expandida, U, evitando una subestimación superior al 5 %, no estando permitido superar este valor.

Es por ello que el resultado de la incertidumbre expandida, U, será de 1,2 °C.

Por último, antes de expresar la medida, se debe ajustar de forma que tenga la misma resolución que la incertidumbre ajustada.

Entonces el resultado de la medida sería de:

$$\hat{L}_{T^{a}bme280} = 21,0 \pm 1,2 \text{ } ^\circ\text{C}$$

El sensor BME280, además de medir temperatura y humedad relativa, permitirá recoger los datos de la presión.

- FUNCIÓN MODELO PARA LA PRESIÓN DEL SENSOR BME280:

$$P_{\text{Presionbme280}} = \bar{P}_i + C_{\text{inst}} + C_E + C_{\text{deriva}}$$

$$\begin{aligned}\hat{P}_{\text{Presionbme280}} &= E(P_i + C_{\text{inst}} + C_E + C_{\text{deriva}}) \\ &= E(\bar{P}_i) + E(C_{\text{inst}}) + E(C_E) + E(C_{\text{deriva}})\end{aligned}$$

$$\begin{aligned}E(\bar{P}_i) &= \frac{1}{m} \sum P_i = \frac{93901 + 93909 + 93890 + 93880 + 93866 + 93869}{6} \\ &= \mathbf{93885,833 \text{ Pa}}\end{aligned}$$

$$E = 1 \text{ Pa} \quad (C_E) : \text{v. a.} \rightarrow U[-0,5, 0,5] \quad \left\{ \begin{array}{l} E(C_E) = \frac{-0,5 + 0,5}{2} = \mathbf{0 \text{ Pa}} \\ S(C_E) = \frac{0,5 - (-0,5)}{\sqrt{12}} = \frac{1}{\sqrt{12}} \text{ Pa} \end{array} \right.$$

$$C_{\text{inst}} : \text{v. a.} \rightarrow U[-340, 340] \quad \left\{ \begin{array}{l} E(C_{\text{inst}}) = \frac{-340 + 340}{2} = \mathbf{0 \text{ Pa}} \\ S(C_{\text{inst}}) = \frac{340 - (-340)}{\sqrt{12}} = \frac{680}{\sqrt{12}} \text{ Pa} \end{array} \right.$$

$$C_{\text{deriva}} : \text{v. a.} \rightarrow U[-\text{der}_{\text{mín}}, +\text{der}_{\text{máx}}]$$

$$\text{der}_{\text{máx.adm.}} = \pm 30 \text{ Pa}$$

Esta última componente, también sigue una distribución de tipo uniforme o rectangular.

En este punto seremos capaces de dar un valor de la estimación, que será la suma de las cuatro contribuciones que tenemos en la función modelo, siendo:

$$\hat{P}_{\text{Presionbme280}} = 93885,8833 \text{ Pa} \pm U(\hat{P}_{\text{Presionbme280}})$$

Los coeficientes de sensibilidad, son también unitarios, pudiéndose verificar a través de la función modelo. Para presentar los resultados de una forma clarificada y ordenada, se incorporarán a la Tabla 8.



Magnitud de entrada ( $X_i$ )	Distribución de probabilidad	Incertidumbre típica $u(x_i)$	Contribución (incertidumbre típica) $u_i(y) = c_i \cdot u(x_i)$	Tipo	gdl
$\bar{P}_i$	Normal	$\frac{S_i}{\sqrt{m}} = \frac{17}{\sqrt{6}} = 7,0589$	<b>7,0589</b>	A	$(m-1) = 6-1=5$
$C_{inst}$	Uniforme	$\frac{2C_{m\acute{a}x adm}}{\sqrt{12}} = \frac{C_{m\acute{a}x adm}}{\sqrt{3}} = \frac{680}{\sqrt{12}} \text{ } ^\circ\text{C} = 196,2991$	<b>196,2991</b>	B	$\infty$
$C_E$	Uniforme	$\frac{E}{\sqrt{12}} = \frac{1}{\sqrt{12}} \text{ } ^\circ\text{C} = 0,2889$	<b>0,2889</b>	B	$\infty$
$C_{deriva}$	Uniforme	$\frac{2C_{der.m\acute{a}x.adm.}}{\sqrt{12}} = \frac{C_{der.m\acute{a}x.adm.}}{\sqrt{3}} = \frac{60}{\sqrt{12}} \text{ Pa} = 17,3205$	<b>17,3205</b>	B	$\infty$
Incertidumbre típica combinada		$u(\hat{P}_{Presionbme280}) = \sqrt{u^2(\bar{P}_i) + u^2(C_{inst}) + u^2(C_E) + u^2(C_{deriva})} = 197,1884 \text{ Pa}$			
$u_{ef}$		<b>3.044.720</b>			

Tabla 8. Determinación de la incertidumbre típica combinada para la variable presión medida por el sensor BME280 y sus gdl.

Una vez calculada la incertidumbre típica combinada, se procede a hallar el factor de cobertura, como no se cumple la hipótesis de normalidad, no se puede concluir que k sea 2, es por ello que se utiliza la Ecuación [20] para hallar los grados efectivos de libertad:

$$v_{ef} = \frac{u^4(y)}{\sum_{i=1}^N \frac{u_i^4(y)}{u_i}} = \frac{u(\hat{P}_{Presionbme280})^4}{\frac{u\bar{P}_i^4}{m-1} + \frac{u_{C_{inst}}^4}{\infty} + \frac{u_{C_E}^4}{\infty} + \frac{u_{C_{deriva}}^4}{\infty}} \rightarrow$$



$$v_{ef} = (m - 1) \left( \frac{u(\hat{P}_{Presionbme280})}{u\hat{P}_i} \right)^4 = 5 \cdot \left( \frac{197,1884 \text{ Pa}}{7,0589 \text{ Pa}} \right)^4 = 3.044.719,851 = \mathbf{3.044.720}$$

Se irá a la tabla de la Figura 81 y entrando con  $v_{ef} = \infty$  se tendrá un factor de cobertura,  $k=2$ .

