



Universidad de Valladolid



**ESCUELA DE INGENIERÍAS
INDUSTRIALES**

UNIVERSIDAD DE VALLADOLID

ESCUELA DE INGENIERIAS INDUSTRIALES

Grado en Ingeniería en Organización Industrial

**Desarrollo de un sistema de control de
inventario en tiempo real mediante
tecnología RFID.**

Autor:

Moro Pereda, Fernando

Tutor(es):

**del Valle González, María Isabel
Dpto. Tecnología Electrónica**

Valladolid, julio 2024.



Agradecimientos

Quiero agradecer el tiempo, la dedicación y el apoyo de María Isabel y de Oliver que me han asistido durante la realización de este Trabajo de Fin de Grado.



Desarrollo de un sistema de control de inventario en tiempo real mediante tecnología RFID





Resumen

Con el propósito de facilitar el control de los movimientos del inventario en un almacén con un coste bajo, se diseña en este trabajo un sistema de gestión sencillo y eficaz. Este sistema está basado en la tecnología RFID, utilizando Arduino como plataforma “hardware”. Se registran las entradas y las salidas de productos, así como información adicional de cada uno de ellos, relevante para su gestión. Los datos recopilados se almacenan en una base de datos y se podrán visualizar en tiempo real. Además, se realiza un estudio económico para poder evaluar la viabilidad económica de forma cuantitativa. Este sistema pretende reducir el error humano, incrementar la rapidez de gestión y tratar y presentar adecuadamente la información para facilitar el control y la toma de decisiones.

Palabras clave: RFID, logística, Arduino, monitorización y almacén.

Abstract

In order to facilitate the control of inventory movements in a warehouse, at a reduced cost, a simple and effective management system was designed in this document. This system is based on RFID technology, making use of Arduino as the hardware platform. The product's entries and exits are registered, with additional information relative to each one of them, that is relevant to their management. The obtained data is stored in a database and can be visualized in real time. Moreover, an economic study is made to evaluate the economic viability in a quantitative manner. This system strives to reduce human error, increasing the management speed, processing data and properly displaying information to reduce the control effort and simplify the decision-making process.

Key words: RFID, logistics, Arduino, monitoring and warehouse.



Desarrollo de un sistema de control de inventario en tiempo real mediante tecnología RFID





ÍNDICE

Capítulo 1. Introducción y objetivos	13
1.1. Introducción	15
1.2. Objetivos	15
1.3. Estructura del trabajo.....	17
Capítulo 2. Estado del arte	19
2.1. RFID	21
2.1.1. Introducción.....	21
2.1.2. Comparación de la tecnología RFID con los códigos de barras para la automatización de almacenes	22
2.1.3. Elementos del sistema	22
2.1.4. Tipos de tecnología RFID	27
2.1.5. Aspectos de seguridad	33
2.1.6. Innovaciones en tecnología RFID.....	35
2.2. Almacenes.....	36
2.2.1. Introducción.....	36
2.2.2. Tipos de almacenes	37
2.2.3. Gestión de almacenes	38
2.2.4. Principios de almacenamiento.....	41
2.2.5. Principios de transporte	44
2.2.6. Innovaciones en gestión de almacenes	46
Capítulo 3. Desarrollo del Sistema	49
3.1. Introducción	51
3.2. Hardware.....	51
3.2.1. ¿Qué es Arduino?	51
3.2.2. Justificación de su elección.....	52
3.2.3. Arduino Mega 2560 r3	53
3.2.4. Sensor RFID MFRC522	57
3.2.5. Tarjetas MIFARE	58
3.2.6. Pantalla TFT	62



3.2.7. Pulsador.....	63
3.2.8. LEDs.....	64
3.2.9. Protoboard.....	65
3.2.10. Resistencias.....	66
3.2.11. Conexiones serie: UART y SPI.....	67
3.3. Prototipo.....	69
3.3.1. Diagramas del circuito.....	71
Capítulo 4. Programación y guía de uso.....	75
4.1. Introducción.....	77
4.2. Arduino.....	77
4.2.1. Librerías empleadas.....	77
4.2.2. Inicialización.....	79
4.2.3. Bucle principal.....	80
4.2.4. Funciones Leer y Escribir.....	81
4.2.5. Función ExtraerDatosExcel.....	85
4.2.6. Funciones serialEvent, IndicarUbicación y ParpadeoLED.....	86
4.3. Guía de uso de la aplicación de Excel.....	88
4.3.1. Hoja Cuadro de control.....	89
4.3.2. Hoja Planificación y resultados.....	90
4.3.3. Hoja Inventario diario, Entradas, Almacenado y Salidas.....	92
4.3.4. Hojas de almacén (Niveles 1, 2 y 3).....	93
4.3.5. Hoja de Datos.....	94
4.3.6. Hojas Entrada de datos, Salida de datos y Configuración.....	95
4.4. Excel VBA.....	95
4.4.1. Código de la hoja Entrada de datos.....	96
4.4.2. Subrutina CopiarDatosEntrantes.....	98
4.4.3. Subrutina CopiarDirección.....	98
4.4.4. Subrutinas ActualizarInventario y LugarAlmacén.....	100
4.4.5. RecepciónDeLotesDePalés y EnvíoDeLotesDePalés.....	103
4.4.6. Subrutina InventarioDiario.....	106
4.4.7. Subrutina AjustarEjeXGrafica.....	108
4.4.8. Subrutina EliminarDatos.....	108



Capítulo 5. presentación de Resultados	109
5.1. Introducción	111
5.2. Hipótesis	111
5.3. Simulación	112
5.3.1. Situación inicial	112
5.3.2. Planificación de movimientos	114
5.3.3. Recepción y envío de pedidos.....	115
5.3.4. Evaluación del desempeño	116
Capítulo 6. Estudio económico	119
6.1. Introducción	121
6.2. Costes.....	121
6.2.1. Costes de personal y reparto de tiempos.....	121
6.2.2. Amortizaciones	123
6.2.3. Costes materiales	124
6.2.4. Otros costes directos	124
6.2.5. Costes indirectos.....	124
6.3. Presupuesto final.....	125
6.3.1. Coste total desglosado por tareas	125
6.3.2. Coste total desglosado por partidas de gasto	128
Capítulo 7. Conclusiones y líneas de mejora.....	131
7.1. Conclusiones.....	133
7.2. Líneas de mejora	133
Bibliografía.....	137
Anexos.....	143



ÍNDICE DE FIGURAS

Figura 1. Diferentes redes inalámbricas en función de su cobertura.....	21
Figura 2. Esquema básico de un sistema RFID	23
Figura 3. Componentes de un lector RFID	24
Figura 4. Componentes del transpondedor	24
Figura 5. Clasificación de los transpondedores.....	26
Figura 6. Dos etiquetas RFID	27
Figura 7. Clasificación de los sistemas RFID	28
Figura 8. Modos de comunicación.....	29
Figura 9. Alcance de diferentes sistemas RFID	30
Figura 10. Acoplamiento inductivo	31
Figura 11. Acoplamiento eléctrico	32
Figura 12. Acoplamiento electromagnético	32
Figura 13. Disposición en bloque	41
Figura 14. Disposición en línea.....	41
Figura 15. Palé situado longitudinalmente	42
Figura 16. Lado frontal del palé.....	42
Figura 17. Estantería “mini-load” con transelevador automático	43
Figura 18. Estanterías de bloque estático	43
Figura 19. Sistema de islas móviles	44
Figura 20. Componentes de la industria 4.0	48
Figura 21. Arduino Mega 2560 Rev3	54
Figura 22. Diagrama de bloques simplificado del microcontrolador ATmega 2560	56
Figura 23. Nuevo proyecto en el Arduino IDE	57
Figura 24. Sensor MFRC522 empleado para la elaboración del prototipo.....	58
Figura 25. Diagrama de una tarjeta MIFARE	58
Figura 26. Diagrama de bloques del chip de la tarjeta MIFARE S50.....	60
Figura 27. Estructura de la memoria EEPROM de la tarjeta MIFARE S50.....	61
Figura 28. Tarjeta y llaveros MIFARE S50	62
Figura 29. Pantalla 1.8 TFT 8P 128*160	63
Figura 30. Pulsador.....	64
Figura 31. Montaje Pull-Down de un pulsador.....	64
Figura 32. LEDs.....	65
Figura 33. Protoboard.....	65
Figura 34. Resistencias	66
Figura 35. Cable USB tipo A a USB tipo B, conexión PC-Arduino.....	68
Figura 36. Prototipo del proyecto.....	70
Figura 37. Diagrama del circuito Arduino-periféricos.....	72



Figura 38. Conexiones de la pantalla TFT, pulsador y LEDs	73
Figura 39. Conexiones de los LEDs y los lectores RFID	73
Figura 40. Inicialización	80
Figura 41. Bucle principal.....	81
Figura 42. Función Leer.....	83
Figura 43. Función Escribir.....	84
Figura 44. Función ExtraerDatosExcel.....	85
Figura 45. Función serialEvent	86
Figura 46. Función IndicarUbicación	87
Figura 47. Función ParpadeoLED	88
Figura 48. Hoja "Almacén-N1", ejemplo LIFO.....	93
Figura 49. Subrutina de la hoja "Entrada de datos"	97
Figura 50. Subrutina CopiarDatosEntrantes	98
Figura 51. Subrutina CopiarDirección	99
Figura 52. Subrutina ActualizarInventario.....	101
Figura 53. Subrutina LugarAlmacén	102
Figura 54. Subrutina RecepciónDeLotesDePalés.....	104
Figura 55. Subrutina EnvíoDeLotesDePalés	105
Figura 56. Subrutina InventarioDiario	107
Figura 57. Subrutina AjustarEjeXGráfica	108
Figura 58. Estado del almacén al comienzo del 20/06/2024 de la simulación	113
Figura 59. Inventario al comienzo del día 21/06/2024.....	114
Figura 60. Ejemplo de entrada de un producto al almacén	115
Figura 61. Gráficas del cuadro de control al final de la simulación	117
Figura 62. Gráfico circular de las horas dedicadas al desarrollo del proyecto	122
Figura 63. Gráfico circular del coste por tarea del proyecto.....	128
Figura 64. Gráfico circular sobre el coste por partida de gasto	129



ÍNDICE DE TABLAS

Tabla 1. Comparación de las memorias FRAM con las EEPROM	25
Tabla 2. Frecuencias en función del método de acoplamiento y distancia de lectura	29
Tabla 3. Ejemplos de KPI para la gestión de un almacén.....	40
Tabla 4. Especificaciones técnicas del Arduino Mega 2560 Rev3	54
Tabla 5. Especificaciones técnicas del lector MFRC522	58
Tabla 6. Especificaciones técnicas de la tarjeta MIFARE S50.....	59
Tabla 7. Conexiones de la pantalla con Arduino.....	63
Tabla 8. Código de colores de las resistencias.....	66
Tabla 9. Resumen de componentes del prototipo	71
Tabla 10. Canales de salida del transmisor de datos de Excel.....	85
Tabla 11. Ejemplo de planificación de un movimiento. Parte 1.....	92
Tabla 12. Ejemplo de planificación de un movimiento. Parte 2.....	92
Tabla 13. Ejemplo de planificación de un movimiento. Parte 3.....	92
Tabla 14. Ejemplo de planificación de un movimiento. Parte 4.....	92
Tabla 15. Ejemplo de inventario diario.....	92
Tabla 16. Entradas.....	94
Tabla 17. Productos almacenados	94
Tabla 18. Productos enviados o salidas.....	94
Tabla 19. Ejemplo de tabla de Planificación y resultados	106
Tabla 20. Ejemplo de tabla de Inventario diario.....	106
Tabla 21. Registro de productos almacenados al comienzo del 20/06/2024	113
Tabla 22. Columnas de proyección de productos	114
Tabla 23. Planificación de recepciones y envíos	115
Tabla 24. Planificación y resultados al final del 20/06/2024.....	116
Tabla 25. Planificación y resultados al final del 21/06/2024.....	116
Tabla 26. KPIs de la simulación.....	117
Tabla 27. Horas dedicadas a cada actividad.....	122
Tabla 28. Coste del ingeniero en organización industrial	123
Tabla 29. Amortizaciones.	123
Tabla 30. Costes materiales del prototipo	124
Tabla 31. Otros costes directos	124
Tabla 32. Costes indirectos.....	125
Tabla 33. Costes de redacción de la memoria	125
Tabla 34. Costes de programación del archivo Excel.....	126
Tabla 35. Costes de la recopilación de información	126
Tabla 36. Costes de programación Arduino.....	126



Tabla 37. Costes de diseño y montaje del circuito.....	127
Tabla 38. Coste total desglosado por actividades.....	127
Tabla 39. Coste total desglosado por partidas de gasto.....	128



Desarrollo de un sistema de control de inventario en tiempo real mediante tecnología RFID





CAPÍTULO 1. INTRODUCCIÓN Y OBJETIVOS



Desarrollo de un sistema de control de inventario en tiempo real mediante tecnología RFID





1.1. Introducción

El esfuerzo de las empresas por mantener su competitividad en el mercado alcanza a todas las ramas que la componen. En particular, en el ámbito de la logística el mercado actual exige cada vez con más intensidad la inmediatez, la transparencia y la calidad del servicio. Esto motiva la introducción de nuevas tecnologías como el RFID, que mediante el uso de ondas electromagnéticas, permite identificar a distancia objetos y productos junto con sus características relevantes.

Esta tecnología presenta ventajas sobre otros métodos para la gestión de almacenes más establecidos como los códigos de barras. Entre las ventajas se encuentran la automatización, la identificación de productos a mayores distancias, la identificación de múltiples productos simultáneamente, la reducción de errores y la reducción de tiempos. Sin embargo, no es una tecnología libre de inconvenientes, como puede ser el coste, por esto, alternativas como los códigos de barras no serán sustituidas de inmediato.

La industria tiende a la digitalización de todos sus procesos, integrando progresivamente tecnologías como el internet de las cosas, la realidad aumentada, robots autónomos e incrementando la comunicación entre diferentes dispositivos de la fábrica, lo que se conoce como la industria 4.0 y el RFID formará parte ella.

En este trabajo se desarrolla un sistema de control de un almacén en tiempo real basado en tecnología RFID. Para realizarlo, se han empleado programas y dispositivos ampliamente utilizados y conocidos como son Excel y Arduino.

Se pretende mejorar el desempeño de un almacén mediante esta tecnología con un coste reducido. Esta digitalización de la información contribuirá a la mejora de los procesos realizados en un almacén, reduciendo los tiempos de gestión, reduciendo errores a la hora de registrar la información de los productos, facilitando la labor de los empleados que operan en él, reduciendo la cantidad de documentos físicos empleados y ofreciendo información del sistema actualizada y centralizada en un solo ordenador.

1.2. Objetivos

El principal objetivo de este trabajo es crear un sistema que mediante el uso de la tecnología RFID permita registrar, almacenar y presentar la información relativa a los palés que gestiona un almacén. Después, los datos deben ser procesados para ofrecer información que facilite y mejore el desempeño del almacén. Esto debe realizarse en tiempo real y con un bajo coste. A



continuación, para asegurar la consecución del objetivo principal, se establecen los siguientes objetivos secundarios:

1. Obtener el conocimiento necesario y establecer las bases sobre la tecnología RFID y sobre los almacenes para desarrollar un sistema de gestión aplicable a un almacén real.
2. Adquirir las habilidades y competencias necesarias en programación VBA y en el lenguaje Arduino para cumplir con los objetivos marcados en este apartado.
3. Desarrollar de un prototipo basado en Arduino que permita la lectura y escritura de tarjetas o etiquetas RFID desde un ordenador. Es importante que el prototipo sea de bajo coste. Este objetivo busca digitalizar el almacén para conocer en tiempo real los movimientos que se realizan y para reducir la cantidad de papel empleado para la gestión de los movimientos, lo cual hace que el proceso sea más fácil y simple.
4. Desarrollar un sistema que registre la fecha y hora real de los movimientos de entrada y salida de los productos en el almacén, junto con la información de las etiquetas. Estos datos deben conservarse aunque se desconecte el ordenador del microcontrolador. Además, se deben poder planificar dichos movimientos.
5. Programar un libro de Excel capaz de indicar cuáles son los productos que deben salir del almacén y en qué ubicaciones se deben situar los palés entrantes en base al método FIFO o LIFO. De este modo, se automatiza la toma de decisiones a la hora de ubicar los palés entrantes o de obtener los de salida evitando errores humanos.
6. Indicar de manera clara y sencilla a los empleados la ubicación que le corresponde a un palé que llega al almacén. Esto pretende reducir errores y mejorar la fluidez de almacenado al disminuir el tiempo que supone encontrar una ubicación disponible.
7. Extraer a partir de la información recopilada varios indicadores relevantes para la evaluación del desempeño del almacén y que cubran diferentes aspectos del mismo para poder señalar áreas de mejora.



1.3. Estructura del trabajo

Este trabajo se estructura del siguiente modo: en el primer capítulo se localizan la introducción, los objetivos y la estructura del trabajo. En el segundo capítulo se realiza el estado del arte tanto de la tecnología RFID como de los almacenes, estableciendo las bases para después indagar en las innovaciones más recientes.

Posteriormente, en el tercer capítulo se mostrarán todos los componentes empleados para la elaboración del prototipo, proporcionando tablas con las especificaciones técnicas de los dispositivos y la explicación del funcionamiento de cada uno de ellos. Este capítulo concluye con la presentación del prototipo final, que consiste en una imagen del mismo con sus elementos enumerados y con su correspondiente diagrama de conexiones.

A continuación, en el cuarto capítulo se profundiza en la programación del dispositivo Arduino y de la aplicación de Excel. Además, en este capítulo se localiza la guía de uso del sistema de control de almacenes desarrollado en este trabajo. La explicación detallada del funcionamiento de los programas, tanto de Arduino como de Excel, viene acompañada de diagramas de flujo que pretenden facilitar su comprensión.

El quinto capítulo presenta los resultados de este trabajo, ofreciendo un ejemplo sencillo del funcionamiento de este sistema. En este capítulo se realizarán procesos como la planificación, recepción y expedición de productos y la evaluación del desempeño del almacén tomado como ejemplo.

El capítulo sexto recoge el estudio económico donde se calcularán los costes que resultan de la realización de este trabajo, el cual se puede dividir en cinco actividades a completar que son: la redacción de la memoria, la recopilación de la información, la programación del microcontrolador Arduino, la programación de la aplicación de Excel y el diseño y montaje del circuito.

Finalmente, el séptimo capítulo está dividido en conclusiones y en líneas de mejora.



Desarrollo de un sistema de control de inventario en tiempo real mediante tecnología RFID





CAPÍTULO 2. ESTADO DEL ARTE



Desarrollo de un sistema de control de inventario en tiempo real mediante tecnología RFID





Este capítulo está dividido en dos secciones. La primera, trata sobre la identificación por radio frecuencia (a partir de ahora, RFID) y la segunda sobre almacenes y su gestión.

2.1. RFID

En este apartado se establecerán las bases del estado del arte con la descripción y clasificación de los dispositivos RFID más comunes y relevantes y se finalizará con la exposición de las tecnologías, aplicaciones, tendencias y retos más recientes.

2.1.1. Introducción

La tecnología RFID identifica objetos, personas o animales y sus características a distancia, mediante ondas electromagnéticas. Un sistema RFID consta principalmente de una tarjeta o etiqueta que contiene la información y un lector que extrae dichos datos. Sus aplicaciones son variadas, como la logística, seguridad, medicina, automovilística e incluso de monitorización de la salud estructural de edificios (Costa et al., 2021). En particular, para la logística de almacenes y seguimiento de productos, es una tecnología que se encuentra en expansión, debido a que es una alternativa superior, en muchos aspectos, a los códigos de barras.

La tecnología RFID se clasifica dentro de las Redes Inalámbricas de Área Personal (WPANs, ver Figura 1), que conectan diversos equipos electrónicos a una distancia de hasta 10 metros (González y Villapol, 2013).

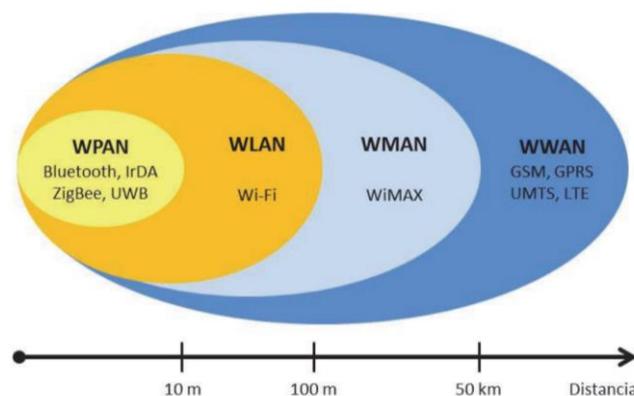


Figura 1. Diferentes redes inalámbricas en función de su cobertura

Nota. Adaptado de Clasificación de las redes Inalámbricas [Fotografía], por J. Salazar, Techpedia (https://upcommons.upc.edu/bitstream/handle/2117/100918/LM01_R_ES.pdf).



2.1.2. Comparación de la tecnología RFID con los códigos de barras para la automatización de almacenes

Según Wyld (2006) la tecnología RFID presenta numerosas ventajas en comparación con los códigos de barras, las principales son las siguientes. En primer lugar, la información se puede leer aunque no exista una línea de visión directa entre el lector y la etiqueta. Además, un lector RFID puede leer múltiples etiquetas simultáneamente y a distancias mayores. También, almacenan una cantidad mayor de información, pudiendo identificar a un producto concreto y sus características, en lugar de solo su tipo. Las etiquetas RFID se pueden actualizar y son más duraderas pudiendo trabajar en entornos hostiles, mientras que los códigos de barras no pueden ser leídos si la superficie está dañada o contaminada. Por último, las etiquetas RFID pueden ser leídas de manera automática, lo cual reduce errores y tiempos de escaneo e implica la posibilidad de realizar un seguimiento en tiempo real.

Si bien es cierto que debido a complicaciones como que los materiales metálicos o los líquidos pueden interferir en la señal o que para ciertas compañías la implantación de un nuevo sistema resulta una inversión elevada o poco rentable, puede ser una opción preferible mantener el sistema de seguimiento de productos con código de barras o incluso desarrollar un sistema mixto que emplee ambas tecnologías (White et al. 2007). Sin embargo, la tecnología RFID ofrece más ventajas y funcionalidades por lo que resulta preferible para numerosas empresas frente a los códigos de barras.

2.1.3. Elementos del sistema

Cualquier sistema RFID consta de dos elementos principalmente, que son el lector y el transpondedor (ver Figura 2). En primer lugar, todos los lectores comparten las siguientes características:

El lector generalmente contiene un módulo de radiofrecuencia, una unidad de control y un elemento de acoplamiento con el transpondedor. Además, muchos lectores están equipados con una interfaz adicional para permitirles enviar la información recibida a otro sistema (Como puede ser un PC o un robot). (Finkenzeller, 2010, p. 9)

Por otro lado, la estructura básica y más común de los transpondedores consiste en una bobina que actúa como una antena, para recibir y emitir la información, un chip, que modula la señal del lector y contiene los datos y una carcasa.

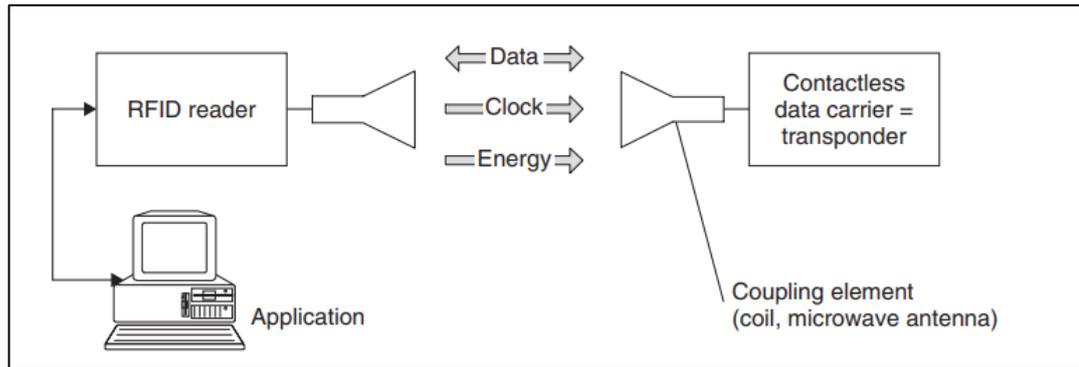


Figura 2. Esquema básico de un sistema RFID

Nota. Adaptado de *RFID Handbook* (p. 8), por K. Finkenzeller, 2010, Wiley.

2.1.3.1. Lector

Según Finkenzeller (2010), “los lectores de todos los sistemas se pueden reducir a dos bloques funcionales fundamentales: la interfaz de radiofrecuencia y la unidad de control” (p. 318). Además, existen otros componentes como la interfaz de comunicación, la fuente de alimentación, y la antena (elemento de acoplamiento con el transpondedor). Luego, el lector puede estar conectado con una aplicación software que tiene el propósito de gestionar el sistema RFID, todos estos elementos se muestran en la Figura 3. Esta gestión suele implicar el procesamiento de los datos recopilados, incluir una interfaz para el usuario o emplear dichos datos para automatizar un sistema.

La interfaz o módulo de radiofrecuencia se encarga de proporcionar la energía necesaria al transpondedor para poder funcionar. También, cumple la función de comunicación, al modular la información para enviarla al transpondedor y demodular la que recibe del mismo. La modulación consiste en convertir una señal digital en una señal analógica que puede ser enviada por una antena y recibida por otra. Es decir, es la conversión de datos digitales en una señal de radio. La demodulación es el proceso inverso.

La unidad de control, se encarga de ejecutar los comandos que provienen de la aplicación software y controlar la comunicación con el transpondedor. Además, el lector puede ofrecer funciones adicionales si es capaz de comunicarse con varios transpondedores al mismo tiempo o interactuar con tarjetas encriptadas. Esta unidad de control está compuesta por un microprocesador, una memoria RAM y otra ROM y si ofrece funcionalidades de encriptación es común encontrar un módulo especializado en cálculos criptográficos.

La interfaz de comunicación puede ser un bus como el Serial UART, SPI o I2C. Estos permiten la transferencia de datos en ambos sentidos entre el lector RFID y la aplicación.

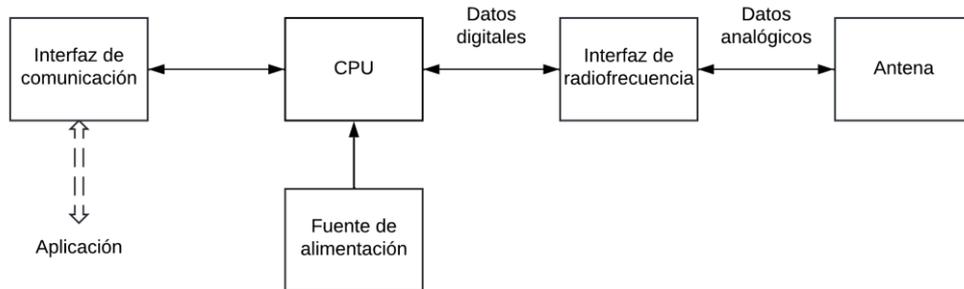


Figura 3. Componentes de un lector RFID

2.1.3.2. Transpondedor

Como se ha mencionado previamente, un transpondedor o etiqueta, consta de una antena, un chip y de la carcasa que los recubre. La antena del transpondedor recibe la señal, que es transmitida al chip. El chip contiene 3 elementos principales que son la memoria, la CPU y la interfaz que conecta la antena con el resto del chip (ver Figura 4).

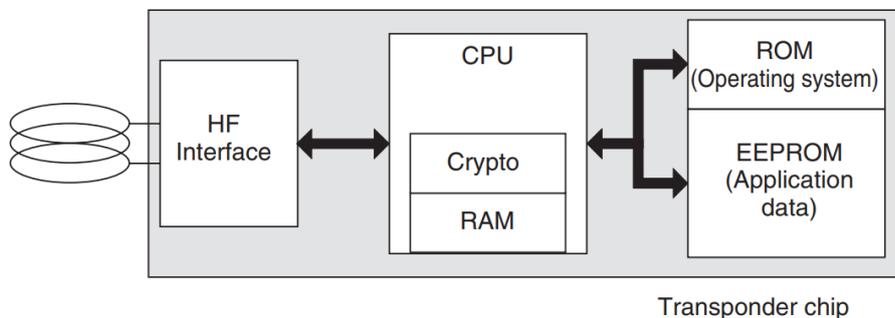


Figura 4. Componentes del transpondedor

Nota. Adaptado de *RFID Handbook* (p. 301), por K. Finkenzeller, 2010, Wiley.

En primer lugar, según Finkenzeller (2010), la interfaz demodula la señal del lector que se recibe en la antena. Es decir, a partir de la señal recibida transmite datos digitales a la CPU. Además, mediante un modulador de carga, envía de vuelta al lector la información suministrada por la CPU.

En segundo lugar, los transpondedores pueden integrar diferentes tipos de memoria. Los más sencillos contendrán una memoria como puede ser una PROM para almacenar el número de serie y sobre la cual, el lector no podrá escribir. Después, aquellos que permiten la lectura y la escritura pueden



incorporar una SRAM (de hasta 64 Kbytes) o una EEPROM (que almacena los datos de la aplicación), además de una ROM donde se almacenará el sistema operativo. Si presentan funcionalidades de criptografía, contendrán en la memoria las claves para permitir el acceso a los contenidos. Las claves evitan los accesos no autorizados tanto para lecturas como para escrituras y se emplean en llaves de vehículos o tarjetas de acceso. Tarjetas RFID como las MIFARE, que se emplean en este trabajo, constan de una memoria segmentada. Es decir, la memoria está dividida en secciones y para acceder a cada una de ellas será necesario introducir una clave, de tal forma que en una misma tarjeta se pueden almacenar datos para diferentes aplicaciones.

Para almacenar los datos escritos en el transpondedor de manera permanente es necesario emplear una memoria no volátil, como puede ser una EEPROM o una FRAM, ya que una SRAM perderá toda la información cuando se interrumpa el suministro de energía. Si comparamos los rendimientos de las FRAM con las EEPROM, comprobamos que las FRAM ofrecen prestaciones superiores con una mayor densidad de integración, mayor cantidad de ciclos de escritura, un consumo energético inferior y también, un tiempo de escritura inferior (ver Tabla 1).

Característica	FRAM	EEPROM
Tamaño de la celda de memoria (μm^2)	80	130
Ciclos de escritura soportados	10^{12}	10^5
Voltaje de escritura (V)	2	12
Energía de escritura (μJ)	0,0001	100
Tiempo de escritura (μs)	0,1	10.000

Tabla 1. Comparación de las memorias FRAM con las EEPROM

Nota. Adaptado de RFID Handbook (p. 311), por K. Finkenzeller, 2010, Wiley.

De todos modos, la tecnología FRAM no tiene un uso tan extendido a pesar de ofrecer mejores prestaciones que las memorias EEPROM, ya que, según Finkenzeller (2010) “existen problemas al combinar los procesadores CMOS y los circuitos analógicos (interfaz de radiofrecuencia) con celdas FRAM en un solo chip” (p. 311).

En tercer lugar, los transpondedores, también incorporan un microprocesador que suele estar acompañado de una unidad criptológica para realizar los cálculos para los procedimientos de encriptación.

Por otra parte, los transpondedores se pueden clasificar en función de 3 criterios. Según su alimentación, según el almacenamiento de la información y según el formato de construcción (ver Figura 5).

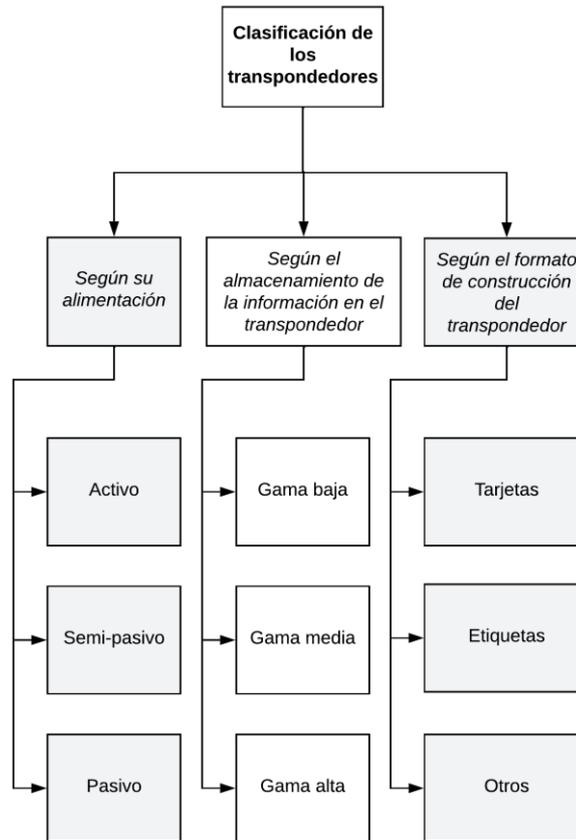


Figura 5. Clasificación de los transpondedores

A continuación, se explican en detalle cada una de las clasificaciones.

2.1.3.2.1. Según su alimentación

Los transpondedores pueden ser activos, semipasivos o pasivos. Los activos cuentan con una batería que provee al transpondedor la energía necesaria para comunicarse a largas distancias. Además, son los únicos que son capaces de establecer la comunicación con el lector por iniciativa propia y por eso, esta tecnología se emplea en aplicaciones que requieran información en tiempo real. Sin embargo, estos dispositivos suponen un mayor coste y tienen un ciclo de vida limitado (Khattab et al., 2017).

Los semipasivos también cuentan con una batería pero esta solamente alimenta al chip. Por este motivo, necesitan la energía del lector para alimentar el transmisor de señal.



Finalmente, los pasivos requieren que el lector suministre energía para poder funcionar. Son los más comunes en el mercado, suponen un coste menor y su ciclo de vida es mayor, pero su distancia de comunicación se reduce y presentan limitaciones de computación y de consumo de energía.

2.1.3.2.2. Según el almacenamiento de la información en el transpondedor

Según Finkenzerler (2010), dependiendo de la gama del sistema, el transpondedor será capaz de realizar más o menos funciones. Entre los dispositivos de gama baja, encontramos aquellos que solamente devuelven un 1 o un 0 lógico en función de si se encuentran dentro de la zona de interrogación o no. Este tipo de sistemas se suele emplear para la seguridad en los comercios. Además, en la gama baja también encontraríamos dispositivos de solo lectura. A continuación, en la gama media, se pueden encontrar más funciones como leer y escribir en el transpondedor, memorias con mayor capacidad, transpondedores activos y pasivos. Y finalmente, la gama alta ofrece tarjetas inteligentes y funciones de criptografía.

2.1.3.2.3. Según su el formato de construcción del transpondedor

El transpondedor se puede integrar en diferentes formatos como pueden ser tarjetas, etiquetas, monedas, llaveros y relojes. El material que se emplea para el recubrimiento suele ser plástico ver Figura 6 y Figura 28, aunque también se construyen con carcasas de cristal.

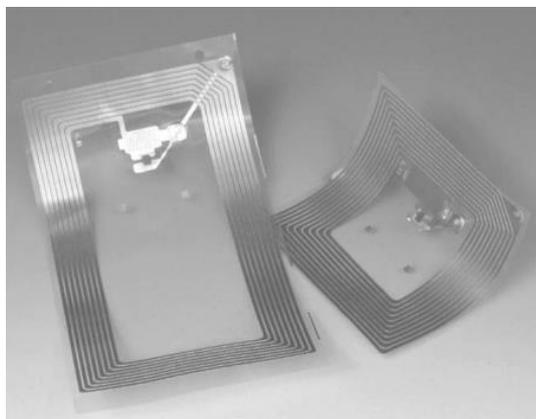


Figura 6. Dos etiquetas RFID

Nota. Adaptado de RFID Handbook (p. 20), por K. Finkenzerler, 2010, Wiley.

2.1.4. Tipos de tecnología RFID

Según el procedimiento de comunicación, frecuencia, alcance y propiedad física de acoplamiento, podemos clasificar a los diferentes sistemas RFID (Figura 7).

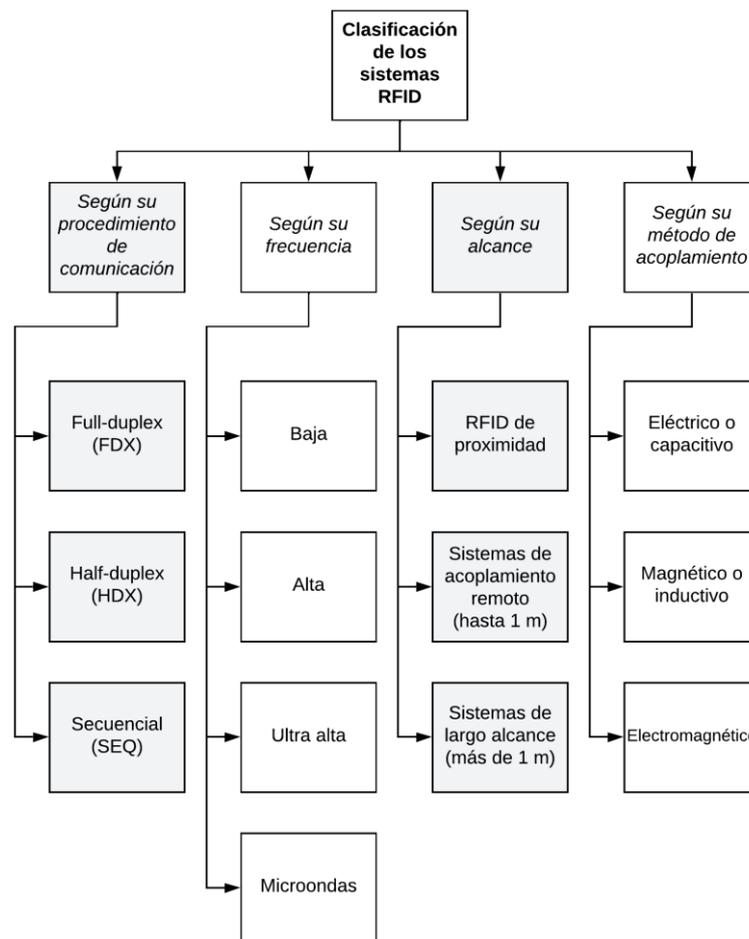


Figura 7. Clasificación de los sistemas RFID

2.1.4.1. Según su procedimiento de comunicación

Cuando un transpondedor entra en la zona de interrogación del lector, se inicia la comunicación entre ambos y podrá seguir diferentes modos de comunicación. En primer lugar, en el procedimiento “full-duplex” (FDX) el transpondedor envía datos al lector “up-link” y los recibe del mismo simultáneamente “down-link”. Después, estaría el “half-duplex” (HDX), donde el transpondedor y el lector se alternan para enviar información. En ambos casos, el lector suministra energía de manera continuada. Por último, la comunicación puede ser secuencial, este caso es similar al “half-duplex” y se diferencia en que durante las transmisiones de información al lector desde el transpondedor (“up-link”), el lector corta la alimentación (Figura 8).

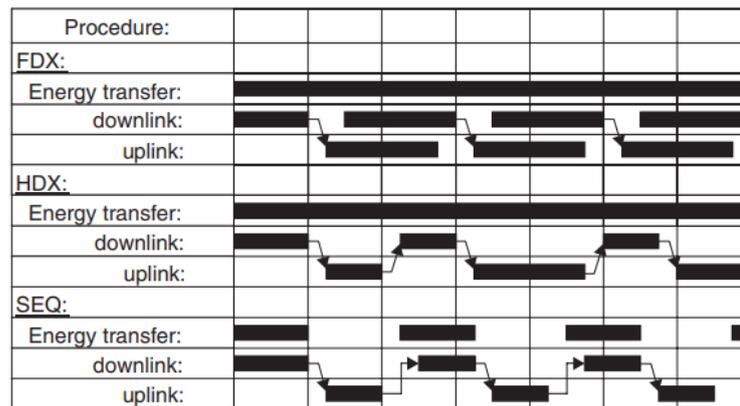


Figura 8. Modos de comunicación

Nota. Adaptado de *RFID Handbook* (p. 40), por K. Finkenzeller, 2010, Wiley.

2.1.4.2. Según su frecuencia

Generalmente, los dispositivos trabajan en un rango de frecuencias de entre 135 kHz y 5.8 GHz. Las frecuencias bajas de 135 kHz y altas de 13,56 MHz se emplean para etiquetas y tarjetas RFID de proximidad. Por otra parte, las frecuencias ultra-altas (UHF) se sitúan en los 868 MHz en Europa y las microondas en los 2,5 GHz o 5,8 GHz, estas frecuencias permiten alcances mayores especialmente si se usan transpondedores activos. En las frecuencias ultra-altas y microondas operan los dispositivos de largo alcance (superior a un metro) (Khattab et al., 2017).

Después, la frecuencia de los dispositivos también está relacionada con el método de acoplamiento (ver Tabla 2). Los sistemas que se acoplan capacitivamente o inductivamente funcionan con frecuencias entre 100 kHz y 30 MHz y los sistemas electromagnéticos funcionan con frecuencias ultra-altas y microondas. Los sistemas que operan con frecuencias más bajas, atraviesan mejor los objetos, mientras que aquellos que funcionan con microondas, soportan mejor las interferencias electromagnéticas, por lo que son los más indicados en industrias como la automovilística.

Frecuencia	Hz	Método de acoplamiento	Distancia
Baja	135 kHz	Inductivo o capacitivo	Corta
Alta	13,56 MHz	Inductivo o capacitivo	Corta
Baja	135 kHz	Inductivo	Hasta 1 metro
Alta	13,56 MHz	Inductivo	Hasta 1 metro
Ultra alta	868 MHz	Electromagnético	Larga
Microondas	2,5 GHz	Electromagnético	Larga
Microondas	5,8 GHz	Electromagnético	Larga

Tabla 2. Frecuencias en función del método de acoplamiento y distancia de lectura

2.1.4.3. Según su alcance

El rango que se puede conseguir es desde unos pocos milímetros, hasta 15 metros. Según Finkenzerler (2010), la selección del alcance para un sistema RFID depende de las características de la aplicación. Si la distancia entre los transpondedores que se leen consecutivamente es escasa, como puede ser, la lectura de los autobuses, la distancia de lectura deberá ser corta para evitar lecturas involuntarias.

Por otro lado, se tendrá que tener en cuenta el tiempo que permanece el transpondedor en la zona de interrogación, que deberá cubrir una región mayor si las etiquetas la atraviesan velozmente.

Finalmente, en función de la precisión, se podrá elegir un lector direccional ofrecerá un alcance mayor para aplicaciones más precisas como se indica en la Figura 9.

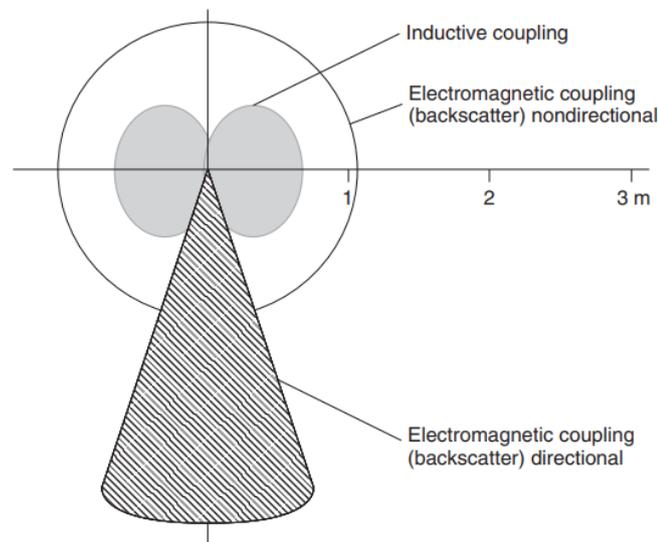


Figura 9. Alcance de diferentes sistemas RFID

Nota. Adaptado de *RFID Handbook* (p. 27), por K. Finkenzerler, 2010, Wiley.

2.1.4.4. Según su método de acoplamiento

Existen tres tipos de acoplamiento: inductivo o magnético, capacitivo o eléctrico y electromagnético (Finkenzerler, 2010).

- **Acoplamiento inductivo.** La mayoría de etiquetas que utilizan este método de acoplamiento son pasivas, por lo que el lector debe proveer a la etiqueta de energía. Esto se consigue haciendo pasar una corriente a través de la bobina del lector, lo cual generará un campo magnético. Este campo se alterna, obteniendo así una frecuencia. Finalmente, este campo magnético,

por inducción mutua, generará una corriente inducida en la bobina de la etiqueta o transpondedor (ver Figura 10).

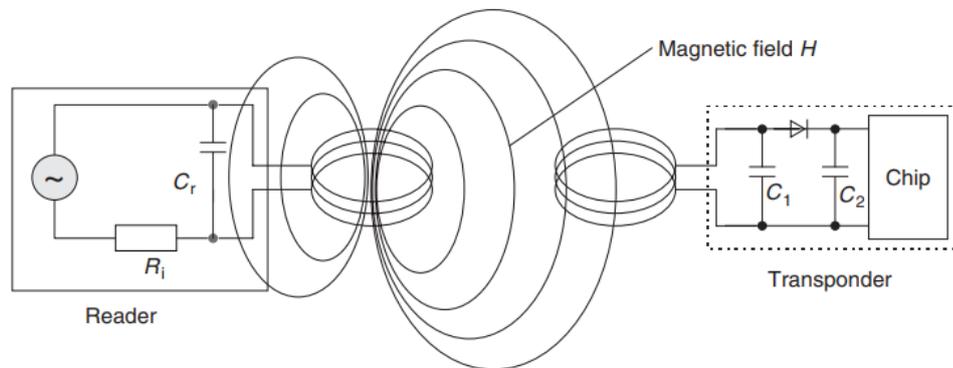


Figura 10. Acoplamiento inductivo

Nota. Adaptado de *RFID Handbook* (p. 41), por K. Finkenzeller, 2010, Wiley.

A continuación, una vez que el chip del transpondedor está provisto de energía para funcionar, podrá transmitir sus datos al lector mediante la *modulación de carga*. El funcionamiento es el siguiente: el transpondedor consta de una resistencia de carga, que se puede activar o desactivar, esto provocará una variación de la impedancia en la bobina del lector, lo cual alterará su voltaje.

- Acoplamiento eléctrico. El lector es una superficie conductora a través de la cual se hace pasar un voltaje elevado de entre centenas y miles de voltios. Junto con un potencial nulo, se crea un campo eléctrico. El transpondedor capacitivo funciona con una corriente que se genera al atravesar dicho campo (ver Figura 11).

Finalmente, con una resistencia de modulación en el transpondedor, que se conecta y desconecta al circuito en función de la información a transmitir, se puede variar el voltaje que circula a través del lector. De este modo, midiendo estas variaciones de voltaje se puede leer la información.

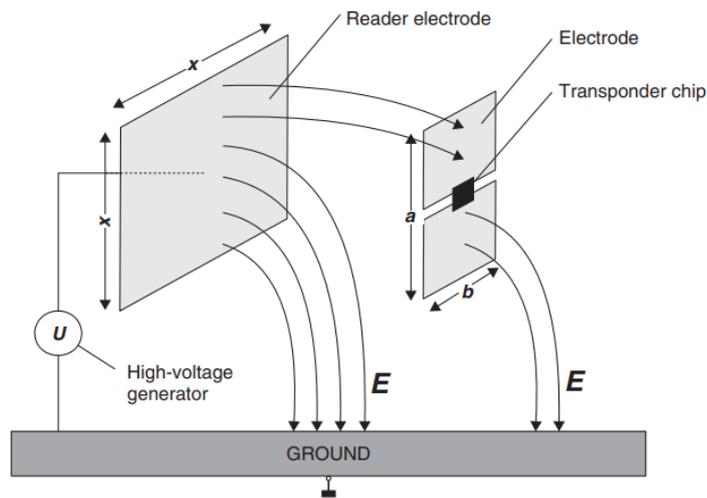


Figura 11. Acoplamiento eléctrico

Nota. Adaptado de *RFID Handbook* (p. 51), por K. Finkenzeller, 2010, Wiley.

Acoplamiento electromagnético por retrodispersión. Los sistemas de largo alcance funcionan con este método de acoplamiento. Estos, obtienen la energía necesaria para su funcionamiento a partir de la señal electromagnética en muchas ocasiones. Cuanta más energía requiera el chip del transpondedor, más se limitará la distancia máxima de funcionamiento.

Después, la antena del transpondedor refleja de vuelta la señal al lector, pero con una resistencia que se puede conectar o desconectar, varía la potencia con la que se emite esa señal, y de este modo se puede transmitir la información al lector (retrodispersión modulada, ver Figura 12).

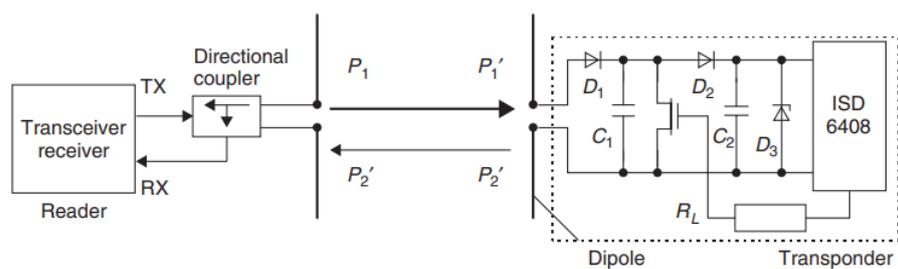


Figura 12. Acoplamiento electromagnético

Nota. Adaptado de *RFID Handbook* (p. 48), por K. Finkenzeller, 2010, Wiley.



2.1.5. Aspectos de seguridad

La tecnología RFID puede sufrir una serie de ataques que se suelen realizar sobre el transpondedor o sobre la señal de radiofrecuencia del sistema RFID. A continuación se presentan los ataques más comunes.

Ataques al transpondedor. En estos casos, el atacante necesita tener el transpondedor o etiqueta a su alcance.

- Destrucción de la etiqueta RFID. Hay varios escenarios en los que se podría dar esta situación. Por ejemplo, el caso más sencillo sería cortar la etiqueta, aplastarla o destruirla químicamente. Otro caso sería introducir la etiqueta en un campo con una elevada intensidad, lo cual provocaría la destrucción térmica del transpondedor.
- Protección del transpondedor. Este riesgo consiste en bloquear la señal del lector mediante un objeto metálico como puede ser el papel de aluminio. Esto ocasiona que las ondas emitidas por el lector reboten en la superficie metálica y no alcancen la etiqueta.
- Clonación y manipulación de transpondedores. Los transpondedores de solo lectura, se limitan a emitir un código que los identifica cuando son introducidos en la zona de interrogación de un lector. Por lo tanto, un atacante podría leer dicho identificador y grabarlo en una memoria de un transpondedor similar, de tal manera que el lector no sería capaz de distinguir el original de la copia.
Por otro lado, si el transpondedor es de lectura y escritura, puede contener secciones en su memoria que no cuenten con ningún tipo de encriptación o que empleen métodos de cifrado inseguros, lo cual permitiría que el atacante modificase la información o realizase una copia de la misma.