La incertidumbre expandida será:  $U = k \cdot u = 2 \cdot 197,1884 \text{ Pa} = \mathbf{394,3768 \text{ Pa}}$

Para la expresión final se redondeará este valor a 2 cifras significativas para evitar una subestimación por encima del 5 %, de modo que el valor de la incertidumbre expandida será de: **390 Pa**.

Por último, antes de expresar la medida, se debe ajustar de forma que tenga la misma resolución que la incertidumbre ajustada.

Entonces el resultado de la medida sería de:

$$\hat{P}_{Presionbme280} = 93886 \pm 390 \text{ Pa}$$

La última variable que quedaría por analizar, sería la humedad relativa. Se comenzará analizándola respecto al sensor DHT22.

- FUNCIÓN MODELO PARA LA HUMEDAD RELATIVA DEL SENSOR DHT22:

$$H_{\text{Humdht22}} = \bar{H}_i + C_{\text{inst}} + C_E + C_{\text{deriva}}$$

$$\begin{aligned}\hat{H}_{\text{Humdht22}} &= E(H_i + C_{\text{inst}} + C_E + C_{\text{deriva}}) \\ &= E(\bar{H}_i) + E(C_{\text{inst}}) + E(C_E) + E(C_{\text{deriva}})\end{aligned}$$

$$E(\bar{H}_i) = \frac{1}{m} \sum H_i = \frac{30,0 + 30,3 + 30,4 + 30,0 + 30,2 + 30,2}{6} = \mathbf{30,1833 \%}$$

$$E = 0,1 \% : \text{v. a.} \rightarrow U[-0,05, 0,05] \left\{ \begin{array}{l} E(C_E) = \frac{-0,05 + 0,05}{2} = \mathbf{0 \%} \\ S(C_E) = \frac{0,05 - (-0,05)}{\sqrt{12}} = \frac{0,1}{\sqrt{12}} \%$$

$$C_{\text{inst}} : \text{v. a.} \rightarrow U[-5,5] \left\{ \begin{array}{l} E(C_{\text{inst}}) = \frac{-5 + 5}{2} = \mathbf{0 \%} \\ S(C_{\text{inst}}) = \frac{5 - (-5)}{\sqrt{12}} = \frac{10}{\sqrt{12}} \%$$

$$C_{\text{deriva}} : \text{v. a.} \rightarrow U[-\text{der}_{\text{mín}}, +\text{der}_{\text{máx}}]$$

$$\text{der}_{\text{máx.adm.}} = \pm 0,8 \%$$

Esta última componente, también sigue una distribución de tipo uniforme o rectangular.

En este punto seremos capaces de dar un valor de la estimación, que será la suma de las cuatro contribuciones que tenemos en la función modelo, siendo:

$$\hat{H}_{\text{Humdht22}} = 30,1833 \% \pm U(H_{\text{Humdht22}})$$

Los coeficientes de sensibilidad, son también unitarios, pudiéndose verificar a través de la función modelo. Para presentar los resultados de una forma clarificada y ordenada, se incorporarán a la Tabla 9.



Magnitud de entrada ( $X_i$ )	Distribución de probabilidad	Incertidumbre típica $u(x_i)$	Contribución (incertidumbre típica) $u_i(y) = c_i \cdot u(x_i)$	Tipo	gdl
$\bar{P}_i$	Normal	$\frac{S_i}{\sqrt{m}} = \frac{0,1602}{\sqrt{6}} =$ <b>0,0654</b>	<b>0,0654</b>	A	(m-1)= 6-1=5
$C_{inst}$	Uniforme	$\frac{2C_{m\acute{a}x\ adm}}{\sqrt{12}} =$ $\frac{C_{m\acute{a}x\ adm}}{\sqrt{3}} =$ $\frac{10}{\sqrt{12}} \% =$ <b>2,8868</b>	<b>2,8868</b>	B	$\infty$
$C_E$	Uniforme	$\frac{E}{\sqrt{12}} = \frac{0,1}{\sqrt{12}} \text{ } ^\circ\text{C} =$ <b>0,0289</b>	<b>0,0289</b>	B	$\infty$
$C_{deriva}$	Uniforme	$\frac{2C_{der\ m\acute{a}x\ adm.}}{\sqrt{12}} =$ $= \frac{C_{der\ m\acute{a}x\ adm.}}{\sqrt{3}} =$ $= \frac{1,6}{\sqrt{12}} \% =$ <b>0,4619</b>	<b>0,4619</b>	B	$\infty$
Incertidumbre típica combinada		$u(\hat{H}_{Humdht22})$ $= \sqrt{u^2(\bar{P}_i) + u^2(C_{inst}) + u^2(C_E) + u^2(C_{deriva})}$ $=$ <b>2,9244 %</b>			
$u_{ef}$		<b>19.989.692</b>			

Tabla 9. Determinación de la incertidumbre típica combinada para la variable humedad relativa medida por el sensor DHT22 y sus gdl.

Una vez calculada la incertidumbre típica combinada, se procede a hallar el factor de cobertura, como no se cumple la hipótesis de normalidad, no se puede concluir que  $k$  sea 2, es por ello que se utiliza la Ecuación [20] para hallar los grados efectivos de libertad:

$$u_{ef} = \frac{u^4(y)}{\sum_{i=1}^N \frac{u_i^4(y)}{u_i}} = \frac{u(H_{Humdht22})^4}{\frac{u\bar{H}_i^4}{m-1} + \frac{u_{C_{inst}}^4}{\infty} + \frac{u_{C_E}^4}{\infty} + \frac{u_{C_{deriva}}^4}{\infty}} \rightarrow$$





$$v_{ef} = (m - 1) \left( \frac{u(\hat{H}_{Humdht22})}{u\bar{H}_i} \right)^4 = 5 \cdot \left( \frac{2,9244 \%}{0,0654 \%} \right)^4 = 19.989.692,43 = \mathbf{19.989.692}$$

Se irá a la tabla de la Figura 81 y entrando con  $v_{ef} = \infty$  se tendrá un factor de cobertura,  $k=2$ .

La incertidumbre expandida será:  $U = k \cdot u = 2 \cdot 2,9244 \% = \mathbf{5,8488 \%}$

Para la expresión final se redondeará este valor a 2 cifras significativas para evitar una subestimación por encima del 5 %, de modo que el valor de la incertidumbre expandida será de:  $\mathbf{5,8 \%}$ .

Por último, antes de expresar la medida, se debe ajustar de forma que tenga la misma resolución que la incertidumbre ajustada.

Entonces el resultado de la medida sería de:

$$\hat{H}_{Humdht22} = 30,2 \pm 5,8 \%$$

Para realizar la comprobación con el otro sensor, se analizará la humedad para el sensor BME280 como se muestra a continuación.



- FUNCIÓN MODELO PARA LA HUMEDAD RELATIVA DEL SENSOR BME280:

$$\mathbf{H}_{\text{Humbme280}} = \bar{\mathbf{H}}_i + \mathbf{C}_{\text{inst}} + \mathbf{C}_E + \mathbf{C}_{\text{deriva}}$$

$$\begin{aligned}\hat{\mathbf{H}}_{\text{Humbme280}} &= E(\mathbf{H}_i + \mathbf{C}_{\text{inst}} + \mathbf{C}_E + \mathbf{C}_{\text{deriva}}) \\ &= E(\bar{\mathbf{H}}_i) + E(\mathbf{C}_{\text{inst}}) + E(\mathbf{C}_E) + E(\mathbf{C}_{\text{deriva}})\end{aligned}$$

$$\begin{aligned}E(\bar{\mathbf{H}}_i) &= \frac{1}{m} \sum \mathbf{H}_i = \frac{29,42 + 29,88 + 29,86 + 29,63 + 29,76 + 29,64}{6} \\ &= \mathbf{29,6983 \%}\end{aligned}$$

$$E = 0,01 \% : \text{v. a.} \rightarrow U[-0,05, 0,05] \left\{ \begin{array}{l} E(\mathbf{C}_E) = \frac{-0,005 + 0,005}{2} = \mathbf{0 \%} \\ S(\mathbf{C}_E) = \frac{0,005 - (-0,005)}{\sqrt{12}} = \frac{\mathbf{0,01}}{\sqrt{12}} \% \end{array} \right.$$

$$\mathbf{C}_{\text{inst}} : \text{v. a.} \rightarrow U[-4,6, 4,6] \left\{ \begin{array}{l} E(\mathbf{C}_{\text{inst}}) = \frac{-4,6 + 4,6}{2} = \mathbf{0 \%} \\ S(\mathbf{C}_{\text{inst}}) = \frac{4,6 - (-4,6)}{\sqrt{12}} = \frac{\mathbf{9,2}}{\sqrt{12}} \% \end{array} \right.$$

$$\mathbf{C}_{\text{deriva}} : \text{v. a.} \rightarrow U[-\text{der}_{\text{mín}}, +\text{der}_{\text{máx}}]$$

$$\text{der}_{\text{máx.adm.}} = \pm \mathbf{0,8 \%}$$

Esta última componente, también sigue una distribución de tipo uniforme o rectangular.

En este punto seremos capaces de dar un valor de la estimación, que será la suma de las cuatro contribuciones que tenemos en la función modelo, siendo:

$$\hat{\mathbf{H}}_{\text{Humbme280}} = \mathbf{29,6983 \%} \pm U(\mathbf{H}_{\text{Humbme280}})$$

Los coeficientes de sensibilidad, son también unitarios, pudiéndose verificar a través de la función modelo. Para presentar los resultados de una forma clarificada y ordenada, se incorporarán a la Tabla 10.



Magnitud de entrada ( $X_i$ )	Distribución de probabilidad	Incertidumbre típica $u(x_i)$	Contribución (incertidumbre típica) $u_i(y) = c_i \cdot u(x_i)$	Tipo	gdl
$\bar{P}_i$	Normal	$\frac{S_i}{\sqrt{m}} = \frac{0,1725}{\sqrt{6}} =$ <b>0,0704</b>	<b>0,0704</b>	A	(m-1)= 6-1=5
$C_{inst}$	Uniforme	$\frac{2C_{m\acute{a}x adm}}{\sqrt{12}} =$ $\frac{C_{m\acute{a}x adm}}{\sqrt{3}} =$ $\frac{9,2}{\sqrt{12}} \% =$ <b>2,6558</b>	<b>2,6558</b>	B	$\infty$
$C_E$	Uniforme	$\frac{E}{\sqrt{12}} = \frac{0,01}{\sqrt{12}} \text{ } ^\circ C =$ <b>0,0029</b>	<b>0,0029</b>	B	$\infty$
$C_{deriva}$	Uniforme	$\frac{2C_{der m\acute{a}x.adm.}}{\sqrt{12}} =$ $= \frac{C_{der m\acute{a}x.adm.}}{\sqrt{3}} =$ $= \frac{1,6}{\sqrt{12}} \% =$ <b>0,4619</b>	<b>0,4619</b>	B	$\infty$
Incertidumbre típica combinada		$u(\hat{H}_{Humdht22})$ $= \sqrt{u^2(\bar{P}_i) + u^2(C_{inst}) + u^2(C_E) + u^2(C_{deriva})}$ $=$ <b>2,6966 %</b>			
$u_{ef}$		<b>10.763.301</b>			

Tabla 10. Determinación de la incertidumbre típica combinada para la variable humedad relativa medida por el sensor BME280 y sus gdl.