Ataques a la interfaz. La interfaz que conecta la etiqueta y el lector es la señal electromagnética, que también es susceptible de ser atacada. En ocasiones, puede resultar un objetivo más atractivo debido a que el ataque se puede llevar a cabo a distancia. Los ataques pueden ser los siguientes.



- Interceptación de la comunicación. “La interceptación de la comunicación entre el lector y el transpondedor es una de las amenazas más prominentes para la tecnología RFID” (Finkenzeller, 2010, p. 217). Consiste en leer el intercambio de datos que se produce entre el lector y el transpondedor. Dicho ataque se puede realizar con un dispositivo situado a distancias mayores de la zona de interrogación del lector. Si el lector es inductivo, se podrá leer su información hasta una distancia de 3 metros, mientras que si es electromagnético, la distancia de lectura podría llegar a varios cientos de metros. Por otro lado, los transpondedores emiten señales más débiles, por lo que las distancias hasta las que estos pueden ser leídos se reducen.
- Bloqueo de señal. Se emite una señal de radio con la suficiente intensidad como para que la comunicación entre el lector y el transpondedor no sea posible. Generalmente, se ataca al “up-link” de la señal (que es el enlace de comunicación de datos del transpondedor al lector), ya que la señal que emite el transpondedor es menos potente que la del lector y por lo tanto resulta más sencillo causar interferencias.
- Extensión del rango de lectura. Este tipo de ataque permite leer el transpondedor desde una distancia mayor a la habitual para reducir la probabilidad de que el atacante sea descubierto. Se aumenta la potencia con la que el lector emite la señal para que la modulación del transpondedor sea percibida en un radio mayor. Además, se emplea en combinación con otros ataques como la interceptación de la señal o clonación.
- Ataque de denegación de servicio mediante etiquetas de bloqueo. Existen ciertos lectores que son capaces de distinguir y comunicarse con diferentes transpondedores situados simultáneamente en la zona de interrogación. Mediante una tarjeta o etiqueta de bloqueo se hace creer al lector que hay numerosos transpondedores cerca lo cual bloquea su funcionamiento.
- Ataque de Relé. Consiste en simular la presencia de un transpondedor en la zona de interrogación del lector. Mediante el uso de dos dispositivos llamados proxy y topo, se puede utilizar un transpondedor que se encuentre a una distancia lejana del lector. El proxy es el dispositivo que simula ser el transpondedor original y que se sitúa frente al lector. Por otro lado, el topo



se sitúa junto al transpondedor auténtico y simula ser el lector. Entonces, el topo realiza la captura de información de la etiqueta o tarjeta en cuestión y la retransmite al lector a través del proxy. En ciertas ocasiones, el lector emitirá información de vuelta por lo que el proceso ocurrirá en el sentido opuesto. De este modo se puede realizar la comunicación a grandes distancias. Este tipo de ataques se caracterizan por su apariencia de legitimidad, lo cual dificulta su detección.

2.1.6. Innovaciones en tecnología RFID

Además de los diferentes tipos de tecnología que se han expuesto en el apartado anterior, existen alternativas más modernas que permiten reducir ciertos inconvenientes. El precio, el consumo de batería, el tamaño, la distancia máxima de lectura, por nombrar algunos de ellos, son obstáculos que se pretenden superar con los siguientes avances.

2.1.6.1. RFID armónico

El funcionamiento de esta tecnología consiste en que el transpondedor envía la información al lector en una frecuencia armónica, que es un múltiplo entero de la frecuencia original emitida por el lector. La popularidad de esta tecnología se ha incrementado en los últimos años y presenta las siguientes ventajas e inconvenientes (Mondal et al., 2021).

Se pretende mejorar la lectura de las etiquetas RFID en ambientes con ruido de fondo e incrementar la distancia máxima de lectura. Además, debido a la naturaleza de este tipo de dispositivos, no requieren de sistemas de seguridad adicionales para evitar su clonación, ya que cada dispositivo tiene una huella digital única. Sin embargo, los retos a los que se enfrenta esta tecnología son la miniaturización y las bandas de frecuencias disponibles.

2.1.6.2. RFID sin chip

Los transpondedores clásicos constan de una antena junto con un chip. La antena era la encargada de reflejar de vuelta la señal al lector y el chip, codificaba la información. Ahora, las antenas de estas etiquetas que no cuentan con un chip, tendrán que cumplir ambas funciones. Una reducción de costes debido a la simplicidad de fabricación, una tecnología más sostenible y duradera son las ventajas que presentan las etiquetas sin chip. Sin embargo, su lectura es más complicada (Costa et al., 2021).



Una de las principales aplicaciones es el uso de estas etiquetas como sensores de temperatura, humedad o pH, que en función del parámetro a medir, variarán la frecuencia que reflejan al lector.

2.1.6.3. Chips auto-calibrados

Los chips auto-calibrados en los transpondedores electromagnéticos tienen una especial utilidad como sensores. En función del medio en el que se encuentre el sensor, la impedancia en la antena variará respecto a la impedancia del chip, por lo que, mediante condensadores que se pueden conectar y desconectar del circuito, estas impedancias se igualan. Entonces, sabiendo el número de condensadores conectados, se obtendrá información sobre el medio en el que se encuentra el chip (Costa et al., 2021).

2.1.6.4. Reducción del tamaño de los transpondedores

Con la evolución de esta tecnología se consigue reducir el tamaño de los transpondedores, como es el caso de los dispositivos antirrobo de Inditex. Según García (2023) Inditex planea llevar a cabo una reducción masiva de sus alarmas rígidas de plástico, por chips de pequeño tamaño para finales de 2024, con el propósito de mejorar la experiencia de sus clientes, al no tener que separar las alarmas de las prendas, reduciendo así el tiempo de pago.

2.2. Almacenes

De un modo paralelo al estado del arte del RFID, se comenzará con la exposición de los conceptos básicos y se concluirá este apartado con las innovaciones más relevantes en este campo.

2.2.1. Introducción

Los almacenes son elementos fundamentales para la fabricación y distribución de productos de una empresa. Su necesidad se presenta ante las diferencias entre la llegada de aprovisionamientos y el ritmo de producción. Además, a la hora de distribuir los productos finales resulta necesario para adaptarse a las variaciones de la demanda y de la producción, para almacenar aquellos productos que no vayan a ser consumidos inmediatamente después de su fabricación y para suministrar artículos a regiones distantes de la fábrica (Arbones Malisani, 1990).



2.2.2. Tipos de almacenes

2.2.2.1. Centros de distribución

Se encargan de almacenar artículos para su posterior entrega a un consumidor. Según Bartholdi y Hackman (2019), existen los siguientes tipos de almacenes en base a los clientes que demandan sus productos:

- Centro de distribución minorista. Son los almacenes que suministran los productos a negocios de venta minorista. Generalmente, cada pedido puede suponer la entrega de cientos o miles de productos y se suelen realizar en franjas temporales regulares.
- Centro de distribución de piezas de servicio. Se caracterizan por distribuir un volumen elevado de componentes, ya sea, para automóviles, maquinaria o equipamiento médico. Además, suele haber miles de componentes diferentes por lo que predecir la demanda de cada uno de ellos resulta una tarea complicada. Para protegerse ante rupturas de stock, se mantienen stocks de seguridad relativamente elevados de cada pieza. Es por esto por lo que este tipo de centros de distribución suponen una gran inversión. Por otro lado, los pedidos pueden ser regulares, que son grandes y predecibles, pero también de emergencia que son en general pequeños e impredecibles.
- Centro de distribución de comercio electrónico. Son centros que reciben pedidos de pequeño tamaño como norma general, inferiores a 4 unidades. Sin embargo, la cantidad de pedidos y las exigencias de tiempos de entrega son elevadas.
- Almacén de logística de terceros (3PL). Se encargan de realizar las funciones de almacenaje de otra empresa, ya sea como servicio subcontratado o como recurso de emergencia en caso de un exceso de inventario. Estos almacenes aprovechan las economías de escala (reducción de los costes unitarios al almacenar más unidades en el mismo espacio) almacenando en sus instalaciones productos de diferentes compañías.
- Almacén de perecederos. Para productos que requieren condiciones de temperatura y humedad específicos para la conservación de sus productos. Los productos permanecen en el almacén durante periodos cortos de tiempo. Además, debido al coste de la refrigeración, el espacio disponible resulta relativamente escaso. Es frecuente el uso del sistema FIFO (First-In-First-Out) para enviar primero los productos con la fecha de caducidad más



cercana. Además, están sujetos a mayores restricciones de gestión, especialmente aquellos pertenecientes a la industria alimentaria.

2.2.2.2. Almacenes industriales:

Según Arbones Malisani (1990), “comprenden el conjunto de almacenes de una industria para almacenar las materias primas y productos terminados” (p. 27). Además, también incluyen a aquellos destinados al almacenaje de productos semielaborados, piezas (bien sean de recambio o para conformar el producto final), herramientas y utillajes y aprovisionamientos generales como pueden ser combustibles, lubricantes o embalajes.

2.2.2.3. Depósitos:

Es un lugar preparado para el almacenaje de mercaderías en base a un acuerdo entre el depositante y el depositario.

2.2.3. Gestión de almacenes

2.2.3.1. Funciones básicas

- Recepción de los productos. El proceso comienza con el envío por parte de los proveedores de una notificación con la fecha en la que se van a entregar las mercancías y con los detalles del pedido. Cuando el pedido llega al almacén, se debe aceptar comparando la notificación con el albarán de entrega y con el pedido que se realizó (Hompel y Schmidt, 2007).

A continuación, se recogen las mercancías en una terminal con el suficiente espacio disponible y se procede a realizar la inspección que incluye el conteo, verificación de los tipos de productos, pesaje, verificación de las dimensiones, inspecciones visuales y test de laboratorio (si fuese necesario), entre otros. Además, se realizará una inspección específica a aquellos productos que sean devueltos por los clientes.

Una vez realizada la inspección se agrupan las mercancías de tal forma que todos los palés del almacén tengan unas dimensiones estandarizadas, con el propósito de optimizar sus costes de transporte y almacenaje.

- Almacenamiento. En primer lugar, un sistema informático decide el lugar de almacenaje óptimo de las mercancías y se procede a llevar los artículos al lugar designado con la mayor transparencia posible. A continuación, se identifican los productos que no hayan sido identificados en la recepción. Para seleccionar el lugar de almacenaje se tendrán en cuenta parámetros como los requisitos técnicos, de optimización y de seguridad y legales.



- “Picking” o preparación de pedidos. En función de los objetivos que se quieran lograr hay varias estrategias disponibles. FIFO, es apropiada para productos con fechas de caducidad cercanas. LIFO (Last-In-First-Out), para almacenes en los que los productos solamente pueden entrar y salir por un lado como en los de bloque estático. Camino más corto, sería otra estrategia para recuperar los artículos de las estanterías en el menor tiempo posible. Todos los movimientos de las unidades de carga deberán estar sincronizados con el sistema informático.
- Recogida de pedidos. Cada pedido requiere de un tipo de artículos en una cantidad en particular. Para conformar el pedido, será necesario un sistema de recogida de pedidos que estará conformado por tres elementos. Por el sistema de flujo de materiales, para obtener los artículos que conforman el pedido de la manera más eficiente posible, por la organización y por el flujo de información.
- Embalaje. Se eligen los embalajes y cajas adecuados para optimizar el espacio y los costes de envío. En numerosas ocasiones esta selección se realiza manualmente, aunque también existen sistemas de gestión de almacenes que lo pueden calcular.
- Envíos. Se encarga de llevar los pedidos a los medios de transporte necesarios.

2.2.3.2. Software de gestión

Según Bartholdi y Hackman (2019), “Un centro de distribución moderno puede alcanzar 60.000 metros cuadrados de extensión y contener 100.000 unidades de almacenamiento, con cientos de trabajadores recopilando y consolidando miles de pedidos a tiempo para cumplir con los horarios de envío diarios.” (p. 33). Por lo tanto, para gestionar y coordinar de manera eficiente las operaciones diarias de un almacén se emplean programas informáticos conocidos como “Warehouse Management Systems” (WMS). Esencialmente, son bases de datos que ofrecen información en tiempo real del inventario de las unidades de almacenamiento (cantidad de productos) y del inventario de las ubicaciones de almacenamiento (dónde se encuentran los productos). Los programas WMS contienen herramientas para gestionar operaciones básicas como pueden ser: la recepción de productos, almacenamiento, seguimiento, “picking”, embalaje y envíos. También, registran la información sobre cada producto como su localización, tamaño y embalaje.



La actividad principal de un WMS es registrar las entradas y salidas de productos para realizar los pagos a los proveedores y emitir las facturas a los clientes. También, es esencial que el programa sea capaz de conocer las localizaciones de todos los productos y de las posiciones vacantes. De este modo se pueden conocer dónde situar los productos entrantes y realizar el “picking”.

Por otro lado, también se llevan a cabo tareas como recibir todos los pedidos y reorganizarlos en “listas de picking” que enumeran productos del mismo tipo para simplificar su recogida. Otra tarea que llevan a cabo los WMS es realizar el seguimiento del ensamblaje de cada pedido, es decir, monitoriza el proceso de agrupar los productos adquiridos por el cliente en un mismo embalaje.

Gracias a estos sistemas de gestión de almacenes se han podido reducir los tiempos de entrega de manera significativa. Algunas de las marcas que ofrecen WMS más ampliamente utilizados en la industria son SAP, Oracle, Manhattan Associates' y Blue Yonder.

2.2.3.3. Indicadores clave

Según Hompel y Schmidt (2007), con la operación diaria del almacén, se generan datos que se conocen como información básica, que engloba a los datos maestros como los pesos, tamaños, cantidades y descripciones de los productos. Datos de inventario como el stock total, stock de seguridad y el stock medio. También, formarían parte de los datos básicos los datos de movimiento de los productos y otros datos sobre los costes y empleados. A partir de la información básica, se obtienen los indicadores clave (KPI). Algunos de los más comunes se presentan en la Tabla 3.

KPI	Definición
Disponibilidad para la entrega	$\frac{\text{Núm. de pedidos entregados a tiempo}}{\text{Núm. de pedidos totales}}$
Ocupación del almacén	$\frac{\text{Núm. de ubicaciones ocupadas}}{\text{Núm. de ubicaciones totales}}$
Rotación de inventario	$\frac{\text{Coste de los bienes vendidos [€]}}{\text{Valor medio del inventario [€]}}$
Coste por unidad de almacenamiento	$\frac{\text{Coste total [€]}}{\text{Capacidad de almacenamiento [Uds.]}}$
Cobertura de inventario	$\frac{\text{Volumen de existencias [Uds.]}}{\text{Demanda media [Uds./días]}}$

Tabla 3. Ejemplos de KPI para la gestión de un almacén.

La rotación indica las veces que se ha repuesto el inventario en un periodo de tiempo determinado y la cobertura de inventario calcula el periodo de tiempo restante hasta que se agoten las existencias, pueden ser días u otra unidad de tiempo.

2.2.4. Principios de almacenamiento

2.2.4.1. Almacenamiento sobre el suelo

Con este método de almacenamiento los palés o cajas se sitúan uno encima del otro sin necesidad de ninguna estructura adicional. Su sencillez se ve reflejada en los costes que supone y en la flexibilidad a la hora de modificar su disposición en planta, aunque también repercute en la altura máxima que la pila puede alcanzar. Las disposiciones en planta más comunes son en bloque y en línea (ver Figura 13 y Figura 14, respectivamente). En bloque se aprovecha más eficientemente el espacio disponible, pero solamente es útil para los sistemas LIFO en los que se envían primero los productos situados en la única fila accesible. En línea se facilita el acceso a cada palé a costa de aprovechar menos el espacio (Hompel y Schmidt, 2007).

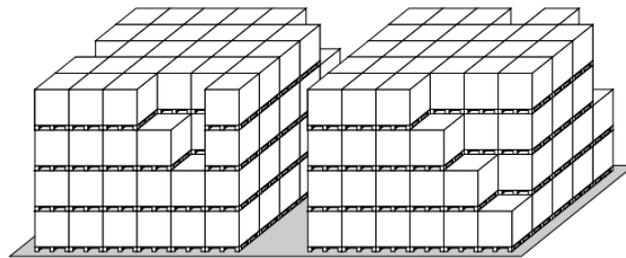


Figura 13. Disposición en bloque

Nota. Adaptado de *Warehouse Management. Automation and Organisation of Warehouse and Order Picking Systems* (p. 93), por Hompel y Schmidt, 2007, Springer.

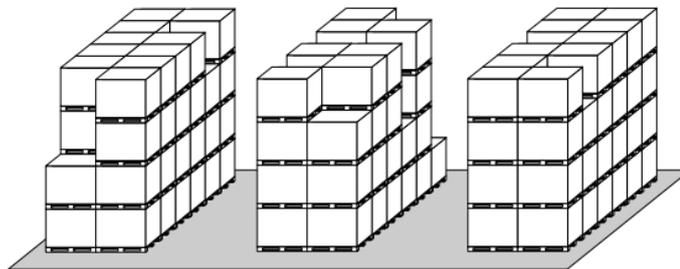


Figura 14. Disposición en línea

Nota. Adaptado de *Warehouse Management. Automation and Organisation of Warehouse and Order Picking Systems* (p. 93), por Hompel y Schmidt, 2007, Springer.

2.2.4.2. Almacenamiento en estantes

- Estantería de palés. A la hora de realizar el almacenamiento, se usan con frecuencia las estanterías de palés, en las que pueden ser colocados de dos maneras distintas. Si se almacenan longitudinalmente (Figura 15), cada palé se sostiene mediante dos vigas metálicas que proporcionan soporte en el lado frontal (Figura 16) y trasero de la unidad de carga. Si se almacenan transversalmente, la capacidad de almacenamiento horizontal se verá reducida debido a que el lado largo de las unidades de carga se situará paralelo a las vigas (Hompel y Schmidt, 2007).



Figura 15. Palé situado longitudinalmente

Nota. Adaptado de *Gente trabajando* [Fotografía], por Tiger Lily, 2020, Pexels (<https://www.pexels.com/es-es/foto/gente-trabajando-hombres-industria-4481532/>).



Lado frontal

Figura 16. Lado frontal del palé

Nota. Adaptado de *euro palet* [Fotografía], por OpenClipart-Vectors, 2013, Pixabay (<https://pixabay.com/es/vectors/euro-palet-paleta-madera-transporte-161949/>).

- Estanterías de contenedores o “mini-load”. Para objetos de pequeño peso y tamaño, puede resultar más eficiente a la hora de aprovechar el espacio



emplear estanterías con contenedores pequeños o cajas (Figura 17). Se suelen emplear junto a dispositivos automatizados (traselevadores, ver página 46) para realizar los movimientos de las mercancías.



Figura 17. Estantería "mini-load" con transelevador automático

Nota. Adaptado de *Warehouse Management. Automation and Organisation of Warehouse and Order Picking Systems* (p. 97), por Hompel y Schmidt, 2007, Springer.

- Estanterías de bloque estático. Se eliminan los pasillos entre las islas de estanterías y se almacenan de tal forma que las unidades de carga pueden entrar y salir por uno o dos lados, en función de si el sistema es FIFO (dos lados) o LIFO (un lado, ver Figura 18).



Figura 18. Estanterías de bloque estático

Nota. Adaptado de *Warehouse Management. Automation and Organisation of Warehouse and Order Picking Systems* (p. 101), por Hompel y Schmidt, 2007, Springer.



2.2.4.3. Sistemas dinámicos

Con el propósito de maximizar el espacio utilizado, se crean sistemas dinámicos que pueden alterar su disposición en planta, desplazando las estanterías para abrir y cerrar pasillos donde se requiera. De este modo, se incluyen más estanterías y menos pasillos, aprovechando mejor el espacio (Figura 19). Sin embargo, estos desplazamientos deben ser realizados a velocidades inferiores a los 5 m/min por lo que se deben de reducir los movimientos de estanterías al mínimo.



Figura 19. Sistema de islas móviles

Nota. Adaptado de Warehouse Management. Automation and Organisation of Warehouse and Order Picking Systems (p. 103), por Hompel y Schmidt, 2007, Springer.

Por otro lado, estarían las estanterías rotatorias que pueden presentar una disposición vertical u horizontal. Estas permiten aumentar la eficiencia del espacio utilizado limitando la entrada y salida de las mercancías a una pequeña sección del dispositivo. Además, reducen el error humano y aumentan la velocidad de recogida.

2.2.5. Principios de transporte

2.2.5.1. Transportadores

En primer lugar, los transportadores de rodillos hacen uso de la gravedad y de motores para mover los bienes, que deberán tener una superficie plana. El diseño es sencillo y versátil, ya que puede adaptarse para bienes ligeros o de



una tonelada. Además, se puede usar como un espacio intermedio de almacenaje, aunque es notablemente ruidoso (Hompel y Schmidt, 2007).

Por otro lado, las cintas transportadoras, desplazan las mercancías con una superficie continua que reduce el ruido, al contrario que los rodillos. Una característica muy relevante es la elevada fricción de la cinta que permite realizar tramos inclinados. Además, esto permite un desplazamiento más suave y controlado. Sin embargo, generalmente solo pueden soportar hasta 50 kg.

2.2.5.2. Vehículos

Los montacargas pueden ser de varios tipos para adaptarse a diferentes escenarios como mover cargas situadas en posiciones elevadas de incluso 12 metros, desplazar al operario junto con la horquilla para la preparación de pedidos o para manipular mercancías largas (Hompel y Schmidt, 2007).

2.2.5.3. Grúas

Las grúas son de gran utilidad para realizar el cambio de método de transporte de mercancías, especialmente si son pesadas. Al contrario que los transportadores, no tienen una trayectoria predefinida para mover las cargas. Existen múltiples tipos como las grúas puente, grúas en suspensión, transelevadores, grúas de columna y grúas pórtico. Según Hompel y Schmidt (2007), estas son sus características.

- Grúas puente. Se sitúan sobre unos rieles en la parte superior de la nave industrial, que soportará todo su peso y puede desplazarse a lo largo de la misma. Además, cuentan con otro grado de libertad al poder desplazar el carro perpendicularmente, para así cubrir toda la superficie.
- Grúas en suspensión. Son similares a las grúas puente con la diferencia que los rieles están sujetos al techo, lo cual incrementa la superficie de trabajo.
- Transelevadores. Sus rieles pueden situarse en el suelo o en el techo y aunque existen modelos que cubren la toda la superficie de la planta, es común ver transelevadores que solamente se pueden desplazar longitudinalmente a lo largo de los pasillos para elevar las cargas. Su propósito radica en transportar las unidades de carga de las estanterías. Los transelevadores pueden asemejarse a las grúas puente con la diferencia de que en lugar de un cable o cadena, la carga se sostiene con unas barras telescópicas. Otros modelos, se suelen construir con una o dos



columnas metálicas a través de las cuales se desplaza un carro de elevación.

- Grúas de columna. Su área de trabajo recorre un volumen cilíndrico, aunque no todas pueden girar 360°. Al estar montada sobre un pilar, su movimiento es rotatorio y permite cierto desplazamiento horizontal a lo largo del brazo.
- Grúas pórtico. Son la alternativa a las grúas puente cuyo peso no puede ser soportado por la estructura de la nave y por lo tanto deben tener sus propios pilares. Dada la magnitud de esta grúa, sus movimientos serán más limitados.

2.2.5.4. Robots automatizados

Los robots automatizados, también conocidos como AGVs (“Automated Guided Vehicles”) se aprovechan en algunos almacenes para reducir la mano de obra y de este modo los costes, incrementar la seguridad laboral al sustituir a los empleados en áreas más peligrosas, incrementar la transparencia de los movimientos y reducir errores. Sin embargo, conseguir un funcionamiento adecuado puede resultar ser una tarea complicada, porque existen numerosas situaciones que el robot debe reconocer y saber cómo actuar. Otros robots, personas y cambios en el entorno son problemas a los que se deberá enfrentar (Hompel y Schmidt, 2007).

Dentro de esta categoría también se incluyen las paletizadoras automatizadas que se encargan de apilar las mercancías en palés. Sustituyen a los operarios en tareas físicas y tediosas además de reducir el tiempo requerido para completar la tarea.

2.2.6. Innovaciones en gestión de almacenes

2.2.6.1. Industria 4.0

La gestión de una cadena de valor cualquiera pretende mejorar la coordinación y eficiencia de todos sus procesos con el objetivo de proporcionar a sus clientes un servicio de mayor calidad, con menores costes y tiempos de entrega (Tubis y Rohman, 2023). Para ello, es necesario la digitalización de los procesos logísticos. Según Strandhagen et al. (2017), actualmente se están desarrollando nuevas tecnologías para la gestión de almacenes y se pueden clasificar en 4 grupos:



- De ayuda a la decisión y toma de decisiones. Empleando inteligencia artificial, análisis de “big data”, realidad aumentada y realidad virtual.
- Identificación e interconectividad. Incluye el Internet de las cosas (IoT), que consiste en una red de objetos inteligentes que se comunican e interactúan entre sí. Este grupo también incluye sensores inteligentes y RFID para la identificación y seguimiento de productos.
- Flujo de información. Pretende conectar una empresa con sus clientes y proveedores obteniendo información de los mismos para realizar la planificación en tiempo real. Además, también incluye tecnologías como la nube, que permite acceder a recursos como los programas de gestión de almacenes de manera remota y proporciona copias de seguridad automáticas.
- Automatización. Orientado a sustituir la mano de obra humana por robots industriales y vehículos guiados automáticamente.