Una vez calculada la incertidumbre típica combinada, se procede a hallar el factor de cobertura, como no se cumple la hipótesis de normalidad, no se puede concluir que  $k$  sea 2, es por ello que se utiliza la Ecuación [20] para hallar los grados efectivos de libertad:

$$u_{ef} = \frac{u^4(y)}{\sum_{i=1}^N \frac{u_i^4(y)}{u_i}} = \frac{u(H_{Humbme280})^4}{\frac{u\bar{H}_i^4}{m-1} + \frac{u_{C_{inst}}^4}{\infty} + \frac{u_{C_E}^4}{\infty} + \frac{u_{C_{deriva}}^4}{\infty}} \rightarrow$$

$$v_{ef} = (m - 1) \left( \frac{u(\hat{H}_{Humbme280})}{u\hat{H}_i} \right)^4 = 5 \cdot \left( \frac{2,6966 \%}{0,0704 \%} \right)^4 = 10.763.301,07 = \mathbf{10.763.301}$$

Se irá a la tabla de la Figura 81 y entrando con  $v_{ef} = \infty$  se tendrá un factor de cobertura,  $k=2$ .

La incertidumbre expandida será:  $U=k \cdot u = 2 \cdot 2,6966 \% = \mathbf{5,3932 \%}$

Para la expresión final se redondeará este valor a 2 cifras significativas para evitar una subestimación por encima del 5 %, de modo que el valor de la incertidumbre expandida será de: **5,4 %**.

Por último, antes de expresar la medida, se debe ajustar de forma que tenga la misma resolución que la incertidumbre ajustada.

Entonces el resultado de la medida sería de:

$$\hat{H}_{Humbme280} = 29,7 \pm 5,4 \%$$

Previamente a la exposición de las conclusiones finales que se derivan del estudio, se mostrarán los resultados tabulados, véase Tabla 11.

	DHT22	BME280	Referencia Lab.
TEMPERATURA	$\hat{L}_{T^{a}dht22} = 20,2 \pm 1,1 \text{ } ^\circ\text{C}$	$\hat{L}_{T^{a}bme280} = 21,0 \pm 1,2 \text{ } ^\circ\text{C}$	$20,01 \pm 0,2 \text{ } ^\circ\text{C}$
HUMEDAD RELATIVA	$\hat{H}_{Humdht22} = 30,2 \pm 5,8 \%$	$\hat{H}_{Humbme280} = 29,7 \pm 5,4 \%$	$34,6 \pm 1,83 \%$
PRESIÓN	-	$\hat{P}_{Presionbme280} = 93886 \pm 390 \text{ Pa}$	$94328 \pm 45 \text{ Pa}$

Tabla 11. Estimaciones finales de las variables climáticas estudiadas.

En vistas a los resultados obtenidos, se ha podido comprobar que las estimaciones extraídas a partir de la muestra, se encuentran esclarecidamente dentro de los márgenes establecidos en los certificados de calibración, los cuales sustentan la referencia empleada como patrón para el estudio, de modo que estos económicos dispositivos se sitúan en un extraordinario lugar para optar por ellos en trabajos de medición de estas variables climáticas, como son temperatura, humedad relativa y presión. Se puede observar que el comportamiento de los sensores es el descrito en la situación inicial, el sensor DHT22 ofrece una medida más próxima al valor



nominal de la referencia, sin embargo, la componente de incertidumbre en el caso de la humedad, resulta ser mayor en este sensor que en el sensor BME280, lo cual se aleja de lo esperado a priori. Lo que sí deja evidencia, es que la respuesta de los sensores refleja muy bien la tendencia temporal y ofrece la medida en un rango de precisión admisible, resultando muy útil para recoger y controlar las variables climáticas en un entorno industrial, en que resulte de gran valor el conocimiento y control de las mismas.



Universidad de Valladolid

IMPLANTACIÓN, ESTUDIO Y CONTROL DE CALIDAD DE VARIABLES  
CLIMÁTICAS EN EL ÁMBITO DE LA METROLOGÍA 4.0.

Capítulo 5. Parte experimental y resultados.



ESCUELA DE INGENIERÍAS  
INDUSTRIALES



Universidad de Valladolid

IMPLANTACIÓN, ESTUDIO Y CONTROL DE CALIDAD DE VARIABLES  
CLIMÁTICAS EN EL ÁMBITO DE LA METROLOGÍA 4.0.

Capítulo 6. Estudio económico.



ESCUELA DE INGENIERÍAS  
INDUSTRIALES

# **CAPÍTULO 6.**

## **ESTUDIO ECONÓMICO.**



## 6.1 RECURSOS UTILIZADOS.

Los recursos empleados en este proyecto se detallan a continuación, clasificados conforme su tipología:

- Recursos informáticos (hardware y software):

- Ordenador portátil Packard Bell Intel(R) Core(TM) i3 CPU M350 con 4,00 GB de memoria RAM.
- Sistema operativo Windows 7 de 64 bits.
- Las aplicaciones ofimáticas utilizadas son: Microsoft Office 2019, plataforma Thingier.io y Arduino 1.8.8.

- Recursos materiales de oficina:

- Papel DIN A4: 210x297 mm.
- CD-ROM.
- Recursos de impresión.

- Recursos materiales de laboratorio y dispositivos periféricos:

- Sensores DHT22 y BME280.
- Dos placas NodeMCU v3-ESP8266.
- USB, conector y cables de puente flexibles.
- Equipos empleados como referencia patrón:
  - Termohigrómetro (marca ALMEMO, modelo 3290-8, código L9708119 (E1/104)).
  - Medidor de presión (marca ALMEMO, modelo FD A612-MA, código 10203887 (E1/110)).

## 6.2 COSTES DEL PROYECTO.

### 6.2.1. COSTES DIRECTOS.

#### COSTE DEL PERSONAL

En este apartado se incluirán las horas empleadas en la ejecución del proyecto de modo que se expondrán los costes totales del personal.





Balance anual detallado	Días/año
Días anuales	365
Sábados y domingos	-104
Días efectivos de vacaciones	-30
Días festivos	-14
Días pedidos por diversas causas: enfermedad, ... (cálculo estimativo)	-5
<b>Total de días laborables</b>	<b>212</b>

Tabla 12. Determinación de los días hábiles al año.

Una vez se tienen los días hábiles calculados, si la jornada laboral es de 8 h/día, se multiplicará esta tasa al total de días laborables/año hallado y se obtendrán las horas totales trabajadas a lo largo del año, siendo de 1.696 h/año.

El proyecto ha sido desarrollado y analizado por un Ingeniero Mecánico, su salario anual es de unos 30.000,00 € aproximadamente, incluyendo en el mismo, el sueldo neto junto con las prestaciones de IRPF y demás contribuciones a la Seguridad Social. Es por ello que el precio/hora del trabajador se calculará dividiendo los 30.000,00 € anuales entre las 1.696 h/año, resultando ser de 17,70 €/h.

Por último, para determinar el coste total del personal, se calcularán las horas empleadas en la ejecución de este TFG y se multiplicarán a la tasa del precio/hora del Ingeniero. Es a través de la Tabla 13 donde se segregarán las horas destinadas al proyecto en función de las tareas realizadas.

Tareas	Horas
Planificación del estudio y subdivisión del trabajo.	25
Documentación.	150
Sección experimental.	120
Realización de la memoria.	160
<b>Horas totales realización proyecto.</b>	<b>455</b>

Tabla 13. Cálculo horas totales empleadas en la ejecución del proyecto.

Disponiendo de las horas totales del proyecto que multiplicadas por la tasa de 17,70 €/h que se había calculado previamente, tiene como objeto poner de manifiesto el coste total del personal que asciende a **8.048,35 €**.



## COSTE DE MATERIAL AMORTIZABLE

En esta sección de costes, se incluirán los equipos informáticos así como el software, los cuales suelen quedarse obsoletos una vez que alcanza su período estimable de amortización.

Se tomará un valor de coste actual y su amortización lineal en el tiempo, expuestas junto al mismo las horas de trabajo del elemento en cuestión. Se considerará el precio del equipo empleado, la computadora portátil, así como las licencias del sistema operativo (Windows 7) y la del paquete office (Microsoft 2019).

En estos costes no sería necesario incluir ni la plataforma Thinger.io ni Arduino, pues su hardware y software son de código abierto (Open Source).

Elemento	Precio (€)	Años	Días	Horas	Coste Final (€)
Ordenador portátil Packard Bell Intel(R) Core(TM) i3 CPU M350 con 4,00 GB de memoria RAM.	600,00	7	212	300	15,16
Microsoft Windows 7.	200,00	7	212	300	5,05
Microsoft Office 2019.	150,00	5	212	250	4,42
Sensores DHT22 y bme280.	7,44	3	212	360	0,53
Dos placas NodeMCU v3-ESP8266.	11,86	3	212	360	0,53
USB, conector y cables de puente flexibles.	17,54	5	212	360	0,74
Termohigrómetro.	240,00	8	212	360	6,37
Medidor de presión.	471,00	8	212	360	12,50
<b>Total</b>	<b>1.697,84</b>				<b>45,30</b>

Tabla 14. Coste total del material amortizable.

Para el cálculo de los costes amortizables de los equipos y licencias ofimáticas empleadas, se ha procedido como se muestra a continuación a través del ejemplo del cálculo amortizable del ordenador portátil empleado, multiplicando el número de años por los días hábiles de un año y una jornada laboral de 8 h/día, todo ello dividiendo al producto resultante de las horas que se utiliza ese elemento por el precio del mismo.

$$\text{Coste Final} = \frac{600,00 \text{ €}}{7 \text{ años} \cdot \frac{212 \text{ d}}{\text{año}} \cdot \frac{8 \text{ h}}{\text{d}}} \cdot 300 \text{ h} = 15,16 \text{ €}$$

Por lo que resultaría un coste de amortización total de **45,30 €**.

### **COSTE DE MATERIAL NO AMORTIZABLE**

Habría que añadir a los costes anteriores, los costes procedentes de todo aquel material que no sea amortizable con el tiempo, en la Tabla 15 se recogen estos costes.

Elemento	Coste Final (€)
Papel DIN A4: 210x297mm	20,00
Recursos de impresión	50,00
Otros	35,00
Total sin IVA	105,00
IVA 21%	22,05
<b>Total (incorporado IVA)</b>	<b>127,05</b>

Tabla 15. Coste total del material no amortizable.

Es por ello que el coste del material no amortizable asciende a **127,05 €**. De este modo, los costes directos totales se recogen en la Tabla 16.

Categoría costes directos	Costes (€)
Coste del personal	8.048,35
Coste de bienes amortizables	45,30
Coste de bienes no amortizables	127,05
<b>Total</b>	<b>8.220,7</b>

Tabla 16. Costes directos totales del proyecto.