Estas tecnologías forman parte de la industria 4.0, definida según Lasi et al. (2014), como fábricas inteligentes, que mediante la incorporación masiva de sensores y actuadores, son capaces de tener un alto grado de automatización. Además, se caracterizan por disponer de sistemas ciberfísicos, para realizar la monitorización y el control de procesos. También, la industria 4.0 se caracteriza por la descentralización de los procesos, ya que, el incremento de dispositivos inteligentes aumenta la capacidad de autogestión. Por otro lado, los productos y servicios presentan un mayor grado de personalización para las necesidades de cada consumidor y se incrementa la eficiencia a la hora de aprovechar los recursos. Los aspectos clave de la industria 4.0 son: el internet de las cosas, ciberseguridad, realidad aumentada, “Big data”, robots autónomos, manufactura aditiva, simulación, integración de sistemas y computación en la nube (Figura 20).



Desarrollo de un sistema de control de inventario en tiempo real mediante tecnología RFID

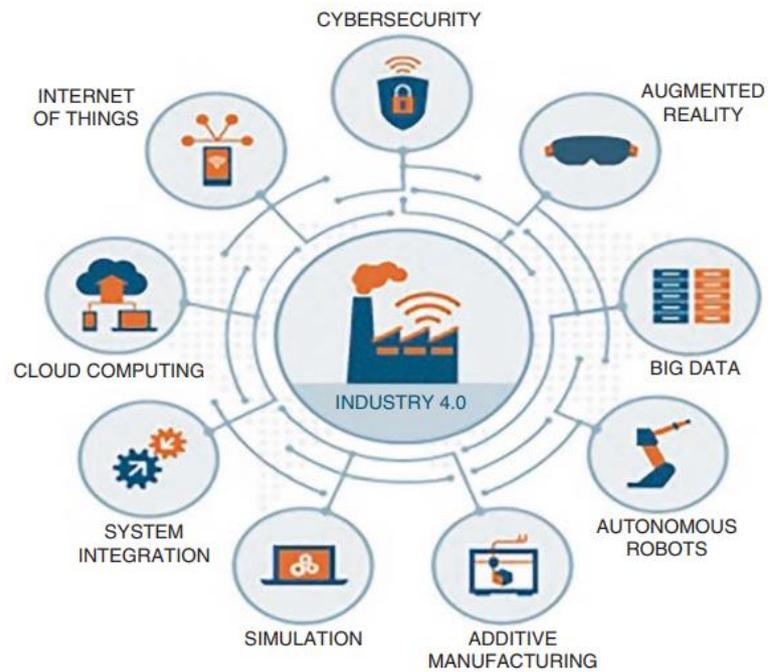


Figura 20. Componentes de la industria 4.0

Nota. Adaptado de Internet of Things for Industry 4.0: Design, Challenges and Solutions (p. 20), G. R. Kanagachidambaresan, 2020, EAI/Springer Innovations in Communication and Computing.



CAPÍTULO 3. DESARROLLO DEL SISTEMA



Desarrollo de un sistema de control de inventario en tiempo real mediante tecnología RFID





3.1. Introducción

En este capítulo se explicarán, en el apartado de descripción del hardware, las características y el funcionamiento de los dispositivos empleados para el desarrollo del sistema, de forma independiente, finalizando con la exposición del prototipo completo. Después, en el apartado de descripción del software, se explicará el funcionamiento de los programas desarrollados. El sistema o prototipo que se ha realizado en el presente trabajo lee las tarjetas o etiquetas RFID de los artículos que entran y salen de un almacén, permite cambiar el contenido de estas e indica el lugar en el que deben ser almacenados los palés a través de una pantalla. Además, guarda en un archivo de Excel toda la información relativa a las entradas, salidas y sobre productos en stock, junto con indicadores y gráficas que, permiten al usuario realizar la planificación de los movimientos y evaluar el desempeño global del almacén.

3.2. Hardware

Este apartado comienza introduciendo la plataforma Arduino y justificando su uso. Después, se mostrarán por separado los elementos utilizados para el desarrollo del prototipo junto con una explicación de su funcionamiento. Estos dispositivos son: Arduino Mega r3 2560, el sensor RFID MFRC522, las tarjetas MIFARE, la pantalla TFT, el pulsador, los LEDs, la protoboard, resistencias y los buses de comunicación.

3.2.1. ¿Qué es Arduino?

“Arduino es una plataforma electrónica y de programación en arquitectura abierta con amplia gama de aplicaciones en ciencias exactas e ingeniería” (Reyes y Cid, 2015). Es decir, es un dispositivo diseñado para permitir al usuario de una manera relativamente sencilla conectar sensores y actuadores y programar su microcontrolador para poder realizar un amplio abanico de proyectos. Arduino es de arquitectura abierta, también llamada “hardware libre”, esto supone que “las especificaciones y los diagramas esquemáticos son de acceso público” (Peña, 2017). Sus aplicaciones destacan por ser muy variadas, desde proyectos educativos de robótica o electrónica hasta la realización de prototipos para aplicaciones de agricultura, salud, mineras o energéticas (Kondaveeti et al., 2021).

Los microcontroladores que emplea para cualquiera de sus modelos son de la familia ATMEL y se programan en el entorno de Arduino que se puede descargar



de manera gratuita desde su página oficial www.arduino.cc. El lenguaje empleado es el C o C++.

Los modelos más relevantes de Arduino son el Arduino UNO R3, Arduino DUE, Arduino Leonardo, Arduino ADK Rev3, Arduino Nano y Arduino Mega 2560 Rev3. Sin embargo, también existen otros modelos destinados a aplicaciones más concretas, como pueden ser los Arduino LiLyPad que fueron concebidos para ser insertados en prendas, lo que se conoce como dispositivos “weareables”, o los Arduino GSM Sheild capaz de conectarse a internet, enviar y recibir SMS y también permite realizar y recibir llamadas (Herrero y Sánchez, 2015).

3.2.2. Justificación de su elección

Para la consecución de los objetivos de este trabajo, se ha elegido como plataforma un dispositivo Arduino. Teniendo en cuenta que en el mercado existen otras alternativas, a continuación se expone por qué ha resultado preferible frente a la utilización de otros dispositivos similares y frente a un Raspberry Pi.

Arduino, compite con otros microcontroladores como pueden ser LaunchPad, Particle Photon, Netduino o BeagleBoard, entre otras. Sin embargo, cuenta con una gran popularidad, lo cual hace que exista una gran cantidad de documentación disponible y una comunidad activa que se dedica al desarrollo de librerías, a resolver dudas en foros y a la creación de contenido multimedia. Además, cuenta con conectores en sus pines, permitiendo la creación de circuitos sin tener que soldar las conexiones previamente. Destaca también, por su fiabilidad, facilidad de uso y por disponer de un amplio abanico de módulos (como los sensores RFID) y “shields”, que son placas que se conectan sobre Arduino para proporcionar más funcionalidades. También cuenta con un entorno de programación sencillo y multiplataforma (Arduino IDE).

Otra plataforma muy conocida es Raspberry Pi. Sin embargo, por sus características no resulta tan adecuada como puede ser Arduino, ya que está orientada a otras aplicaciones. Según Ghael et al. (2020), Raspberry Pi es un potente ordenador en miniatura, muy versátil, empleado para aplicaciones como automatización del hogar, asistente de voz, creación de cámaras que se activan al detectar movimiento, automatizar sistemas, como estación multimedia, para crear servidores o funcionar como un ordenador común. Las especificaciones técnicas de un modelo como el Raspberry Pi 4 Modelo B son las siguientes:

- Procesador Broadcom BCM2711 de 64 bit de 1,5 GHz.



- 8 GB de SDRAM
- Bluetooth 5.0, BLE
- Wi-Fi 2.4 GHz y 5.0 GHz
- 4 puertos USB (2 USB 3.0 y 2 USB 2.0)
- Ethernet
- HDMI

Aunque su procesador es más potente, tiene más memoria que un Arduino 2560 y más funcionalidades (comparar con la Tabla 4), Raspberry es una plataforma orientada al desarrollo software, mientras que Arduino es más apropiado para el desarrollo hardware. Los desarrolladores que usen Arduino tienen a su disposición una mayor cantidad de librerías para el control de sensores y trabajar con esta plataforma resulta más sencillo cuando se trata del desarrollo de proyectos de electrónica (Qamarina et al., 2018).

3.2.3. Arduino Mega 2560 r3

El dispositivo que se emplea en este trabajo es el Arduino Mega 2560 Rev3. Permite la realización de proyectos más complejos que otros modelos por su mayor número de pines con 54 entradas o salidas digitales frente a las 14 del Arduino UNO. Además, cuenta con USB tipo B, para facilitar su conexión y programación con un ordenador. El suministro eléctrico también se puede proporcionar a través del jack de alimentación. Otros puntos fuertes serían, su memoria flash de 256 KB y sus cuatro puertos serie.

Está basado en el microcontrolador ATmega 2560, que tiene una arquitectura AVR. Por otro lado, R3 o Rev3 se refiere a la revisión de la placa, la cual presenta mejoras como un circuito de reset más robusto, nuevos pines como SDA y SCL. También cuenta con un pin IOREF, que sirve como voltaje de referencia (5V o 3,3V) para los “shields”. Finalmente, se ha cambiado el Atmega 8U2 por el 16U2, este es un chip que funciona como un conversor USB a serial, permitiendo la comunicación de un PC con el microcontrolador (Arduino, s.f.). A continuación, se presentan sus características básicas en la Tabla 4.

Arduino Mega 2560 Rev3	
Parámetro	Valor
Frecuencia de funcionamiento	16 MHz
Microcontrolador	ATmega 2560
Conexión	USB
Número de E/S digitales	54
De las cuales pueden actuar como salidas PWM	15
Número de entradas analógicas	16

Bits de resolución de las entradas analógicas	10
Puertos serie	4
Memoria Flash	256 KB
SRAM	8 KB
EEPROM	4 KB
Voltaje de operación	Entre 7 y 12V
Corriente en las E/S digitales	20 mA
Corriente en los pines de 3,3 V	50 mA
Dimensiones (mm)	101,52 x 53,3
Peso	37 g

Tabla 4. Especificaciones técnicas del Arduino Mega 2560 Rev3

Nota. Arduino (s.f.). Arduino Mega 2560 Rev3. Arduino.cc. <https://store.arduino.cc/products/arduino-mega-2560-rev3>

En la Figura 21 se muestra una imagen del dispositivo Arduino empleado.

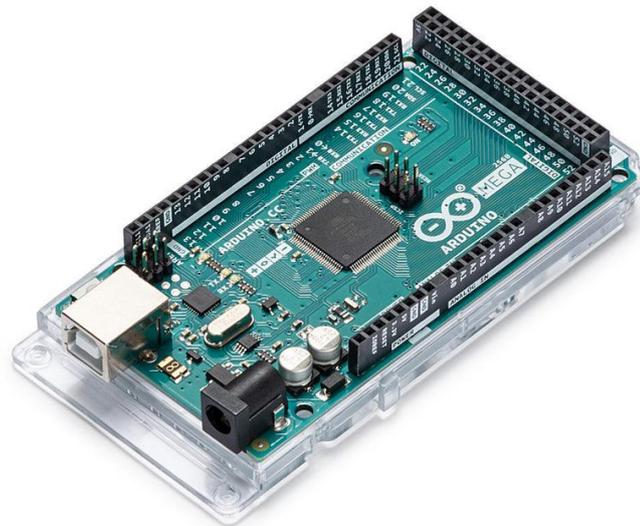


Figura 21. Arduino Mega 2560 Rev3

Nota. Adaptado de Arduino Mega 2560 Rev3 [Fotografía], por Arduino, s.f., Arduino.cc (<https://store.arduino.cc/products/arduino-mega-2560-rev3?queryID=undefined>)

3.2.3.1. Microcontrolador ATmega 2560

El microcontrolador ATmega 2560, emplea un bus de datos de 8 bits y una arquitectura Harvard con el propósito de optimizar el rendimiento, de tal modo que las instrucciones y los datos del programa se almacenan en memorias diferentes para poder acceder a ambas de manera simultánea a través de buses diferentes, ahorrando de este modo tiempo de ejecución (en



comparación con la arquitectura Von Neumann). Un microcontrolador se divide en recursos comunes y especiales. En los recursos comunes se incluyen, entre otros, la unidad de control, la unidad de proceso que en conjunto forman la CPU y los buses (Características generales de los microcontroladores, 2023). Los recursos especiales variarán dependiendo de la aplicación a la que esté destinado. Adaptando al ATmega 2560 la información proporcionada en Características generales de los microcontroladores (2023) e Introducción a los procesadores digitales (2023) se puede decir lo siguiente de sus recursos comunes y especiales:

- La unidad de control tiene el propósito de obtener, decodificar y ejecutar instrucciones. Cuenta con bloques como el contador del programa (indica cual es la dirección de la siguiente instrucción a ejecutar), el registro de instrucciones (las almacena), el decodificador y el secuenciador (las ejecuta), que está conectado a las líneas de control.
- La unidad de proceso está formada por la ALU, que se encarga de realizar las operaciones aritmético-lógicas y por los registros donde se almacena información de manera temporal, como por ejemplo el registro de estado. El ATmega2560 también tiene 32 registros de propósito general de 8 bits cada uno.
- Los buses.
 - En primer lugar, el bus de datos comunica valores de 8 bits entre los diferentes elementos del microcontrolador, como pueden ser las líneas de entrada y salida digitales.
 - En segundo lugar, estaría el bus de instrucciones. Cada instrucción indica las líneas que se deben activar o desactivar en el bus de control.
 - El bus de control transmite señales como el reloj o el RESET.
 - El bus de direcciones que indica la posición de la memoria de la cual se extrae o escribe información, se usa junto con el bus de datos.
- Otros recursos comunes incluyen elementos como la memoria EEPROM, la memoria flash del programa, la memoria SRAM de datos, el reloj y los módulos I/O (de entrada y salida) que conectan la CPU con los pines del Arduino.
- Los recursos especiales empleados en el ATmega 2560 son principalmente el perro guardián y el comparador analógico. El perro guardián tiene el propósito de resetear el programa en caso de que surja algún error. Como este tipo de dispositivos suelen operar de manera autónoma, sin una interfaz gráfica y sin una supervisión constante, requieren de sistemas de este tipo para reaccionar de manera autónoma ante errores. Funciona realizando una cuenta atrás que es

reiniciada por el programa antes de que finalice. Este reinicio se realizará a tiempo si el programa funciona correctamente, de lo contrario, finalizará la cuenta atrás y se forzará un reseteo del sistema. El comparador analógico indica cuál de las dos señales que recibe es mayor. Cuenta también con otros bloques esenciales como la unidad SPI para las comunicaciones con dispositivos periféricos y la unidad de interrupciones. El diagrama de bloques simplificado del microcontrolador ATmega2560 se muestra en la Figura 22.

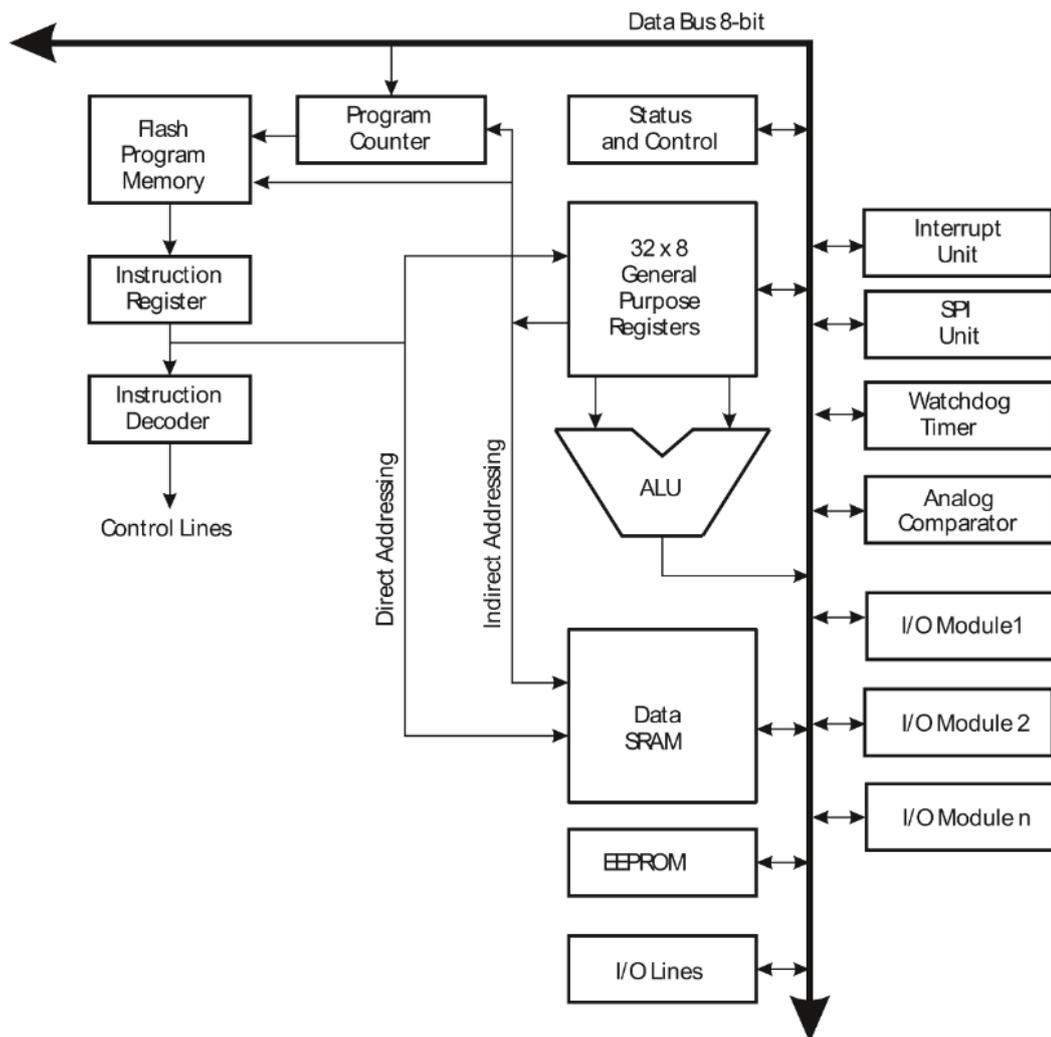


Figura 22. Diagrama de bloques simplificado del microcontrolador ATmega 2560

Nota. Adaptado de *Diagrama de bloques de la arquitectura AVR* (p. 11), de Microchip Technology Inc., 2014, Microchip (https://ww1.microchip.com/downloads/en/devicedoc/atmel-2549-8-bit-avr-microcontroller-atmega640-1280-1281-2560-2561_datasheet.pdf)



3.2.3.2. Entorno de Desarrollo Integrado

El Entorno de Desarrollo Integrado (IDE, ver Figura 23) es el software libre que proporciona Arduino para la escritura del código que controla los sensores y actuadores empleados en el proyecto. Ofrece tres elementos principales que son esenciales a la hora de programar que son el editor de código, el compilador y el depurador. El editor de código, es la plataforma donde se escribe y modifica el código fuente que contendrá las instrucciones, variables, funciones, y bucles. El compilador es un programa que al ejecutarse traduce el lenguaje de alto nivel que se ha plasmado en el editor en lenguaje de máquina y lo almacena en la memoria del Arduino. Finalmente, el depurador se encarga de recorrer el código escrito en el editor en busca de errores y los resalta (Peña C., 2017).



Figura 23. Nuevo proyecto en el Arduino IDE

3.2.4. Sensor RFID MFRC522

Según la clasificación realizada en la Figura 7, que se encuentra en el capítulo del estado del arte, este trabajo se vale del lector MFRC522 que opera con una frecuencia alta de 13,56 MHz, ofreciendo un alcance a corta distancia, con un acoplamiento inductivo, gama alta (ofrece encriptación, lectura y escritura) y se comunica con transpondedores pasivos como son las tarjetas y llaveros MIFARE.

Este periférico se conecta al bus SPI, localizado en los pines 50 (MISO), 51 (MOSI), 52 (SCK), el pin de selección de chip deberá ser una salida digital. Igualmente, el pin de reset (RST) será una salida digital. También cuenta con la conexión a 3,3V, conexión a tierra y un pin IRQ opcional. En la Figura 24 se muestra una imagen del lector MFRC522.

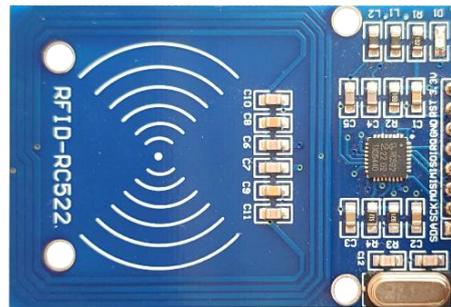


Figura 24. Sensor MFRC522 empleado para la elaboración del prototipo

El sensor MFRC522 es el más popular para Arduino, su precio es asequible, existen librerías específicas para su utilización y presenta las características mostradas en la Tabla 5.

MFRC522	
Parámetro	Valor
Frecuencia de comunicación	13,56 MHz
Voltaje	Entre 2,5V y 3,3V
Interfaces	SPI, UART, I2C
Distancia operativa	Hasta 5cm
Velocidad de transferencia	Hasta 10 Mbit/s (SPI)
Encriptación	MF1xxS20, MF1xxS50, MF1xxS70

Tabla 5. Especificaciones técnicas del lector MFRC522

3.2.5. Tarjetas MIFARE

El sistema RFID de este trabajo emplea como transpondedores las tarjetas MIFARE S50 (ver Figura 25). Se compone de la carcasa de plástico, de una antena o bobina de 4 vueltas y de un chip.

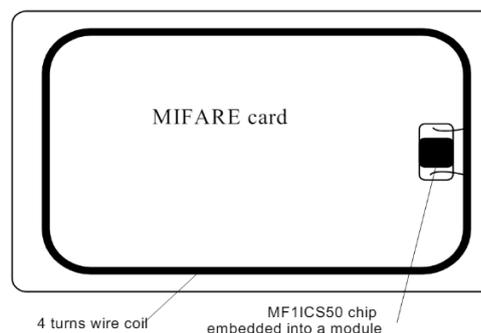


Figura 25. Diagrama de una tarjeta MIFARE

Nota. Adaptado de Organización de la memoria (p. 1), de NXP Semiconductors, 2007, NXP (<https://cdn-shop.adafruit.com/datasheets/S50.pdf>)



A continuación, se muestran las especificaciones técnicas más relevantes de la tarjeta en la Tabla 6.

MIFARE S50	
Parámetro	Valor
EEPROM	1 KB
Retención de los datos	10 años
Ciclos de escritura soportados	100.000
Norma ISO/IEC	14443 Tipo A
Frecuencia de funcionamiento	13,56 MHz
Velocidad de transferencia de datos	106 kbits/s
Distancia operativa	Hasta 10cm
Autenticación de seguridad	Autenticación mutua de tres pasos (ISO/IEC DIS 9798-2)
Claves	2 claves por sector
UID	4 bytes (único para cada tarjeta)

Tabla 6. Especificaciones técnicas de la tarjeta MIFARE S50

La norma ISO 14443A es empleada para diversas aplicaciones, pero es de especial utilidad para la identificación de productos. Esta norma “establece los estándares para la comunicación inalámbrica de corto alcance entre un lector y una tarjeta de proximidad o dispositivo NFC” (Organización Internacional de Normalización, s.f.). Es decir, seguir esta norma supone, entre otras características, operar en una frecuencia de 13,56MHz, permitir su lectura hasta una distancia de 10 cm, comunicarse a velocidades de hasta 424 kbps y emplear un sistema de autenticación mutua como garantía de seguridad a la hora de realizar el intercambio de datos (Organización Internacional de Normalización, s.f.). Un dispositivo que se ajuste a la norma ISO 14443A presenta ventajas como pueden ser garantías de seguridad, interoperabilidad, facilidad de uso y versatilidad.

Por otro lado, la autenticación mutua permite que tanto el lector como la tarjeta verifiquen sus identidades antes de realizar una transferencia de datos. Esta medida de seguridad es de vital importancia en aplicaciones como por ejemplo, en un pago con tarjeta en el que al atacante puede falsear una transferencia o, por otro lado, un lector malicioso podría vulnerar la información personal del cliente con consecuencias monetarias o suplantación de identidad (Piramuthu, 2011).

3.2.5.1. EEPROM de la MIFARE S50

El chip contiene una interfaz de radiofrecuencia, una unidad de control digital y una memoria EEPROM (ver Figura 26). Dentro de la unidad de control digital se realizan operaciones aritmético-lógicas, se accede a la EEPROM a través de una interfaz y se realizan las funciones de encriptado. Ahora, se analizará en detalle la memoria, ya que el conocimiento de su estructura es necesario a la hora de programar en Arduino la lectura y escritura de las tarjetas.