### **6.2.2. COSTES INDIRECTOS.**

En este apartado se recogen aquellos costes que no se pueden asociar a un único elemento de referencia, sino que hacen alusión a referencias más generalizables. Dentro de ellos se encontrarían el gasto de electricidad, climatización, iluminación entre otros. Éstos aparecerán recogidos en la Tabla 17.

Elemento	Coste Final (€)
Gasto eléctrico de los dispositivos y equipos	100,00
Climatización/Calefacción	80,00
Iluminación	60,00
Otros	90,00
<b>Total</b>	<b>330,00</b>

Tabla 17. Costes indirectos totales del proyecto.

Los costes indirectos resultarían ser de **330,00 €**.

### 6.3 COSTE TOTAL DEL PROYECTO.

El coste total se determina sumando a los costes totales directos calculados, los costes indirectos totales, véase Tabla 18.

Elemento	(€)
Costes directos totales	8.220,70
Costes indirectos totales	330,00
<b>Coste Total realización proyecto</b>	<b>8.550,70</b>

Tabla 18. Costes totales realización del proyecto.

Resultando ser el coste total del proyecto de **8.550,70 €**.

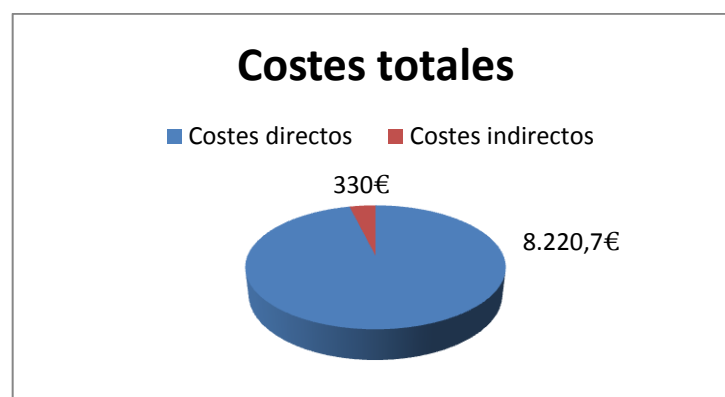


Figura 83. Costes totales del proyecto.



## **CAPÍTULO 7.**

# **CONCLUSIONES Y LÍNEAS FUTURAS.**



## 7.1 CONCLUSIONES.

En este capítulo se expondrán las conclusiones de este TFG, siendo las siguientes:

- Se ha implementado la sensorización dentro del ámbito de la Metrología 4.0 demostrando el comportamiento de los dos sensores empleados para la medición de variables climáticas, ofreciendo una alternativa al programa de Arduino, posibilitando la recogida de datos a través de vía WiFi por medio de una plataforma informática, Thingier.io, siendo a su vez posible el control remoto de las variables a través de la aplicación móvil anteriormente citada.
- Se ha establecido un proceso de corrección y calibración del error existente, determinando una función modelo para cada una de las variables medidas junto con sus respectivos sensores, ofreciendo y garantizando de este modo la trazabilidad necesaria.
- Se ha propuesto una optimización del proceso de toma de datos por medio de funciones aditivas al código del programa, que permiten obtener un mayor rendimiento de las placas, aminorando así el gasto de energía, aumentando consigo los ciclos y la vida útil, mostrando para ello un caso práctico, utilizando una batería idéntica.
- Existen una serie de limitaciones que no permiten obtener la medición exacta, pero se deja evidencia que la componente aleatoria del error, basada ésta en parámetros estocásticos, lejos del control automatizado o manual por parte del ser humano, verá afectada a toda medida realizada, no obstante, el error sistemático sí se ha conseguido reducir en la mayor medida posible, ofreciendo para ello un método de corrección.



## 7.2 LÍNEAS FUTURAS.

Este TFG deja abiertas algunas propuestas en línea al estudio presentado para trabajos futuros:

- El método empleado para el análisis se fundamentaba en reducciones promedias, siendo la media de una muestra aleatoria un estadístico que permite esclarecer el estudio llevado a cabo, no obstante se propone la generación de un algoritmo que cuantifique el error de forma instantánea, minimizando esta discrepancia y que posibilite así un resultado con mayor confiabilidad.

- El método de programación podría ser suplantado o comparada su eficiencia con otros, como bien puede ser Python que también dispone de licencia de código abierto.

- Determinar el lugar más apropiado para la realización de la medida, para lo que se propone llevar a cabo el experimento en un sitio en que se encuentren totalmente constantes los parámetros que infieran en el resultado de las variables/parámetros medidos.

Este estudio se ha llevado a cabo en un entorno industrial, en que influye el entorno, las personas que se encuentran trabajando en el laboratorio, entre otros factores; pese a ello cabe recordar lo económicos que son estos sensores, y el gran balance que ofrecen calidad-precio, resultando ser muy efectivos en términos de control de calidad y garantizando la toma de medidas u órdenes de actuación frente a diversas situaciones.



Universidad de Valladolid

IMPLANTACIÓN, ESTUDIO Y CONTROL DE CALIDAD DE VARIABLES  
CLIMÁTICAS EN EL ÁMBITO DE LA METROLOGÍA 4.0.

Capítulo 7. Conclusiones y líneas futuras.



ESCUELA DE INGENIERÍAS  
INDUSTRIALES





Universidad de Valladolid

IMPLANTACIÓN, ESTUDIO Y CONTROL DE CALIDAD DE VARIABLES  
CLIMÁTICAS EN EL ÁMBITO DE LA METROLOGÍA 4.0.

Capítulo 8. Bibliografía.



ESCUELA DE INGENIERÍAS  
INDUSTRIALES

# **CAPÍTULO 8.**

# **BIBLIOGRAFÍA**



## BIBLIOGRAFÍA

---

### LIBROS Y DOCUMENTOS CIENTÍFICOS

- [1] Joyanes Aguilar, Luis. (2018) Industria 4.0: la cuarta revolución industrial. Barcelona: Alfaomega: Marcombo.
- [2] Monk, Simon. (2017) Programación de Arduino: introducción a sketches. Madrid: Anaya, D.L.
- [3] Oliva Ramos, Rubén. (2018) Monitoreo, control y adquisición de datos con Arduino y Visual Basic.Net., Barcelona: Marcombo.
- [4] Aliverti, Paolo. (2016) El manual de Arduino. Barcelona: Marcombo.
- [5] Ministerio de Industria, Comercio y Turismo. (2018). II Congreso Industria Conectada 4.0. Palacio Municipal de Congresos Madrid.  
<https://cic40.es/tag/industria-4-0/>
- [6] CTI Reseau. (1996) – Centres Techniques Industriels. Metrología. Práctica de la Medida en la Industria. AENOR. ISBN: 2-12-460702-2.
- [7] Coca Rebollero, Pedro y Rosique Jiménez, Juan. (2009) Ed. Pirámide. Tecnología Mecánica y Metrotecnica.
- [8] Perruchet, Christophe y Priel, Marc. Aenor (2001) Estimación de la incertidumbre.: Medidas y ensayos.
- [9] Sebastián Pérez. Miguel A.; Bargueño Fariñas, Vicente y Novo Sanjurjo, Vicente José. Cuadernos de la UNED (2008). Gestión y control de calidad.
- [10] González Gaya, Cristina; Domingo Navas, Rosario y Sebastián Pérez, Miguel A. Cuadernos de la UNED (2008). Técnicas de mejora de la calidad.
- [11] (2000) Guía para la Expresión de la Incertidumbre de Medida. Metrología. CEM. 2ª Edición en español.

### PÁGINAS WEB CONSULTADAS (Fecha última consulta: Junio 2019)

- [12] Industria 4.0: Global Digital Operations Study 2018.Strategy&, consultora de PwC. Madrid, 2018.  
<https://www.pwc.es/es/productos-industriales/industria-4-0-global-digital-operations-study-2018.html>
- [13] En las fábricas de BMW, los robots trabajan junto a las personas. Will Knight. 25 Marzo 2014.



<https://www.infotechnology.com/negocios/En-las-fabricas-de-BMW-los-robots-trabajan-junto-a-las-personas-20140321-0002.html>

[14] El software del Tesla Model 3 se actualiza y recorta distancias con los Model X y S. Iván Martín. 24 Junio, 2018.

<https://topesdegama.com/noticias/motor/tesla-model-3-actualiza-WiFi-autopilot>

[15] <https://www.diariomotor.com/coche/tesla-model-s/>

[16] Barcelona, 26 Febrero, 2019. (Portaltic/EP).

<https://www.europapress.es/portaltic/gadgets/noticia-wearlumb-camiseta-inteligente-detecta-riesgos-posturales-trabajo-ayudar-corregirlos-20190226114511.html>

[17] [https://www.elconfidencial.com/tecnologia/2019-03-06/wearlumb-camiseta-inteligente-lesiones-espalda\\_1862530/](https://www.elconfidencial.com/tecnologia/2019-03-06/wearlumb-camiseta-inteligente-lesiones-espalda_1862530/)

[18] <http://www.soltel.es/en/intrace/>

[19] Abril, 2019. ABC.

<https://directivosygerentes.es/innovacion/noticias-innovacion/tecnologia-cultivo>

[20] Información sobre la plataforma de Arduino y sus funciones principales.

<https://playground.arduino.cc/ArduinoNotebookTraduccion/Structure>

[21] Modos de ahorro de energía –DEEP SLEEP-. Enero, 2017.

<https://www.esploradores.com/practica-9-modos-de-ahorro-de-energia-deep-sleep/>

[22] <https://www.luisllamas.es>

[23] <https://www.prometec.net/sensores-dht11/>

[24] <http://www.electrogeekshop.com/resistencias-pull-up-y-pull-down/>

[25] IBM, FABRICACIÓN.

[https://www.ibm.com/es-es/industries/manufacturing/smart-manufacturing-technology?S\\_PKG=AW&cm\\_mmc=Search\\_Google- -missing\\_category\\_missing\\_category- -WW\\_ES- -+industria++4.0\\_Broad\\_AW&cm\\_mmca1=000032VS&cm\\_mmca2=10005255&mkwid=k\\_Cj0KCQjw6lfoBRCiARIsAF6q06sPxMIDy2ssx4ziT5llyZwyuxQL6F3JB0eFTI9vrY7xN8GnDMC2RT0aAp49EALw\\_wcB\\_k\\_|594|12387&gclid=Cj0KCQjw6lfoBRCiARIsAF6q06sPxMIDy2ssx4ziT5llyZwyuxQL6F3JB0eFTI9vrY7xN8GnDMC2RT0aAp49EALw\\_wcB](https://www.ibm.com/es-es/industries/manufacturing/smart-manufacturing-technology?S_PKG=AW&cm_mmc=Search_Google- -missing_category_missing_category- -WW_ES- -+industria++4.0_Broad_AW&cm_mmca1=000032VS&cm_mmca2=10005255&mkwid=k_Cj0KCQjw6lfoBRCiARIsAF6q06sPxMIDy2ssx4ziT5llyZwyuxQL6F3JB0eFTI9vrY7xN8GnDMC2RT0aAp49EALw_wcB_k_|594|12387&gclid=Cj0KCQjw6lfoBRCiARIsAF6q06sPxMIDy2ssx4ziT5llyZwyuxQL6F3JB0eFTI9vrY7xN8GnDMC2RT0aAp49EALw_wcB)



Universidad de Valladolid

IMPLANTACIÓN, ESTUDIO Y CONTROL DE CALIDAD DE VARIABLES  
CLIMÁTICAS EN EL ÁMBITO DE LA METROLOGÍA 4.0.