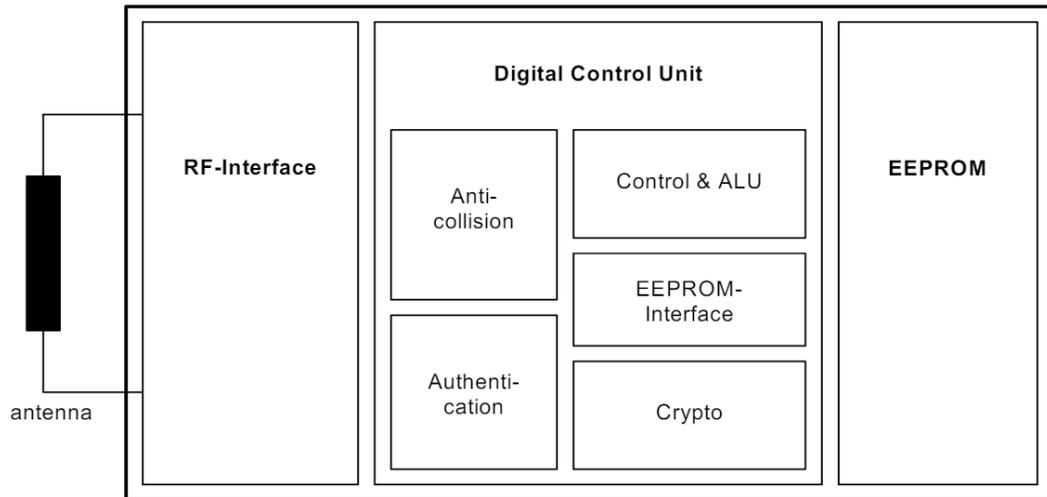


Figura 26. Diagrama de bloques del chip de la tarjeta MIFARE S50

Nota. Adaptado de Organización de la memoria (p. 3), de NXP Semiconductors, 2007, NXP (<https://cdn-shop.adafruit.com/datasheets/S50.pdf>)

Su memoria EEPROM se organiza en 16 sectores y cada uno de ellos se compone de 4 bloques. Un bloque de memoria consta de 16 bytes (ver Figura 27). Sin embargo, no toda la capacidad está disponible para almacenar cualquier tipo de información. Esto se debe a que en cada sector, se reserva el último bloque para almacenar las claves que permiten la lectura o escritura. Además, el bloque 0, situado en el sector 0 se denomina bloque del fabricante y está destinado al almacenamiento del UID de la tarjeta el cual no puede ser modificado (NXP Semiconductors, 2007).



Sector	Block	Byte Number within a Block														Description		
		0	1	2	3	4	5	6	7	8	9	10	11	12	13		14	15
15	3	Key A						Access Bits			Key B					Sector Trailer 15		
	2																	Data
	1																	Data
	0																	Data
14	3	Key A						Access Bits			Key B					Sector Trailer 14		
	2																	Data
	1																	Data
	0																	Data
:	:																	
1	3	Key A						Access Bits			Key B					Sector Trailer 1		
	2																	Data
	1																	Data
	0																	Data
0	3	Key A						Access Bits			Key B					Sector Trailer 0		
	2																	Data
	1																	Data
	0																	Manufacturer Block

Figura 27. Estructura de la memoria EEPROM de la tarjeta MIFARE S50

Nota. Adaptado de *Organización de la memoria* (p. 7), de NXP Semiconductors, 2007, NXP (<https://cdn-shop.adafruit.com/datasheets/S50.pdf>)

El cuarto bloque de cada sector, se denomina sector tráiler y se divide en tres partes, la llave A que ocupa los primeros 6 bytes, los bits de acceso que ocupan los bytes del 6 hasta el 9 y la llave B que se localiza en el resto. La clave A nunca podrá ser leída, aunque se puede modificar. Por otra parte, el uso de la llave B es opcional y en caso de no ser usada sus bytes pueden ser empleados para almacenar datos.

En el resto de bloques, llamados bloques de datos tienen la escritura permitida. La información a almacenar no debe superar los 16 bytes por lo que se tendrá que hacer uso de los siguientes bloques para los datos que los rebasen.

También se pueden configurar algunos bloques como bloques de valores, destinados a aplicaciones como almacenar y modificar el valor de un saldo. Cada bloque de este tipo está limitado a 4 bytes de capacidad, debido a que la información se guarda de manera redundante 3 veces consecutivas. Además, la dirección se almacena 4 veces (4 bytes), la dirección ocupa 1 byte. Esta redundancia pretende garantizar la seguridad e integridad de los datos.

Las operaciones que permite esta memoria son leer, escribir, incrementar, decrementar, transferir y restaurar. Los resultados de las operaciones incrementar o decrementar el valor de un bloque se almacenan en un registro interno de datos. Por otro lado, la operación de transferir copia los contenidos de un registro interno en un bloque y la orden de restaurar realiza el proceso



inverso. Todas las operaciones se realizan sobre un bloque. Estas instrucciones se pueden ejecutar en todos los bloques excepto en los sectores tráiler, en los que solamente se puede leer o escribir.

Los bits de acceso controlan cuales son los requisitos para poder leer o escribir en la memoria. Es decir, en función de los valores que tomen, se pedirá la llave A, la B, cualquiera de las dos o el acceso a la memoria estará denegado.

Es importante tener en cuenta que si se intenta acceder a la memoria sin cumplir con los requisitos establecidos por los bits de acceso, el sector al que se ha intentado acceder se bloqueará permanentemente.

3.2.5.2. Formato

Las tarjetas MIFARE S50 también se fabrican en formato llavero (ver Figura 28). Aunque la apariencia es diferente, las especificaciones técnicas y el transpondedor que se encuentra en su interior son los mismos.



Figura 28. Tarjeta y llaveros MIFARE S50

3.2.6. Pantalla TFT

En este trabajo se emplea para indicar las coordenadas a la que deben ser llevados los palés que entran en el almacén. Su resolución es de 128 por 160 píxeles RGB y puede llegar a mostrar 262.144 colores diferentes. Sus dimensiones son de 34,5 x 58 x 7,5 mm, con un área útil de visualización de 1,8 pulgadas o 32 x 38 mm. Otras características relevantes son que funciona con un controlador ST7735, presenta un consumo de hasta 50 mA, cuenta con retroiluminación y se controla a través del bus SPI. Por otro lado, aunque en este trabajo no se ha utilizado, este modelo de pantalla cuenta con una ranura

para tarjetas SD que permite mostrar imágenes. A continuación se muestran todas las conexiones que se deben realizar en la Tabla 7.

CONEXIONES DE LA PANTALLA CON EL ARDUINO 2560	
Pin Arduino	Pin Pantalla
3,3 V	LED
52	SCK
51	SDA
9	A0
Cualquier salida digital (25)	RESET
10	CS
GND	GND
5 V	VCC

Tabla 7. Conexiones de la pantalla con Arduino

Este dispositivo puede controlarse empleando la biblioteca TFT, que viene instalada por defecto en el IDE de Arduino. Sin embargo, se han empleado las librerías “Adafruit GFX Library” y “Adafruit ST7735 Library” para poder tener ejemplos que faciliten el aprendizaje y para poder emplear funciones más complejas que permitan una mayor libertad a la hora de mostrar la información. En la Figura 29 se muestra la pantalla que se ha empleado.

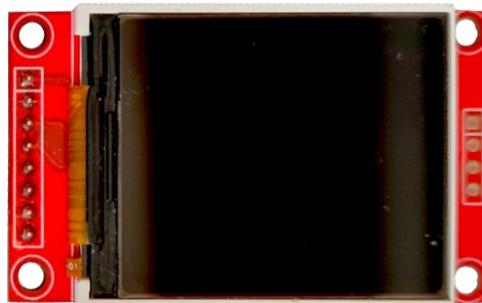


Figura 29. Pantalla 1.8 TFT 8P 128*160

3.2.7. Pulsador

El pulsador se encarga de alternar entre el modo lectura y el modo escritura y su funcionamiento es el siguiente. Solamente cuando es presionado se cierra el circuito. Las patillas que se encuentran enfrentadas en lados opuestos permanecen siempre conectadas, mientras que las patillas contiguas se conectan solamente cuando se cierra el circuito (ver Figura 30).

Según Llamas (2014), para conectar correctamente el pulsador, es necesario conectar en serie una resistencia, que en este caso es de 1 k Ω . De lo contrario,

cuando se presionase el pulsador se produciría un cortocircuito, ya que se estaría conectando 3,3 V directamente con tierra. Luego, a la hora de seleccionar una resistencia hay que tener en consideración que las resistencias mayores reducen la corriente que circula y por lo tanto reducen el consumo aunque se incrementan las posibilidades de una lectura incorrecta. Con las resistencias más pequeñas sucede lo contrario.

Después, hay dos maneras de conectar un pulsador, dependiendo de si permite el paso de la corriente a través del pin o no cuando se pulsa. El montaje será Pull-Up si se comunica un 1 cuando el pulsador está abierto y será Pull-Down si se comunica un 0 cuando el pulsador está abierto, como el pulsador del prototipo de este trabajo.

En el montaje Pull-Up se conecta el pin de entrada digital en serie con la resistencia y la resistencia en serie con el potencial positivo, mientras que el pulsador estará conectado en paralelo con el pin, separando al resto del circuito del potencial negativo. El montaje Pull-Down alternará la posición de la resistencia con el pulsador (ver Figura 31).



Figura 30. Pulsador

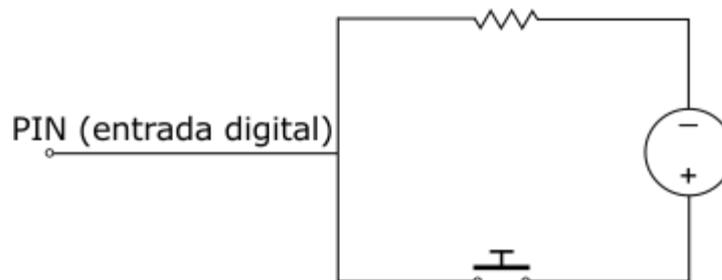


Figura 31. Montaje Pull-Down de un pulsador

3.2.8. LEDs

Cuando se está trabajando con el prototipo de este trabajo, se necesita saber si el dispositivo realiza correctamente las funciones que deseamos y para ello hay que utilizar un elemento que lo comunique. Los LEDs se emplean para indicar qué es lo que está sucediendo con el dispositivo (ver Figura 32), es decir, si ha leído o escrito en la tarjeta o si se ha cambiado de modo. Los LEDs

son diodos y por lo tanto solamente permiten el paso de la corriente en polarización directa, es decir, conectando el ánodo (patilla larga) al voltaje y el cátodo (patilla corta) a tierra. Del mismo modo que con el pulsador, es necesario conectar una resistencia en serie para que la intensidad que circula a través del LED no lo dañe, para el prototipo se han empleado resistencias de 1 k Ω . Generalmente, la corriente máxima admitida por un diodo LED es de unos 20 mA. Entonces, empleando la ley de Ohm, que establece que $V = I \cdot R$, se puede hallar la resistencia mínima que debe utilizarse.

Para un voltaje de 5 V y un diodo que requiere para funcionar aproximadamente 2 V, la caída de tensión en la resistencia deberá ser de unos 3 V y como la intensidad máxima es de 0,02 A entonces la resistencia mínima será de $3 \text{ V} / 0,02 \text{ A} = 150\Omega = R$.

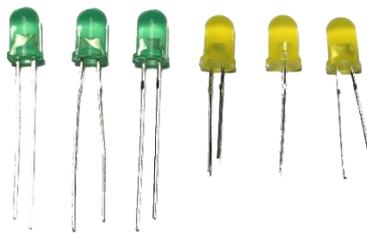


Figura 32. LEDs

3.2.9. Protoboard

Es una placa que permite montar circuitos de manera sencilla, rápida y sin la necesidad de realizar soldaduras. Este tablero conecta todos los pines que se encuentran junto a la misma línea azul (para la conexión a tierra) o roja (para la alimentación). Por otra parte, conecta entre sí los 5 pines perpendiculares a las líneas de colores y cada uno de estos grupos de pines está aislado de los demás (Figura 33).

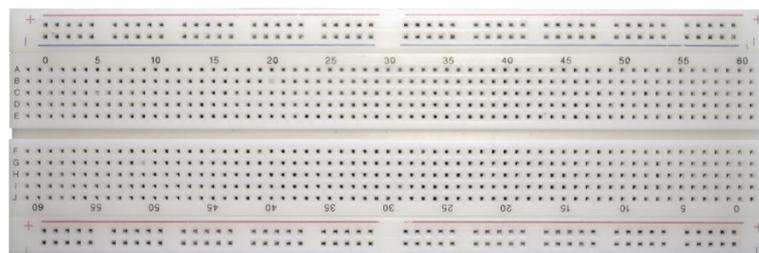


Figura 33. Protoboard

3.2.10. Resistencias

Se emplean en conjunto con el pulsador, con los LEDs y también se conectan en serie con los pines SCK, SDA, AO, RESET y CS de la pantalla. Todas las resistencias que se han empleado son de 1k Ω y su código de colores es marrón para la primera banda (valor 1), negro para la segunda (valor 0), negro para la tercera (valor 0), marrón para la cuarta (multiplicador x10) y marrón para la quinta ($\pm 1\%$ de tolerancia). También existen resistencias con códigos de cuatro bandas, para las cuales las dos primeras bandas formarían un número de dos dígitos, la tercera banda sería el multiplicador y la cuarta la tolerancia. Según Demir et al. (2018), el código de colores de las resistencias es el que se muestra en la Tabla 8.

CÓDIGO DE COLORES DE LAS RESISTENCIAS					
Color	Banda 1	Banda 2	Banda 3 (5 bandas)	Multiplicador	Tolerancia
Negro	0	0	0	x 1	$\pm 20\%$
Café	1	1	1	x 10	$\pm 1\%$
Rojo	2	2	2	x 100	$\pm 2\%$
Naranja	3	3	3	x 1.000	$\pm 3\%$
Amarillo	4	4	4	x 10.000	- 0, + 100%
Verde	5	5	5	x 100.000	$\pm 0,5\%$
Azul	6	6	6	x 1.000.000	$\pm 0,25\%$
Violeta	7	7	7	x 10.000.000	$\pm 0,1\%$
Gris	8	8	8	x 100.000.000	$\pm 0,05\%$
Blanco	9	9	9	x 1.000.000.000	$\pm 10\%$
Dorado	-	-	-	x 0,1	$\pm 5\%$
Plata	-	-	-	x 0,01	$\pm 10\%$

Tabla 8. Código de colores de las resistencias

Las resistencias empleadas en el prototipo se muestran en la Figura 34.



Figura 34. Resistencias



3.2.11. Conexiones serie: UART y SPI

- Protocolo UART. Conexión del dispositivo Arduino Mega 2560 R3 con el ordenador.

El protocolo empleado para conectar Arduino con el ordenador personal se denomina UART (Universal Asynchronous Receiver-Transmitter). Según Peña y Legaspi (2020) este protocolo es un tipo de comunicación en serie, lo cual implica el envío secuencial de bits a través de un único canal, y permite únicamente la conexión entre dos dispositivos. Para el envío de la información se emplea una sola línea, llamada Tx. Generalmente, existe una segunda línea para la recepción de los datos, llamada Rx. También consta de las líneas de alimentación y tierra.

Además, el protocolo UART es asíncrono, es decir, los dispositivos que intervienen en la comunicación no comparten un reloj para sincronizar el envío y la recepción de datos. Entonces, antes de iniciar el intercambio de bits, los dispositivos configuran la misma velocidad de transmisión, que normalmente es de unos 9.600 baudios o bits por segundo. Arduino también admite velocidades de 4.800, 19.200 y 38.400 baudios, entre otras. Arduino Mega 2560 consta de cuatro puertos serie UART. El primer puerto serie es el empleado más frecuentemente y es el que se emplea en este trabajo, debido a su localización en el conector USB tipo B (pines 0 y 1). Esto simplifica su conexión con el ordenador. Estas líneas trabajan con voltajes de entre 5V y 3,3V.

Por otro lado, según Peña y Legaspi (2020), la transmisión de datos se realiza en paquetes compuestos por un bit de inicio, entre cinco y nueve bits de información, un bit de paridad y dos bits de parada, en este orden. El bit de paridad será 0 si la cantidad de bits de información que valen 1 es un número par, de lo contrario valdrá 1. La paridad es una manera de verificar si existen errores en la comunicación, comparando en el dispositivo de recepción, si efectivamente hay un número par o impar de bits con valor 1. El proceso de transmisión de información entre dos dispositivos es el siguiente.

Se parte de la información del bus de datos, que el dispositivo UART de transmisión recibe en formato paralelo. Después, el dispositivo agrega los bits de inicio, paridad y parada para enviarlos bit a bit. El dispositivo UART de recepción, muestrea la señal y realiza el proceso inverso, eliminando todos los bits que no sean los de información y transmitiendo esta información a su bus de datos (Peña y Legaspi, 2020).



Por otro lado, los puertos como el DB9 se dedicaban para realizar las comunicaciones en serie, pero los ordenadores personales ya no incorporan estos puertos físicos (COM) y en su lugar, los emulan a partir de un puerto USB. Windows permite el uso de varios puertos virtuales y asigna un nombre a cada uno (COM1, COM2, COM3 y así sucesivamente) (Axelson, 2000).

El cable empleado, como se puede observar en la Figura 35, consta de un conector USB tipo A en un extremo y tipo B en el otro. Además de transmitir los datos en ambas direcciones, proporciona la alimentación al dispositivo Arduino.



Figura 35. Cable USB tipo A a USB tipo B, conexión PC-Arduino

- Bus SPI.

Según (Dhaker, 2018) el bus SPI (Serial Peripheral Interface) es una interfaz empleada frecuentemente para la conexión de microcontroladores con sensores, conversores analógico-digitales o memorias. Funciona de manera síncrona y es una interfaz maestro-esclavo. Además, opera en modo full-duplex, lo que implica que tanto el maestro como el esclavo envían información simultáneamente. El bus empleado por el dispositivo Arduino de este trabajo emplea 4 cables, aunque también existen otros buses SPI que se componen de 3 cables o líneas. Estas 4 líneas son el reloj, también llamado CLK, el pin CS (Chip Select), MOSI (Master Out Slave In) y MISO (Master In Slave Out). La información se intercambia del maestro al esclavo a través de la línea MOSI y del esclavo al maestro mediante la MISO. En concreto, el Arduino Mega 2560 Rev3 soporta el bus SPI en los pines 50 (MISO), 51 (MOSI), 52 (SCK o Serial Clock) y 53 (SS o Slave Select).



El bus SPI conecta al dispositivo maestro, que es el que genera la señal de reloj, con el dispositivo esclavo. Solamente puede haber un dispositivo maestro, que en este caso es el microcontrolador de Arduino y los dispositivos esclavos son los sensores RFID y la pantalla TFT. Como a través del bus no se pueden comunicar múltiples dispositivos esclavo al mismo tiempo, es necesario seleccionar cada uno de ellos en instantes diferentes empleando un cable CS por cada esclavo. Además, “generalmente, el CS de es una señal activa en nivel bajo” (Dhaker, 2018). Lo que quiere decir que para seleccionar un dispositivo se debe transmitir un cero lógico a través del cable.

Por otra parte, el reloj se caracteriza por su polaridad (CPOL) y por su fase (CPHA). CPOL y CPHA son dos bits y por lo tanto habrá dos polaridades y dos fases distintas posibles. Entonces, mediante la combinación de ambos atributos, el bus SPI puede funcionar en 4 modos distintos.

La polaridad establece a qué nivel está activa la señal de reloj (nivel bajo o alto). Es decir, si la señal se encuentra activa cuando su valor es 1 o 0. Si está activa a nivel alto, quiere decir que el estado activo se corresponde con el 1 lógico. La alternativa es que la señal se encuentre activa a nivel bajo o activa cuando la señal vale 0. Luego, la fase indica en qué flanco de reloj se muestrea la información. Si el CPHA vale 0, la información será muestreada en el flanco de subida y si vale 1, en el flanco de bajada.

Finalmente, a la hora de conectar múltiples esclavos a un dispositivo maestro hay dos maneras posibles de hacerlo. O bien se emplea la conexión directa o la conexión encadenada. En este trabajo se utiliza la primera manera, que emplea tantas líneas CS como dispositivos esclavos haya. Cuando se requiere la intervención de uno de los dispositivos, este se selecciona, permitiendo que acceda a las líneas MOSI/MISO, al mismo tiempo que se deshabilitan los demás. La segunda manera, supone que todos los esclavos conectados comparten las líneas de CS y de CLK, con la diferencia de que la información se transmite en serie de un dispositivo a otro. Esto permite conectar más esclavos cuando el microprocesador no disponga de las suficientes líneas CS para cada uno de ellos.

3.3. Prototipo

Finalmente, empleando los dispositivos anteriormente mencionados se realiza el montaje del prototipo. El lado derecho de la protoboard según se observa en la Figura 36, representa la entrada del almacén con la pantalla, el led verde y el sensor RFID de entrada. El lado izquierdo con el led amarillo con su sensor

RFID representan la salida. En la Figura 36, se muestra el prototipo del proyecto con sus componentes numerados, para facilitar su identificación.

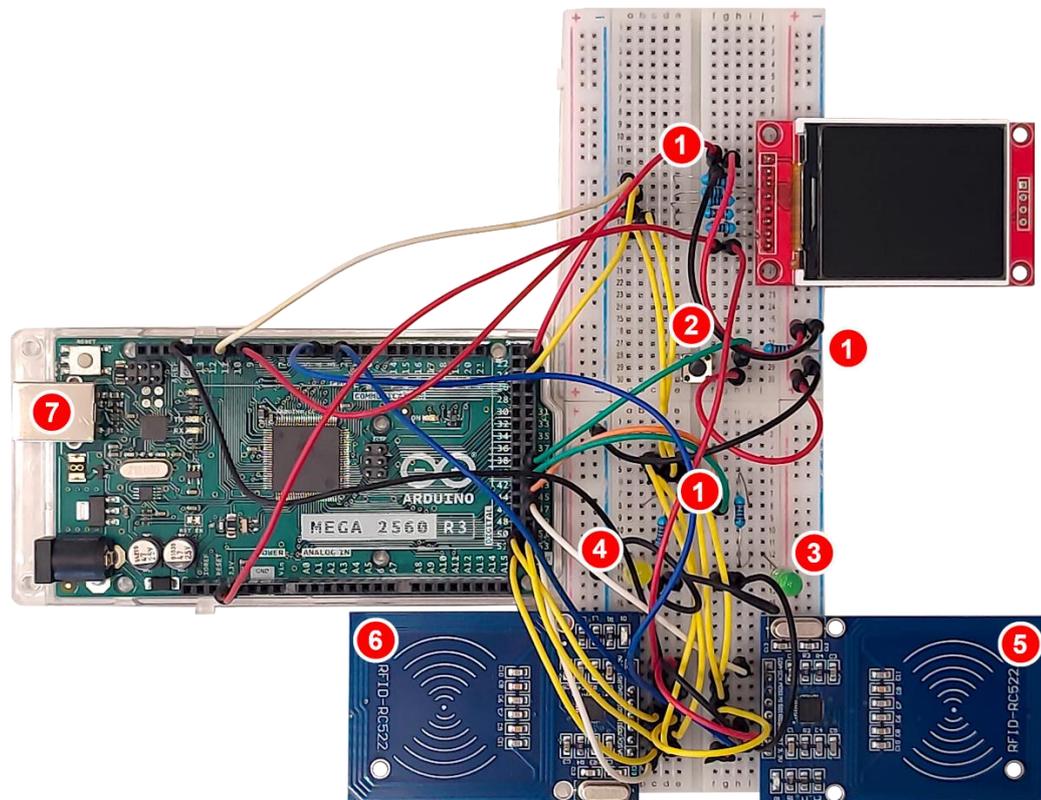


Figura 36. Prototipo del proyecto

- Componente 1: son las resistencias de 1k Ω y se emplean en conjunto con la pantalla, con el pulsador y con los LEDs.
- Componente 2: es el pulsador.
- Componente 3: es el LED verde que indica la lectura de las etiquetas entrantes.
- Componente 4: es el LED amarillo que indica la lectura de las etiquetas salientes.
- Componente 5: es el sensor RFID de entrada.
- Componente 6: es el sensor RFID de salida.
- Componente 7: es el puerto serie 1, UART (USB tipo B).
- Cables amarillos: es el bus SPI.
- Cables rojos: proporcionan la alimentación (3V o 5V).
- Cables negros: es la conexión a tierra (GND).
- Cables azules: son las líneas de RESET.
- Cables verdes y naranja: suministran voltaje para los LEDs y pulsador.
- Cables blancos: son las líneas CS.



- Excepciones: cable amarillo del pin 25 es la línea de RESET y el cable rojo del pin 9 es la línea A0, ambos de la pantalla.

RESUMEN DE COMPONENTES	
COMPONENTE	CANTIDAD
Sensor RFID	2
Pantalla TFT	1
LEDs	2
Resistencias 1 k Ω	8
Pulsador	1
Cables (macho - macho)	33
Tablero de circuitos	1
Arduino Mega 2560 R3	1
Cable USB tipo A a USB tipo B	1
Tarjetas/llaveros MIFARE S50	4

Tabla 9. Resumen de componentes del prototipo

En el siguiente apartado, se muestran las conexiones de Arduino con los periféricos.

3.3.1. Diagramas del circuito

En la Figura 37 se muestra el diagrama de conexiones del prototipo. Este diagrama se ha realizado con Proteus y el archivo original se encuentra en los anexos. En la Figura 38 y en la Figura 39 se muestran los mismos contenidos pero ampliados para facilitar su lectura.

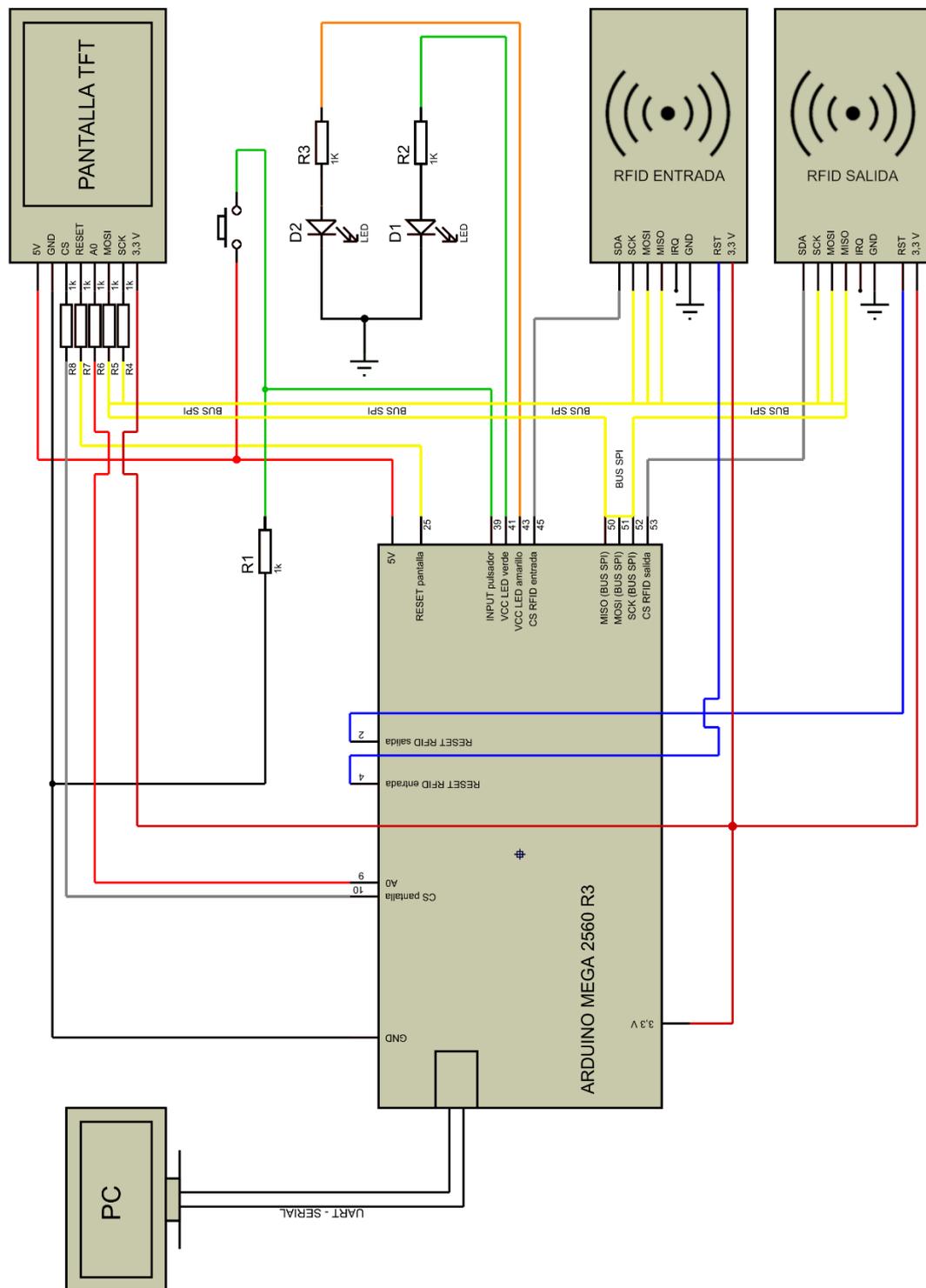


Figura 37. Diagrama del circuito Arduino-periféricos

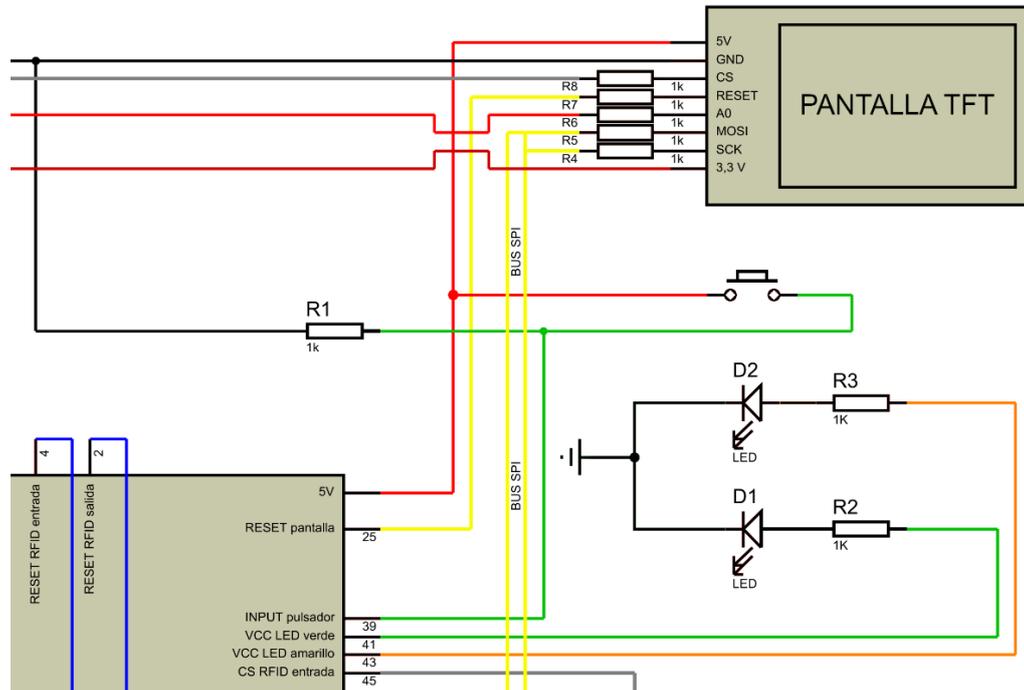


Figura 38. Conexiones de la pantalla TFT, pulsador y LEDs

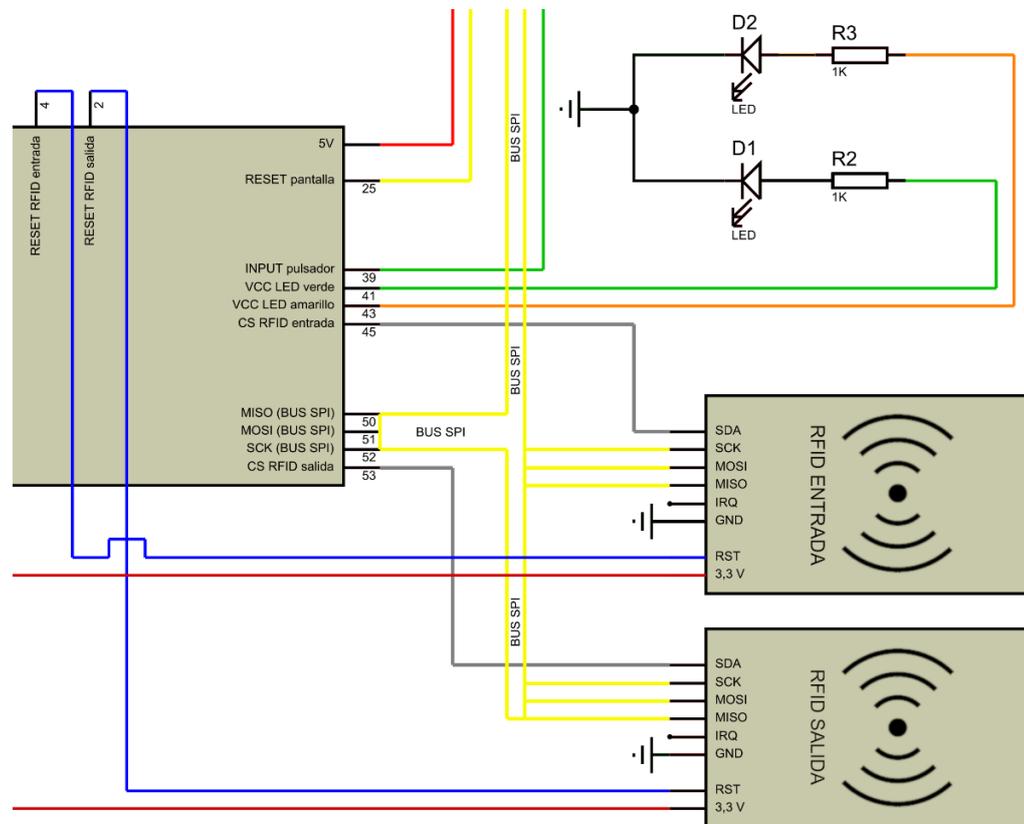


Figura 39. Conexiones de los LEDs y los lectores RFID



Desarrollo de un sistema de control de inventario en tiempo real mediante tecnología RFID





CAPÍTULO 4. PROGRAMACIÓN Y GUÍA DE USO



Desarrollo de un sistema de control de inventario en tiempo real mediante tecnología RFID





4.1. Introducción

Este capítulo comenzará explicando con la ayuda de diagramas de flujo el funcionamiento del programa que ejecutará el microcontrolador Arduino. Después, se explicará el funcionamiento de la aplicación de Excel empezando con la guía de uso y se continuará con la explicación de las macros empleadas.

4.2. Arduino

El programa almacenado en la memoria del dispositivo Arduino Mega r3 2560 es el encargado de realizar las funciones de lectura y escritura sobre las tarjetas RFID mediante el control de los dos sensores MFRC522. Además, gestiona el envío de datos de las tarjetas a Excel y la recepción de datos provenientes de la hoja de cálculo “Salida de datos”. Esta llegada de datos puede estar destinada a la reescritura de las tarjetas o a mostrar por la pantalla TFT la ubicación del almacén donde debe ser situado el palé entrante. También, se permite el cambio de modo, bien sea, lectura o escritura, mediante la activación de un pulsador. Finalmente, dos leds, uno verde y otro amarillo se encenderán para indicar que el proceso de lectura, escritura o cambio de modo ha sido completado correctamente.

La estructura del programa es sencilla: se comienza con las declaraciones de librerías y variables, se continúa con la inicialización y con el bucle principal. Finalmente, el resto del código se agrupa en funciones y cabe remarcar el uso de una función que se ejecuta como subrutina de atención de interrupciones del puerto serie, llamada serialEvent. La cual será esencial para escribir en la tarjeta RFID la información más reciente que el usuario ha introducido. En primer lugar, se describirán las librerías empleadas y luego se mostrarán los diagramas de flujo para el bucle de inicialización, para el bucle principal y para las funciones. El programa completo puede ser localizado en los anexos.

4.2.1. Librerías empleadas

Como se ha mencionado previamente, parte del éxito que ha experimentado Arduino se debe a la participación de la comunidad, especialmente creando librerías. Las librerías son un conjunto de funciones que se importan al código para hacer su programación más sencilla y rápida.

En Arduino hay dos tipos de librerías, las que vienen instaladas por defecto, como la librería SPI que se ha empleado en este proyecto para el control de los periféricos y las de terceros que deben ser descargadas e instaladas.



Para descargar las librerías e incorporarlas al proyecto de Arduino se ha empleado el gestor de librerías, en el que simplemente se busca el nombre de la librería necesaria y se pulsa el botón de instalar. También es posible realizar la instalación de las librerías de otros dos modos. En ambos casos, las librerías se buscan en GitHub directamente:

- Instalando un archivo comprimido en formato ZIP: Con el archivo ZIP descargado se debe buscar en la barra de menús la pestaña de Programa. Después, de todas las opciones que aparecen se elige Incluir librería y finalmente se busca la opción de Añadir librería .ZIP, donde será necesario abrir este fichero.
- De manera manual: se descarga el mismo fichero ZIP y se descomprime en la carpeta de librerías de Arduino y se reinicia el IDE (del Valle, s.f.).

Estas librerías se importan en el programa seleccionando la librería deseada, que se encuentra en la pestaña de Programa, entre las opciones que aparecen en la pestaña Incluir librería. De manera alternativa, se puede escribir #include, seguido del nombre de la librería del siguiente modo: #include <Adafruit_GFX.h>. En este trabajo, las librerías que se han empleado han sido: Adafruit_GFX Library, Adafruit_ST7735 and ST7789 Library, Adafruit BusIO, SPI y MFRC522.

- Para el control de la pantalla se han empleado las librerías Adafruit GFX Library y Adafruit ST7735 Library. La primera es la librería base que ofrece Adafruit para sus dispositivos. Contiene funciones básicas para hacer puntos y líneas, entre otras figuras básicas. Sin embargo, también es necesario descargar la segunda, ya que está creada específicamente para el hardware de la pantalla que se ha utilizado. Con la librería ST7735 vienen ejemplos que permiten familiarizarse con sus funciones y comprobar que la pantalla funciona correctamente. De manera directa en el código, solamente se ha empleado la ST7735 y las funciones que se han usado han sido para: inicializar el dispositivo, rellenar el fondo de un color, escribir texto o números, posicionar el cursor y modificar el color y tamaño del texto. Finalmente, la librería Adafruit BusIO es empleada por Adafruit_GFX Library.
- Para el control de los dos lectores RFID RC522 se han empleado la librería MFRC522 desarrollada por Miguel Balboa. Esta librería permite realizar la lectura y escritura en las tarjetas MIFARE S50 de 1k, comprobar el firmware del lector, cambiar el UID de las tarjetas (en particular, las tarjetas empleadas en este trabajo no admiten cambios de UID) y gestionar el chip select del bus SPI, entre otras funciones. En particular, en el código de este trabajo se ha empleado para introducir



la contraseña que habilita el acceso a la memoria de las tarjetas, para leer y escribir en ellas, para extraer el tamaño y bytes del UID, detectar la presencia de una tarjeta, leer número de serie de la tarjeta, indicar errores, detener la comunicación con las tarjetas y detener la encriptación.

- El bus SPI permite la comunicación del microcontrolador de Arduino con la pantalla y con los dos sensores RFID. Se ha gestionado con la librería SPI que tiene funciones como `begin()`, que configura los pines SCK, MOSI y SS como salidas o como `end()` para desactivar el bus. En el código se ha empleado para la inicialización del bus.

4.2.2. Inicialización

La primera instrucción es `Serial.begin(9600)` la cual establece la velocidad de transmisión de los datos en 9600 baudios por segundo. Esta función por defecto configura 1 bit de parada, ninguno de paridad y establece el tamaño de los datos en 8 bits.

Para realizar el segundo bloque del diagrama (Figura 40), se emplea la función `SPI.begin()` la cual configura los pines SCK, MOSI y SS como salidas digitales, apagando SCK y MOSI y encendiendo el pin SS.

Los lectores RFID se activan con la función `PCD_Init()`, perteneciente a la librería MFRC522 y los pines de los LED (41 y 43) se establecen como salidas digitales mientras que el del pulsador (39) se configura como entrada digital.

Finalmente, la pantalla se activa, se colorea el fondo de color negro, se posiciona el cursor en la esquina superior izquierda y se prepara el color del texto como blanco por defecto.

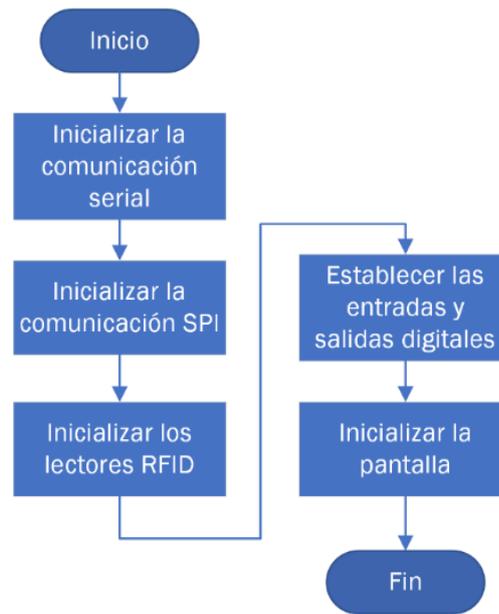


Figura 40. Inicialización

4.2.3. Bucle principal

Un sencillo bucle principal cambia el modo del sistema entre el modo lectura y escritura cuando se activa el pulsador. También, llama cíclicamente a las funciones de lectura o de escritura, según el modo en el que se encuentre. Si observamos el funcionamiento del código, veremos que el programa se inicia en modo lectura, por lo que estará continuamente verificando la entrada de alguna tarjeta en la zona de interrogación del lector de salida o de entrada para transmitir su información por el puerto serie. Al activar el pulsador, parpadeará el LED amarillo indicando que el dispositivo se encuentra en modo escritura y estará a la espera, en cualquiera de los dos sensores, de una tarjeta para escribir en ella la información que recibe a través del puerto serie. Si se vuelve a activar el pulsador, el dispositivo regresará al modo lectura y parpadeará el LED verde (Figura 41).

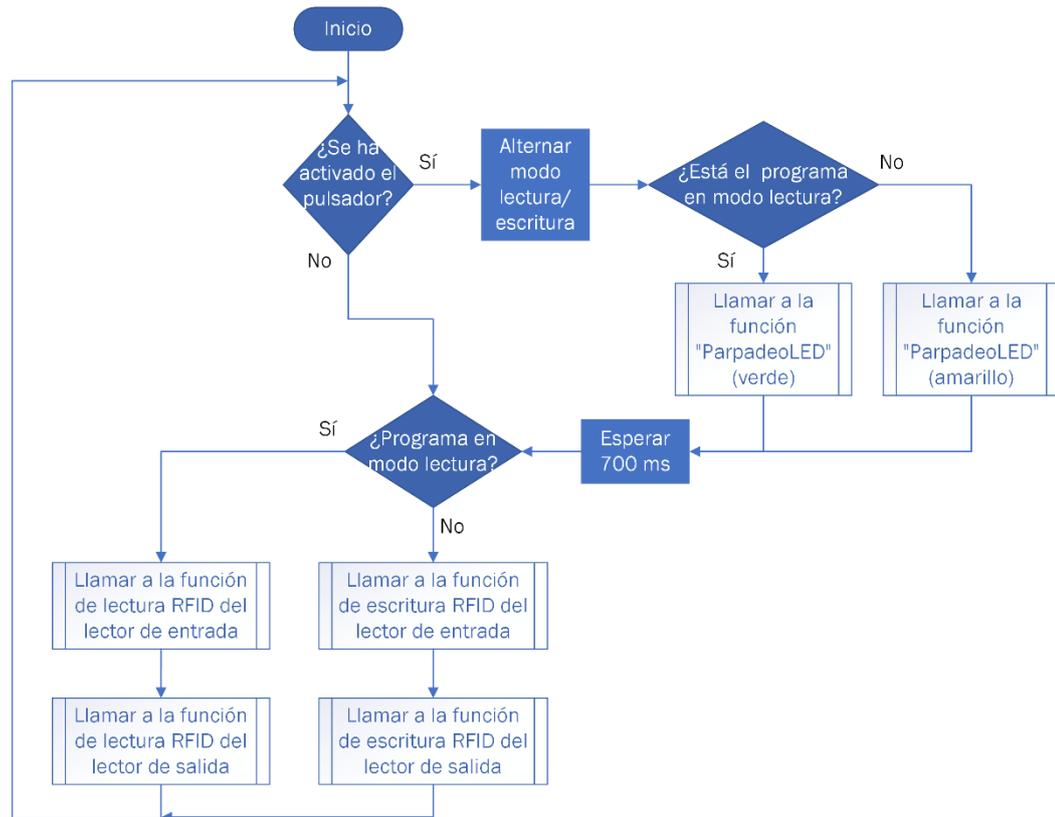


Figura 41. Bucle principal

4.2.4. Funciones Leer y Escribir

La función Leer (ver Figura 42) recibe un parámetro que es una instancia de la clase MFRC522, es decir, un objeto que está definido con su pin de selección de chip y con su pin de reset. Hay dos objetos, Entrada y Salida, que hacen referencia a sus respectivos lectores RFID. Este parámetro se empleará para controlar sobre qué lector RFID se realizan las operaciones como leer o escribir.

Esta función empieza comprobando si una tarjeta se encuentra en la zona de interrogación del lector. En caso de que no se detecte la tarjeta, se finaliza la ejecución de la función. Cuando se detecta un transpondedor el código comienza introduciendo la clave A para leer los contenidos del bloque 4, donde se almacena el tipo de producto y después se envía esta información por el puerto serie. Se procede enviando también el UID de la tarjeta y del mismo modo, se comunica la fecha por el puerto serie obteniéndola del mismo modo que el tipo de producto pero accediendo esta vez al bloque 1. Una vez finalizada la lectura se detiene la comunicación con la tarjeta y la encriptación. Por último, se comunica a través del puerto serie la entrada o la salida del palé y se llama a la función ParpadeoLED para indicar que el movimiento ha sido registrado correctamente. Después de cada envío de información se escribe una coma en



el puerto serie para separar cada tipo de dato y finalmente, cuando termina la función de lectura se envía un salto de página. Los datos se envían con este formato para que Excel los clasifique cada tipo de dato en una columna en concreto y cada lectura en una fila distinta.

La función Escribir (ver Figura 43) se encarga de trasladar los datos que se desean grabar y que se han recibido del libro de Excel a la tarjeta. Esta función, se desarrolla de un modo similar a la de lectura. Se comprueba la existencia de una tarjeta en las proximidades del lector, se introduce la clave para escribir en el bloque 1 de la memoria y se escribe en el mismo la fecha de caducidad. Del mismo modo, se escribe el tipo de producto en el bloque 4. Después de terminar la comunicación y la encriptación, parpadea el LED de salida o amarillo. Durante la escritura de la tarjeta, puede resultar llamativo que no se especifica en ningún momento la lectura del puerto serie para introducir los datos que se indican en el archivo Excel. Esto es debido a que los datos a escribir se encuentran en unas variables globales que son actualizadas cada vez que Excel comunica información a través del puerto serie. De este modo se escribe en las tarjetas la información más reciente introducida por el usuario. La actualización de las variables canal_1 (para el tipo de producto) y canal_2 (para la fecha) es realizada por la función ExtraerDatosExcel, la cual es llamada por la función serialEvent, que se ejecuta cada vez que se recibe información a través del puerto serie.

Además, para borrar de la memoria de la tarjeta los bytes que puedan quedar de escrituras anteriores, los bytes de las variables canal_1 y canal_2 son copiados en un buffer y se rellenan los bytes restantes con espacios, hasta completar los 16 bytes que tiene un bloque de datos. De este modo se eliminan los bytes que hayan podido quedar de la escritura previa en caso de que el tamaño del dato anterior fuese más grande que el del más reciente. Por otro lado, en caso de que cualquiera de las operaciones de lectura, escritura o de introducción de claves no se completen debidamente se enviará un mensaje de error a través del puerto serie.