Capítulo 8. Bibliografía.



ESCUELA DE INGENIERÍAS  
INDUSTRIALES



Universidad de Valladolid

IMPLANTACIÓN, ESTUDIO Y CONTROL DE CALIDAD DE VARIABLES  
CLIMÁTICAS EN EL ÁMBITO DE LA METROLOGÍA 4.0.

Capítulo 9. Anexos.



ESCUELA DE INGENIERÍAS  
INDUSTRIALES

## **CAPÍTULO 9**

### **ANEXOS**



## ANEXO 1. Entorno de programación.

P1.- Programa establecimiento de conexión de la placa con el WiFi utilizado, uva\_IOT en este TFG:

---

```
#include <ESP8266WiFi.h>

char ssid[] = "uva_IOT";
char password[] = "www.uva.es";

void setup() {
  Serial.begin(9600);
  Serial.println("WiFi test!");

  WiFi.begin(ssid, password);
  while (WiFi.status() != WL_CONNECTED) {
    Serial.print(".");
    delay(500);
  }

  Serial.println("");
  Serial.println("WiFi connected");
  Serial.println(WiFi.localIP());
}

void loop() {
}
```



**P2.- Programa medición de las variables temperatura y humedad relativa por el sensor DHT22 y expresión de resultados a través del monitor serie por pantalla, expresada la temperatura en grados Celsius y en Fahrenheit (en este ejemplo de programa se puede observar cómo con un factor de calibración se corregiría la medida):**

---

```
#include "DHT.h"

#define DHTPIN 4 //pin al que se encuentra conectado el sensor DHT22
#define DHTTYPE DHT22 //dentro de la familia de los sensores DHT
especificar el tipo

DHT dht(DHTPIN, DHTTYPE);

void setup() {
  Serial.begin(9600);
  Serial.setTimeout(2000);

  while(!Serial) {}

  Serial.println("Device Started");
  Serial.println("-----");
  Serial.println("Running DHT!");
  Serial.println("-----");
}

int timeSinceLastRead = 0;
void loop() {

  // Toma de datos cada 2 segundos, expresados en milisegundos ->2000
  if(timeSinceLastRead > 2000) {

    float h = dht.readHumidity();
    // Read temperature as Celsius (the default)
    float t = dht.readTemperature();
    // Read temperature as Fahrenheit (isFahrenheit = true)
    float f = dht.readTemperature(true);

    // Factor de calibración de la temperatura
    float cal_temp = 0;

    // Comprobación de que las lecturas sean las correctas
    if (isnan(h) || isnan(t) || isnan(f)) {
```

Factor de corrección que habría que introducir para que la medida que recoja el programa sea la correcta.



```
Serial.println("Failed to read from DHT sensor!");
timeSinceLastRead = 0;
return;
}

//temperatura en grados Fahrenheit
float hif = dht.computeHeatIndex(f, h);
// temperatura en grados Celsius (isFahreheit = false)
float hic = dht.computeHeatIndex(t, h, false);

Serial.print("Humidity: ");
Serial.print(h);
Serial.print(" %\t");
Serial.print("Temperature: ");
Serial.print(t+cal_temp);
Serial.print(" *C ");
Serial.print(f);
Serial.print(" *F\t");
Serial.print("Heat index: ");
Serial.print(hic);
Serial.print(" *C ");
Serial.print(hif);
Serial.println(" *F");

timeSinceLastRead = 0;
}
delay(100);
timeSinceLastRead += 100;
}
```





### P3.- Programa establecimiento conexión plataforma Thingier.io con la placa NodeMCU para la lectura de datos del sensor BME280.

---

```
#define _DISABLE_TLS_
#include <ESP8266WiFi.h> //Librería dentro del procesador ESP8266 para
establecer la conexión WiFi
#include <ThingierESP8266.h> //Librería de la plataforma Thingier.io
#include <Wire.h>
#include <SPI.h> //Bus de datos empleado, en este caso es el SPI
#include <Adafruit_Sensor.h>
#include <Adafruit_BME280.h>

// Parámetros de establecimiento de conexión con la plataforma Thingier.io
#define usuario "nubethmcu2k19"
#define device_Id "IdNodeMCU2"
#define device_credentials "DPjmjhPN9hu@"

ThingierESP8266 thing(usuario, device_Id, device_credentials);

// Parámetros el establecimiento de la conexión vía WiFi
const char WiFi_ssid[]="uva_IOT"; //Nombre de la red
const char WiFi_password[]="www.uva.es"; //Contraseña de red

// asignación de los pines de la placa a Arduino
#define D1 5
#define D2 4
#define D4 2
#define D3 0

// asignación del bus de datos SPI a los pines de la placa
#define BME_SCK D1
#define BME_MISO D4
#define BME_MOSI D2
#define BME_CS D3

#define SEALEVELPRESSURE_HPA (1012.9)

Adafruit_BME280 bme(BME_CS, BME_MOSI, BME_MISO, BME_SCK); //
software SPI

unsigned long delayTime;

void setup() {

// Inicialización del sensor BME280
bme.begin();
```



```
// Inicialización de la red WiFi para establecer la comunicación con la API
thing.add_WiFi(WiFi_ssid, WiFi_password);

// Lectura de datos desde la API
thing["bme280"] >> [(pson& out){
  out["Temperatura"] = bme.readTemperature();
  out["Humedad"] = bme.readHumidity();
  out["Presion"] = bme.readPressure();
  };
}

void loop() {
  thing.handle();
}
```