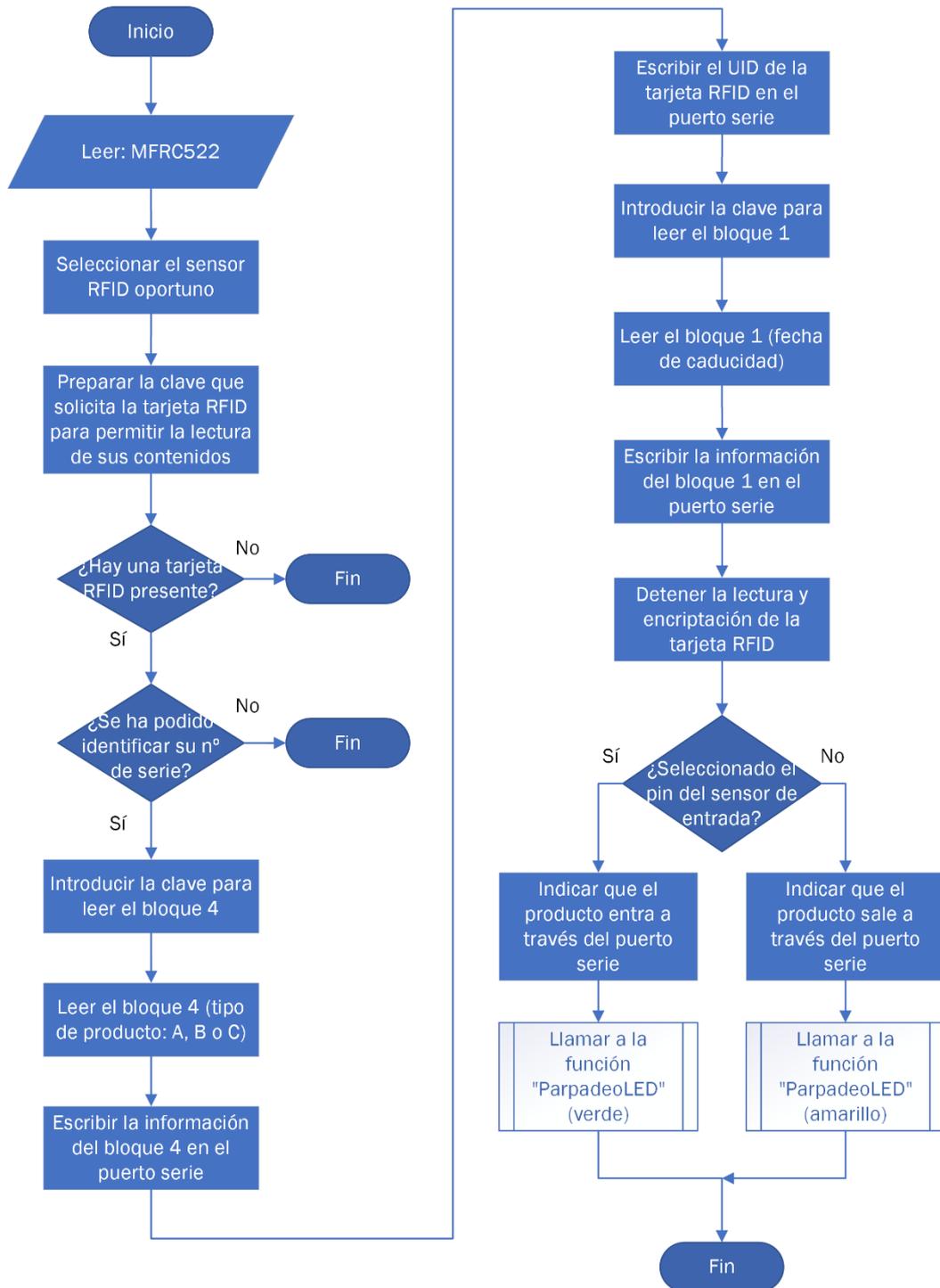


Figura 42. Función Leer

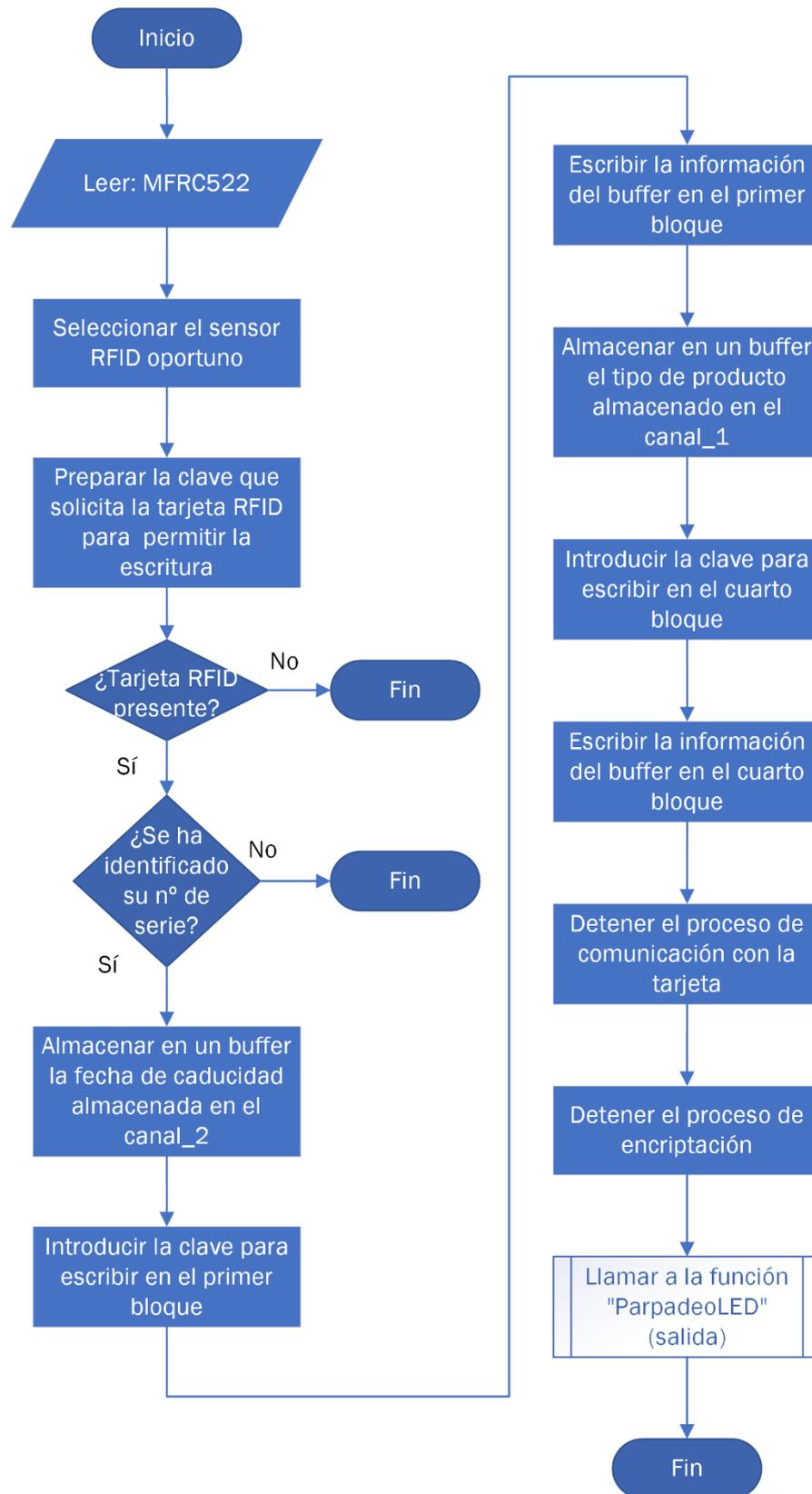


Figura 43. Función Escribir

4.2.5. Función ExtraerDatosExcel

Para extraer los datos provenientes de la hoja Excel se emplea la función Serial.readBytes que almacena los 21 bytes que llegan en la variable datos_Excel. El tamaño de los datos siempre es de 21 bytes. Después, la variable datos_Excel se separa en las variables canal_1, canal_2 y canal_3.

Desde Excel se envía cada dato a través de un canal, tal y como se muestra en la Tabla 10. A través del puerto serie cada canal está delimitado por una coma y es de este modo como la función es capaz de reconocer y separar la variable datos_Excel en una variable por canal (ver Figura 44). El canal 1 y 2 se emplean para la escritura de la tarjeta y el 3 para mostrar las coordenadas del producto por la pantalla TFT. En el canal 3 (ver Tabla 10), el primer dígito (2) indica la posición “x” en el almacén, los dos dígitos siguientes (06) la posición “y” y el último (1) es la altura o nivel de la estantería.

CH1	CH2	CH3
A	09/02/2026	2061

Tabla 10. Canales de salida del transmisor de datos de Excel

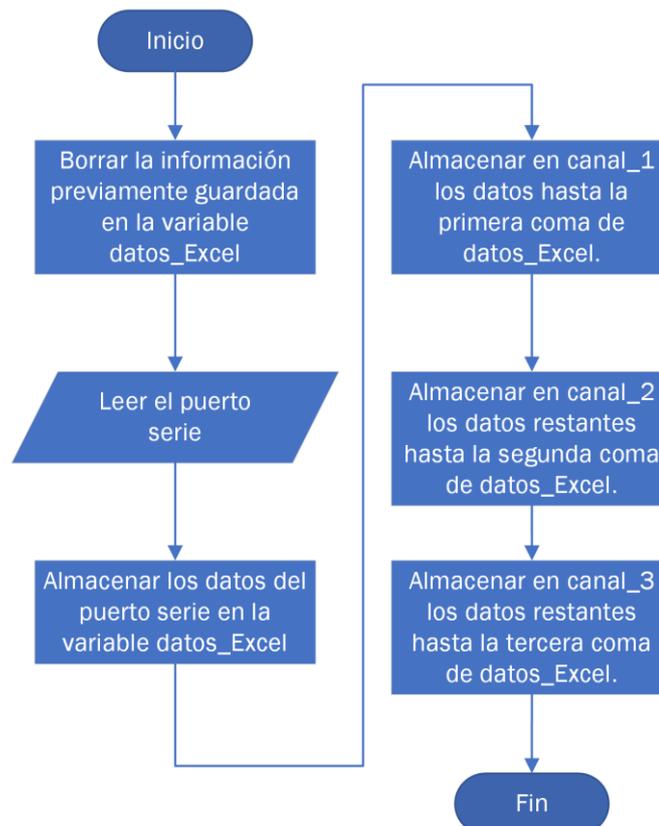


Figura 44. Función ExtraerDatosExcel



4.2.6. Funciones serialEvent, IndicarUbicación y ParpadeoLED

En primer lugar, serialEvent se ejecuta cada vez que llega información al puerto serie. Siempre llama a la función ExtraerDatosExcel para mantener las variables canal_1 y canal_2 actualizadas, en caso de que el usuario desee escribir información en una tarjeta. Esta función también comprueba si los contenidos del canal 3 varían, lo cual implica la entrada de un nuevo palé, llamará a la función IndicarUbicación para actualizar la dirección de la pantalla (ver Figura 45).

Por otra parte, la función IndicarUbicación a partir de la variable canal_3, separa sus contenidos para escribir en la pantalla la posición en el espacio mediante coordenadas cartesianas. Cada coordenada es escrita en una fila. En función de si el producto que entra es A (rojo), B (verde) o C (azul), cambiará el color del texto (ver Figura 46).

Finalmente, ParpadeoLED enciende y apaga 3 veces consecutivas el led seleccionado (ver Figura 47).

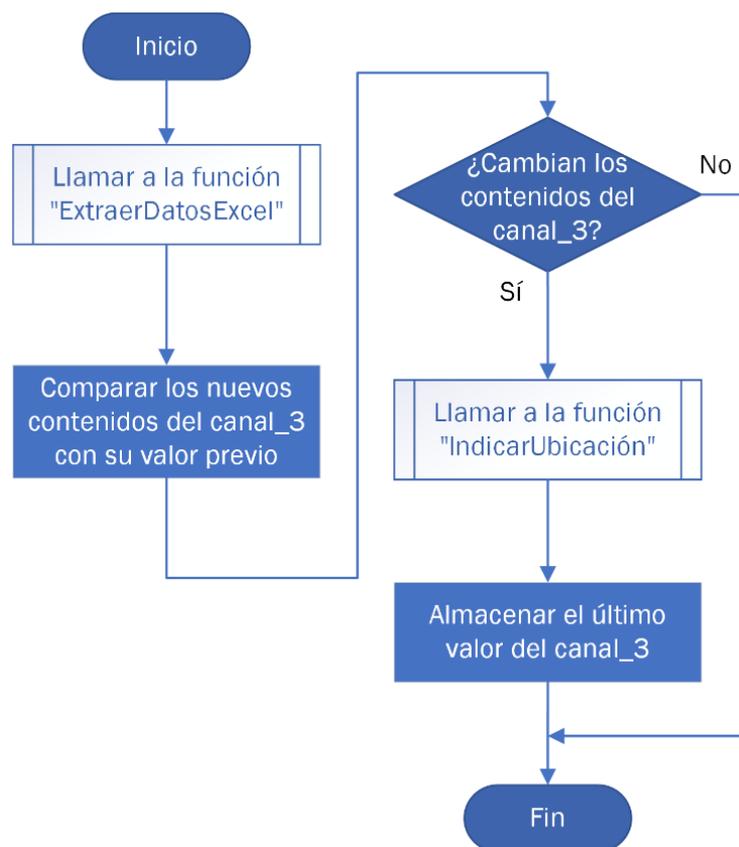


Figura 45. Función serialEvent

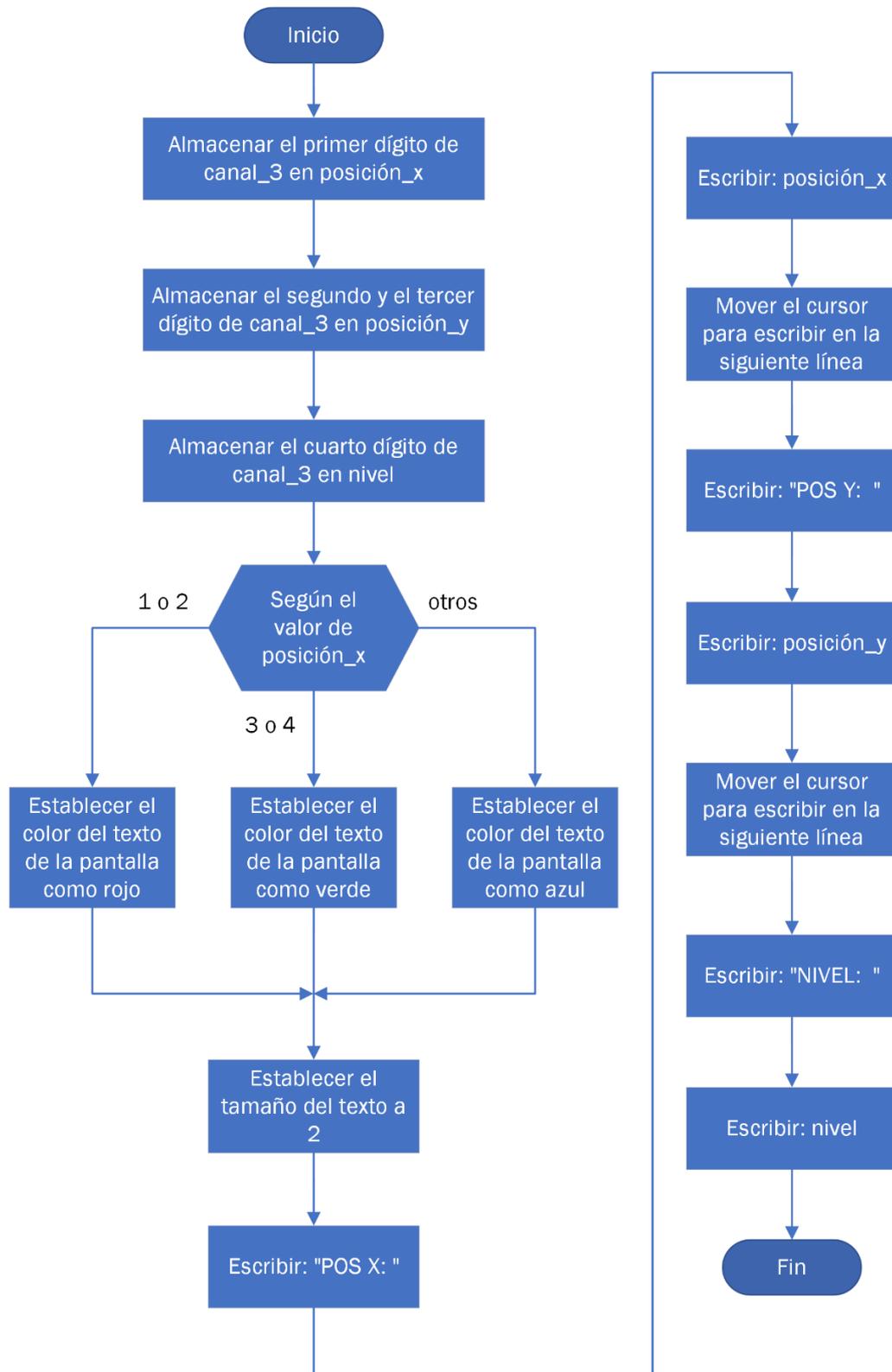


Figura 46. Función IndicarUbicación

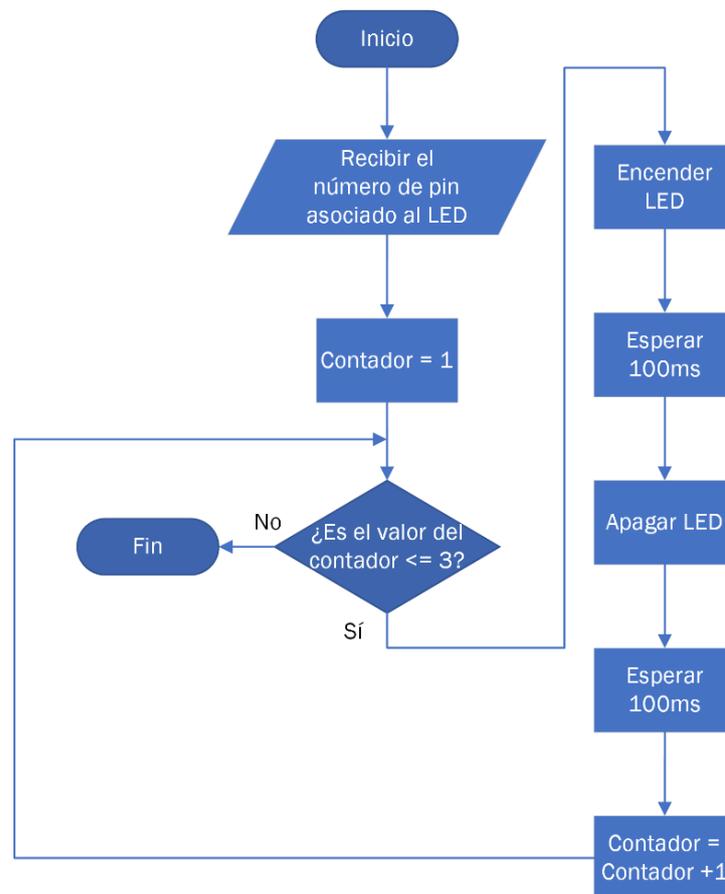


Figura 47. Función ParpadeoLED

4.3. Guía de uso de la aplicación de Excel

El sistema Arduino-RFID requiere de una interfaz con el usuario para poder llevar a cabo el control del almacén. Un archivo Excel se encargará de almacenar, procesar, organizar y presentar toda la información recibida. Se tendrá constancia de todos los movimientos realizados en el almacén identificando cada producto por separado con su información única. Luego, serán clasificados en entradas y salidas, conociendo también, los productos que se encuentran actualmente en el almacén. El procesamiento de datos es sencillo, limitándose principalmente al cálculo de las coordenadas y ubicaciones que debe ocupar un producto entrante o de donde se debe obtener un palé saliente en función de si el sistema opera en modo FIFO o LIFO. Las entradas y las salidas serán agrupadas en lotes y se presentarán gráficas, KPIs y tablas para poder conocer el estado y la evolución del almacén junto con su desempeño.



Por otro lado, en apartados posteriores también se mencionarán los términos coordenadas y ubicación. Las coordenadas se refieren a la longitud (x), profundidad (y) y altura (z) a la que se encuentra un palé en stock. Mientras que la ubicación es el número que surge del conteo de los espacios de almacenamiento de una estantería (por ejemplo, la ubicación 1 de la isla A se encuentra en las coordenadas $x = 1$, $y = 1$ y $z = 1$, ver Figura 48).

El archivo de Excel consta de las siguientes hojas y el funcionamiento de cada una de ellas se explicará a continuación:

1. Cuadro de control
2. Planificación y resultados
3. Inventario diario
4. Entradas
5. Almacenado
6. Salidas
7. Nivel 1 del almacén
8. Nivel 2 del almacén
9. Nivel 3 del almacén
10. Datos
11. Hojas del transmisor de datos

4.3.1. Hoja Cuadro de control

4.3.1.1. Selección del método de gestión FIFO o LIFO

Esta hoja contiene cuatro tablas. La primera, empezando por la parte superior de la hoja, contiene un menú desplegable donde se puede elegir el método de gestión FIFO o LIFO. El método elegido se aplicará a todos los productos. Una vez que se elige el método de gestión y comienza la entrada de productos, no se puede modificar esta elección.

4.3.1.2. Información básica sobre el almacén

Después, la segunda tabla muestra, de cada tipo de producto, la cantidad almacenada, la capacidad restante y el stock de seguridad. El stock de seguridad es el único campo que puede ser modificado por el usuario. A partir de esta tabla se obtiene la primera gráfica que muestra el inventario de cada producto y la segunda gráfica que indica la proporción de productos.



4.3.1.3. Escritura de las tarjetas RFID

La tercera tabla contiene dos campos que deben ser rellenados por el usuario cuando se desee escribir en la tarjeta. El primer campo está destinado al tipo de producto y el segundo a la fecha de caducidad. Para escribir en la tarjeta, solamente es necesario escribir ambos datos, activar el pulsador de cambio de modo en el prototipo Arduino y el LED amarillo parpadeará. A continuación, se deberá colocar la tarjeta en cualquiera de los dos lectores RFID hasta que vuelva a parpadear dicho LED. Una vez completado este proceso se puede volver a activar el pulsador para seguir leyendo la entrada y salida de productos o se puede continuar con la escritura de más tarjetas.

4.3.1.4. Evaluación del rendimiento del almacén

La última tabla muestra 14 KPI para analizar el desempeño del almacén e identificar áreas de mejora, desde proveedores poco eficaces, hasta errores del personal a la hora de realizar envíos. Los KPI son de varios tipos:

- De rotación de inventario para el periodo actual de cada tipo de producto.
- Rupturas de stock.
- La ocupación porcentual del almacén.
- Proporción de devoluciones/envíos.
- Recepciones incorrectas dependiendo del tipo de producto.
- Envíos incorrectos.
- Recepciones tardías dependiendo del tipo de producto.
- Disponibilidad para la entrega.

Finalmente, la tercera gráfica muestra la evolución del inventario diario, es decir, la cantidad de productos de cada tipo en stock al final de cada día. Los datos empleados para la construcción de esta gráfica se encuentran en la hoja de Inventario diario.

4.3.2. Hoja Planificación y resultados

A continuación, en la hoja de Planificación y resultados se realizará la planificación de los envíos y recepciones que tendrán lugar en el almacén. Planificar supone registrar los movimientos (de entrada o salida) que se ejecutarán en el almacén en el futuro cercano.

Cada movimiento que se planifica, engloba una cantidad de palés y tiene información asociada. Se considera como un movimiento la llegada de palés del mismo tipo, a una hora similar. La planificación requiere indicar la fecha, si



el movimiento es de entrada o de salida, el tipo de producto y la cantidad planificada de palés, que podrá no coincidir con la cantidad real. En otra columna adyacente se mostrará la diferencia entre las cantidades reales y planificadas. Además, se debe de indicar también, una hora de llegada o de salida que será comparada con la hora real, considerada como la hora del último palé del lote o movimiento en entrar o salir.

De este modo, en otra columna se mostrarán los retrasos o adelantos que surjan. También hay una columna dedicada a las devoluciones, en la que se tendrá que escribir la cantidad de palés que se han devuelto. Para completar las columnas de cantidades y horas reales será necesario presionar el botón de “Actualizar”, situado a la izquierda del encabezado de la tabla. Es necesario haber planificado todos los movimientos antes de actualizar la tabla.

Si hubiese un envío que no se puede realizar por falta de existencias, en lugar de “Salir” se deberá escribir “Ruptura de stock” antes de actualizar la hoja. También será necesario dejar su fecha, tipo y cantidad.

Con el propósito de facilitar la tarea de planificación, se han destinado otras 3 columnas, llamadas Proyección A, B y C que indican la cantidad de productos que estarán en existencia si se cumple la planificación. Si no se cumpliera, las diferencias de cantidades se ajustarán de manera automática.

Finalmente, en las últimas 4 columnas se realiza el conteo de envíos y recepciones que no cumplan con las cantidades de palés planificadas o cuya llegada o salida del almacén se haya retrasado 15 minutos o más. Estas columnas junto con las de diferencia de cantidades, diferencias horarias y devoluciones son la base para el cálculo de los KPIs. Los retrasos en las entradas discriminan entre tipos de productos, ya que cada uno de ellos se obtiene de un proveedor diferente. Así, es posible evaluar el desempeño de cada uno de ellos. Por otro lado, solamente se cuentan las veces que se han retrasado las salidas, ya que son los mismos empleados los que gestionan los tres tipos de productos.

Además, se pueden aplicar filtros para obtener de manera más sencilla información como cuántas devoluciones se han registrado en el último mes, por ejemplo. Sin embargo, es muy importante realizar la planificación y pulsar el botón de actualizar sin aplicar filtros.

Los campos que se deben rellenar de forma manual son aquellos que presentan un color verde más oscuro de fondo, el resto se rellenarán automáticamente o al presionar el botón de Actualizar (columnas de cantidad real y hora real) una vez se hayan realizado los movimientos planificados. Además, es esencial introducir cada movimiento, de entrada o de salida en



orden cronológico, de lo contrario la actualización no se realizará correctamente. En la Tabla 11, Tabla 12, Tabla 13 y Tabla 14 se muestra un ejemplo de planificación con los resultados para un movimiento.

Fecha	Movimiento	Producto	Cantidad Planificada	Cantidad real
08/05/2024	Entrar	C	2	2

Tabla 11. Ejemplo de planificación de un movimiento. Parte 1

Diferencia de cantidades	Hora planificada	Hora real	Retraso	Devolución
0	4:00:00 PM	4:20:10 PM	20min	

Tabla 12. Ejemplo de planificación de un movimiento. Parte 2

Proyección A	Proyección B	Proyección C	Cuenta retraso salidas	Envíos incorrectos
0	0	2		

Tabla 13. Ejemplo de planificación de un movimiento. Parte 3

Recepciones incorrectas	Cuenta retraso en las entradas >15min
	C

Tabla 14. Ejemplo de planificación de un movimiento. Parte 4

4.3.3. Hoja Inventario diario, Entradas, Almacenado y Salidas

En “Inventario diario” se registra la cantidad de producto A, B o C almacenado al finalizar un día de trabajo. Este se actualiza al actualizar la hoja anterior, de Planificación y resultados, aunque también existe la posibilidad de actualizarla con el botón situado a la izquierda del encabezado de la tabla (Tabla 15).

INVENTARIO DIARIO				
Fecha	Producto A	Producto B	Producto C	
30/04/2024	4	0	2	
01/05/2024	21	10	2	
02/05/2024	21	6	2	

Tabla 15. Ejemplo de inventario diario

Las hojas de Entradas, Almacenado y Salidas simplemente contienen celdas que referencian la información de las tablas de Entradas, Almacenado y Salidas de la hoja de datos. Esta información incluye el tiempo de llegada o salida, tipo de producto, código SKU, fecha de caducidad, coordenadas y ubicación. Puede resultar necesario filtrar los datos de los productos que han estado bajo la gestión del almacén en algún momento y para poder aplicar filtros diferentes a cada tabla, se han separado en hojas distintas.



4.3.4. Hojas de almacén (Niveles 1, 2 y 3)

Cada nivel se representa en una hoja diferente y cada tipo de producto se almacena en una isla. Estas 3 hojas pretenden ofrecer de manera rápida e intuitiva información como la ocupación del almacén, cuáles serán las siguientes ubicaciones a ocupar y a liberar y en qué zonas de la estantería hay productos o no. Observando estas hojas en tiempo real, según se reciben y envían pedidos se puede, se apreciarán los métodos FIFO o LIFO.

- FIFO: los productos comienzan a almacenarse desde la posición número 1. Los productos que abandonan el almacén son los primeros que llegaron. Por lo tanto, supongamos para el escenario presentado en la Figura 48, se debe enviar un palé del tipo A. En este caso le único palé que se moverá será el de la posición número 1. Si hubiese más salidas, se enviarían los pales situados en las ubicaciones 2, 3, 4 y así sucesivamente. Entonces, el resto de palés permanecerán en su sitio y con el paso del tiempo se recibirán palés hasta que se llegue a la ubicación 90. A partir de la cual, si hay espacio libre se continuará el ciclo desde la ubicación número 1.
- LIFO: al igual que en el método anterior, los productos comienzan ocupando las primeras posiciones desde la ubicación 1. Sin embargo, en este caso, los últimos palés que se reciben son los primeros en ser expedidos. Por este motivo, el palé situado en la ubicación número 1 será siempre el último en ser enviado, de tal forma que se podrán almacenar productos hasta ocupar la ubicación 90 de cada estantería.

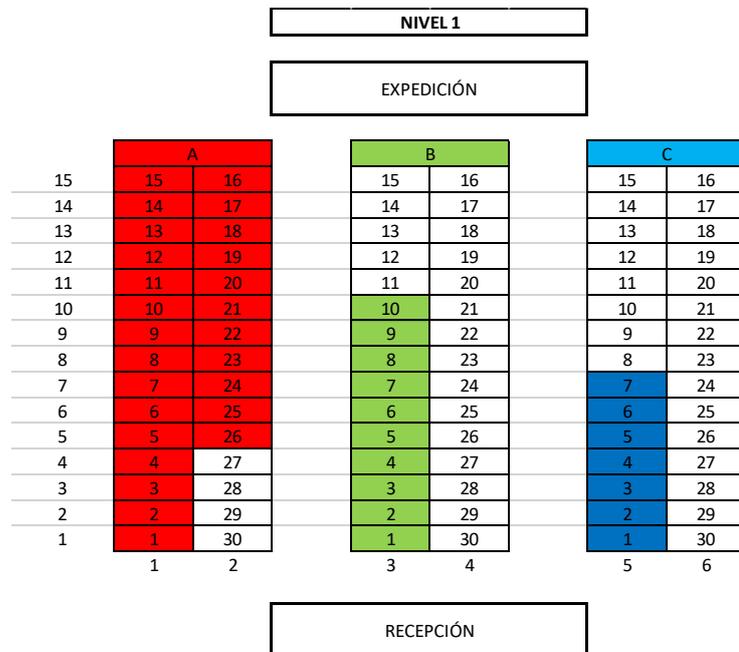


Figura 48. Hoja "Almacén-N1", ejemplo LIFO



4.3.5. Hoja de Datos

En primer lugar, la primera tabla contiene la información recibida desde el dispositivo Arduino. La segunda tabla, procesa estos datos para obtener de cada tipo de producto la ubicación máxima y mínima ocupada en su estantería, las cuales se emplean para resaltar con color rojo, verde o azul las ubicaciones ocupadas de las hojas de Almacén-Nx. Para el ejemplo anterior sería la 26 y la 1, respectivamente para el tipo de producto A. También, en esta misma tabla se hallan las coordenadas de los productos que entran y que salen. Ambas se utilizan para rellenar las coordenadas de las tablas de Entradas y Salidas, situadas en la misma hoja. Además, las coordenadas de los productos que entran son enviadas a Arduino por la subrutina LugarAlmacén.

Después, se rellenan las tablas Entradas (ver Tabla 16), Almacenado (ver Tabla 17) y Salidas (ver Tabla 18), automáticamente con las macros que se verán en apartados posteriores: CopiarDatosEntrantes, CopiarDirección y con la subrutina ActualizarInventario. Entre otras utilidades de estas tablas, destaca la capacidad básica y fundamental de la aplicación de Excel para realizar el control del inventario a lo largo de varios días. Cuando se desconecta el dispositivo Arduino del transmisor de datos, ya sea porque se desconecta el cable, porque se cierra el libro de Excel o porque se apaga el equipo, si los datos no son copiados de la entrada de datos a otra hoja, se perderán. Por este motivo la subrutina CopiarDatosEntrantes es de especial importancia.

ENTRADAS					A	B	C
					33	17	7
Tiempo	Tipo de producto	SKU	Fecha de caducidad	x	y	Nivel	Ubicación
08/05/2024 16:20	C	0001	15/11/2025	5	1	1	1
08/05/2024 16:20	C	0002	15/11/2025	5	2	1	2
08/05/2024 16:20	A	0003	15/11/2025	1	1	1	1

Tabla 16. Entradas

ALMACENADO						
Tipo de producto	SKU	Fecha de caducidad	x	y	Nivel	Posición
C	0001	15/11/2025	5	1	1	1
C	0002	15/11/2025	5	2	1	2
A	0003	15/11/2025	1	1	1	1

Tabla 17. Productos almacenados

SALIDAS							
Tiempo	Tipo de producto	SKU	Fecha de caducidad	x	y	Nivel	Ubicación
08/05/2024 16:21	A	0010	15/11/2025	1	8	1	8
08/05/2024 16:21	A	0009	15/11/2025	1	7	1	7
08/05/2024 16:21	A	0008	15/11/2025	1	6	1	6

Tabla 18. Productos enviados o salidas



4.3.6. Hojas Entrada de datos, Salida de datos y Configuración

Estas dos hojas forman parte de un complemento de Excel que puede activarse desde el menú Archivo, seleccionando después Opciones y luego Complementos. En Complementos se debe elegir la opción Complementos COM del menú desplegable y seleccionar la opción de “Microsoft Data Streamer for Excel”, también llamado Transmisor de datos. El cual debe de conectarse con Arduino e iniciarse para poder leer y escribir en las tarjetas y para poder comunicarse con la pantalla. En la hoja de entrada de datos se sitúa la tabla que muestra la información recibida por el puerto serie dividida en diferentes canales. Junto a cada fila de información, también se proporciona el tiempo de llegada.

El transmisor de datos escribe en el primer canal la cadena de texto que lee del puerto serie hasta que detecta una coma, la siguiente cadena de texto hasta la segunda coma la escribe en el segundo canal y lo mismo sucede con el resto de canales. Para indicar que se ha finalizado el envío de información con el propósito de que la siguiente cadena de texto sea escrita en otra fila, se deberá enviar a través del puerto serie un salto de línea.

Por otro lado, se enviarán de vuelta al dispositivo Arduino los datos que se escriban en los canales de la hoja de Salida de datos y la frecuencia de envío se puede modificar en la hoja de configuración. El envío de datos se realiza de un modo similar a la recepción. Después de enviar la cadena de texto escrita en un canal, se envía además, una coma por el puerto serie. Esto sucede tantas veces como canales se hayan configurado. Finalmente, se envía un salto de línea y se repite el ciclo.

Finalmente, en la hoja de configuración del transmisor de datos se ha establecido el intervalo de envío de datos en 150 ms, 1000 filas de datos, con una orientación de datos en la que el dato más reciente se muestra primero y para la entrada de datos se emplean 4 canales y para la salida 3.

4.4. Excel VBA

El lenguaje de programación Visual Basic for Applications (VBA) desarrollado por Microsoft para sus productos, como puede ser Excel, se emplea en este trabajo para automatizar tareas repetitivas como la clasificación de la información de los productos que entran, que se encuentran en stock y que salen en sus respectivas columnas, enviar información a Arduino, hallar el inventario diario, escribir las horas de llegada y de salida, ajustar gráficas y borrar datos, entre otras.



Este apartado comienza con la explicación del código principal, escrito en la hoja Entrada de datos, el cual llama a la mayoría de las subrutinas de este archivo Excel. Después, se analizará el funcionamiento del resto de las subrutinas que son las siguientes: CopiarDatosEntrantes, CopiarDirección, ActualizarInventario, LugarAlmacén, RecepciónDeLotesDePalés, EnvíoDeLotesDePalés, InventarioDiario, AjustarEjeXGráfica y EliminarDatos.

4.4.1. Código de la hoja Entrada de datos

Cuando el transmisor de datos de Excel recibe del dispositivo Arduino la información sobre la entrada o salida de un palé, el código de la hoja Entrada de datos, se ejecuta automáticamente verificando que si el producto entra, exista espacio suficiente en el almacén para albergarlo o que si sale, que existan palés suficientes de ese tipo para ser enviados, de lo contrario, no registrará dicho movimiento del palé. Para los movimientos de entrada que estén permitidos se ejecutarán las subrutinas CopiarDatosEntrantes, CopiarDirección, LugarAlmacén y ActualizarInventario, en este orden, las cuales se explicarán en los siguientes apartados. Para los movimientos de salida autorizados solamente se llamará a ActualizarInventario (ver Figura 49).

Este sistema de control solamente considera que existen los movimientos registrados en las hojas Entradas y Salidas. En caso de que los movimientos no cumplan las condiciones previamente mencionadas, no se registrarán en estas hojas.

Por otra parte, la ejecución automática del código con la llegada o salida de unidades de almacenamiento se consigue con una subrutina particular en conjunto con una instrucción condicional:

- “Private Sub Worksheet_Change(ByVal Target As Range)”
- “If Not Application.Intersect(KeyCells, Range(Target.Address)) Is Nothing Then”

La variable KeyCells contiene el rango de celdas que queremos observar si su contenido varía. Por otro lado, Target es la variable que contiene el rango de las celdas cuyos contenidos cambian en la hoja. Entonces, si queremos saber cuándo se produce un nuevo movimiento debemos comprobar (con la instrucción condicional) si Target y KeyCells coinciden. En el rango A8:F8 de Entrada de datos aparecen todos los datos del palé más reciente que ha entrado o salido. A8 contiene el tiempo en el que se produce el movimiento con una precisión de centésimas de segundo, por lo que será suficiente comprobar solo esta celda.

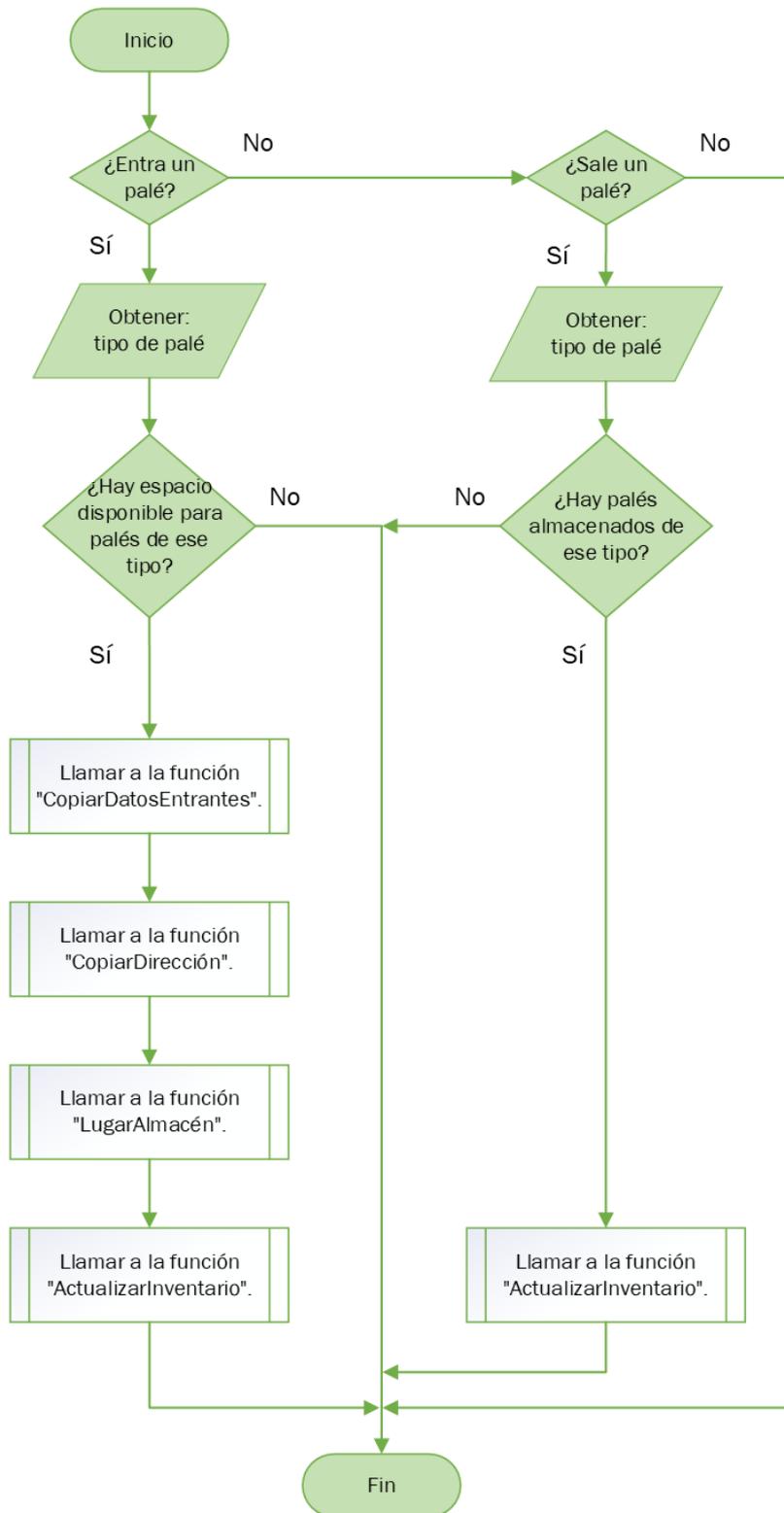


Figura 49. Subrutina de la hoja Entrada de datos

4.4.2. Subrutina CopiarDatosEntrantes

Si se llama a esta subrutina, ha entrado un palé. En la hoja Entrada de datos, el rango A8:E8 muestra los datos más recientes, con la información de la tarjeta RFID y el tiempo de entrada. Entonces, se copia este rango para ser pegado en la columna de entradas de la hoja Datos. Además, se identifica inequívocamente a cada palé asignando un código (SKU) a cada uno que se halla calculando el total de productos que han entrado al almacén (ver Figura 50).

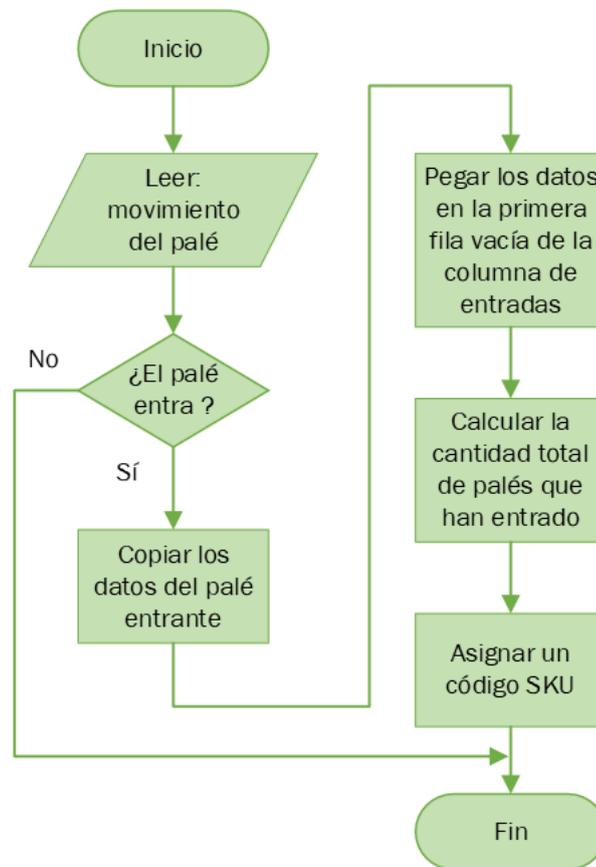


Figura 50. Subrutina CopiarDatosEntrantes

4.4.3. Subrutina CopiarDirección

Como se ha visto previamente, esta es la segunda subrutina llamada por el código de la hoja 4 al detectar la entrada de algún palé. Entonces, se averigua de qué tipo es, para buscar las coordenadas que le corresponden en el almacén. En la hoja "Datos" existen 3 tablas (para A, B y C) que, dependiendo de si se ha seleccionado FIFO o LIFO en la hoja Cuadro de control, calculan



dónde deben almacenados o de dónde se deben obtener los palés. Este cálculo ofrece por separado la fila, columna y nivel designados para el palé. Además, también indica su ubicación en la estantería. Son estos cuatro datos los que se copian y pegan en la columna de entradas, en misma fila donde la función anterior escribió los datos de ese mismo palé (ver Tabla 16 y Figura 51).

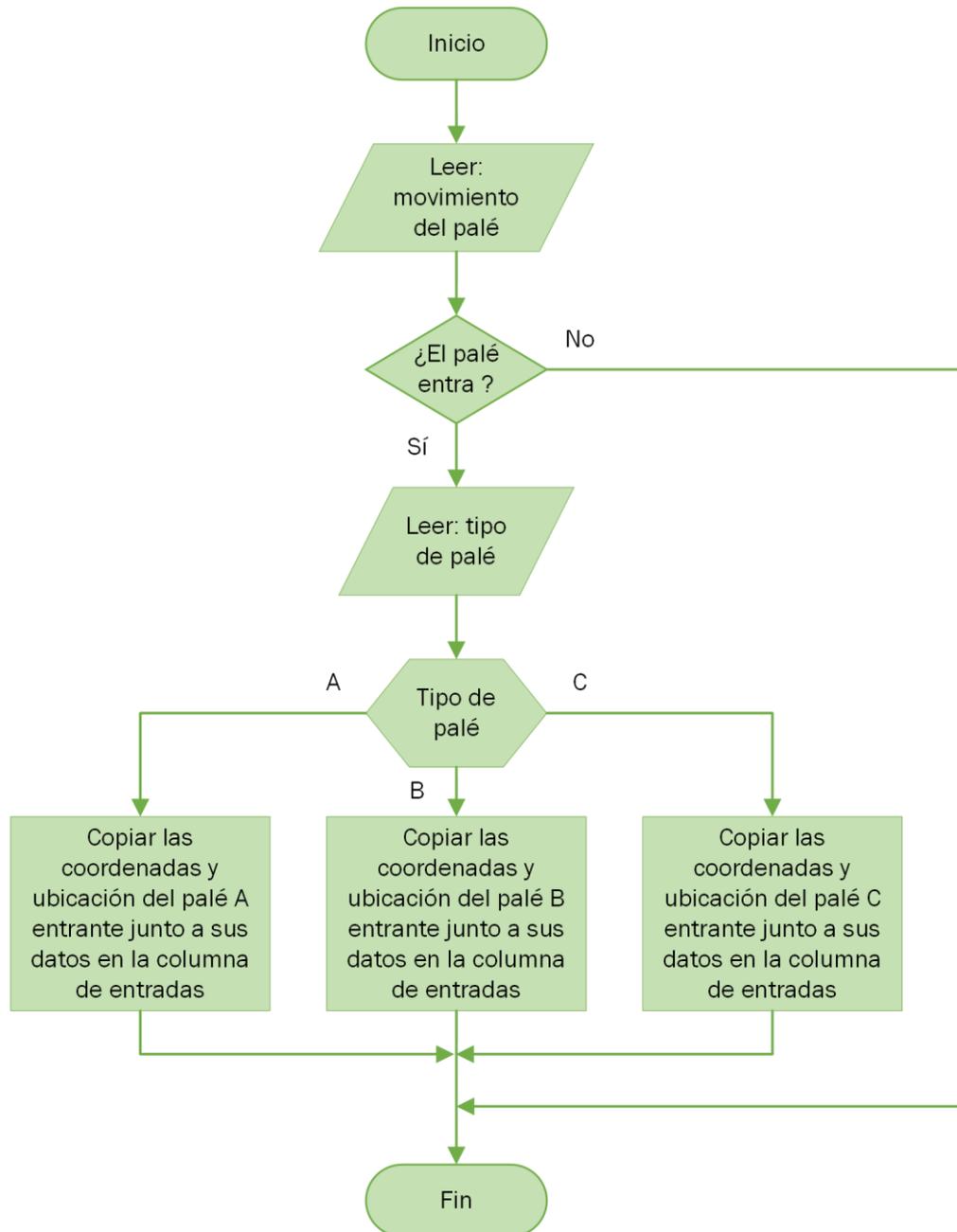


Figura 51. Subrutina CopiarDirección



4.4.4. Subrutinas ActualizarInventario y LugarAlmacén

En primer lugar, el funcionamiento de ActualizarInventario es el siguiente. Al entrar un producto nuevo, se ejecutan en primer lugar las subrutinas que copian los datos del producto en la tabla de entradas. Una vez se ha finalizado este proceso, sencillamente se copia la nueva entrada en la tabla Almacenado. Cuando sale un producto, dejará vacante su ubicación (90 ubicaciones por tipo de producto), la cual es calculada y posteriormente buscada en la columna de productos almacenados. Cuando se localiza la fila que contiene los datos del producto saliente, estos se copian en la primera fila en blanco de la tabla de salidas y se borran los datos de la tabla “almacenado”. Además, será necesario añadir en la tabla de salidas la fecha y hora en la que se produjo el movimiento. Finalmente, se cortan todos los datos de la tabla de productos almacenados situados en filas inferiores a la que se acaba de borrar y se pegan en la fila en blanco para eliminar el espacio (ver Figura 52).

Por otro lado, la macro LugarAlmacén (ver Figura 53) busca las coordenadas en el almacén que le corresponde al producto entrante, teniendo en cuenta su tipo (A, B o C). En la hoja de Datos esta información se presenta por separado, como se ha visto en la subrutina de CopiarDirección. No obstante, se usa la información de otra celda en la que se presentan estas coordenadas de forma aglutinada. Esto se debe a que la ubicación debe de ser enviada por un único canal (canal 3 del transmisor de datos). Finalmente, se escribe la ubicación en la celda C5 de Salida de datos (ver canal 3 de la Tabla 10).

Es cierto que, modificando el código de Arduino, se podrían transmitir las coordenadas a través de diferentes canales, sin embargo, esto no supondría una diferencia significativa en la complejidad o eficiencia del código.

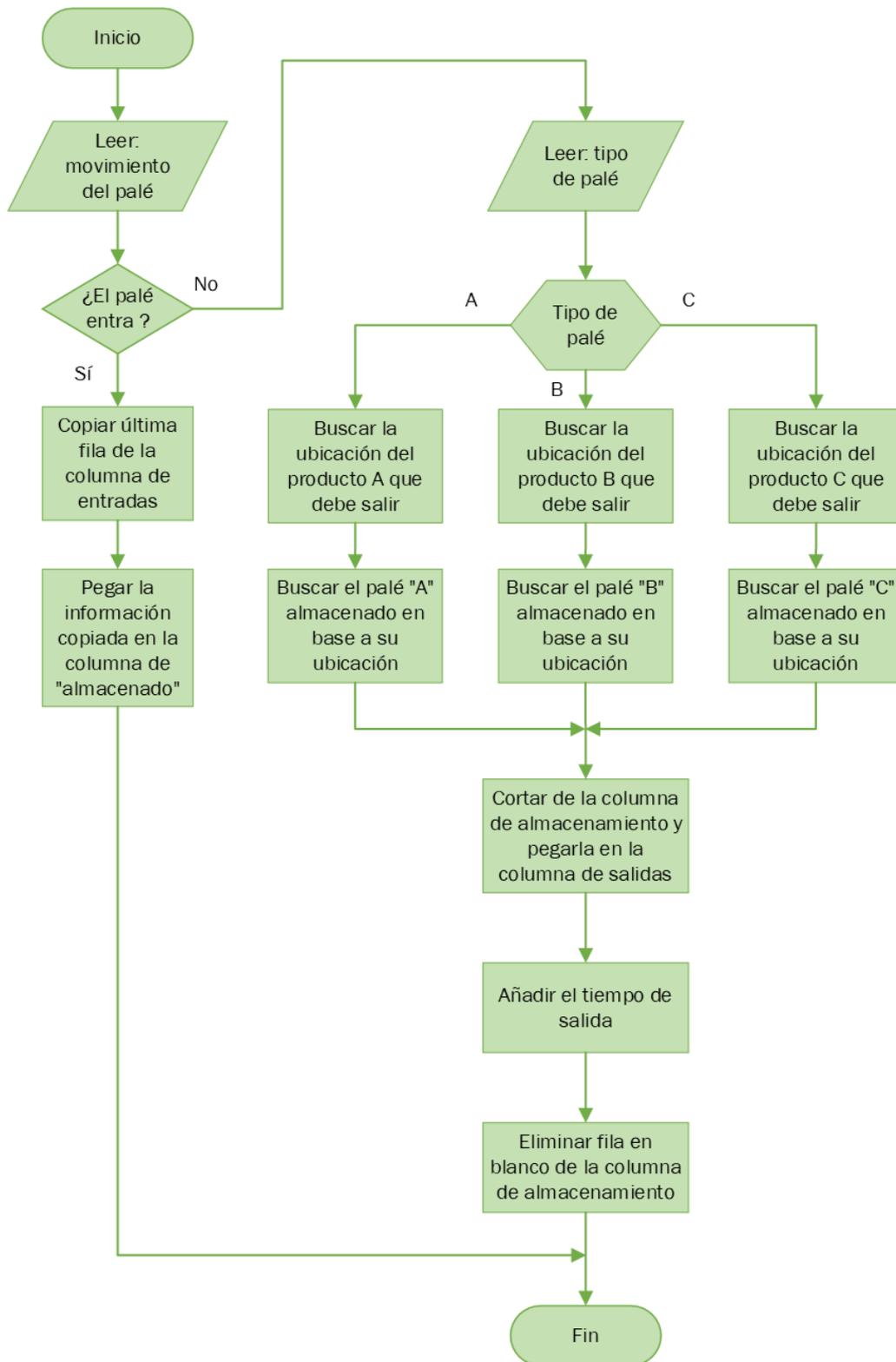


Figura 52. Subrutina ActualizarInventario

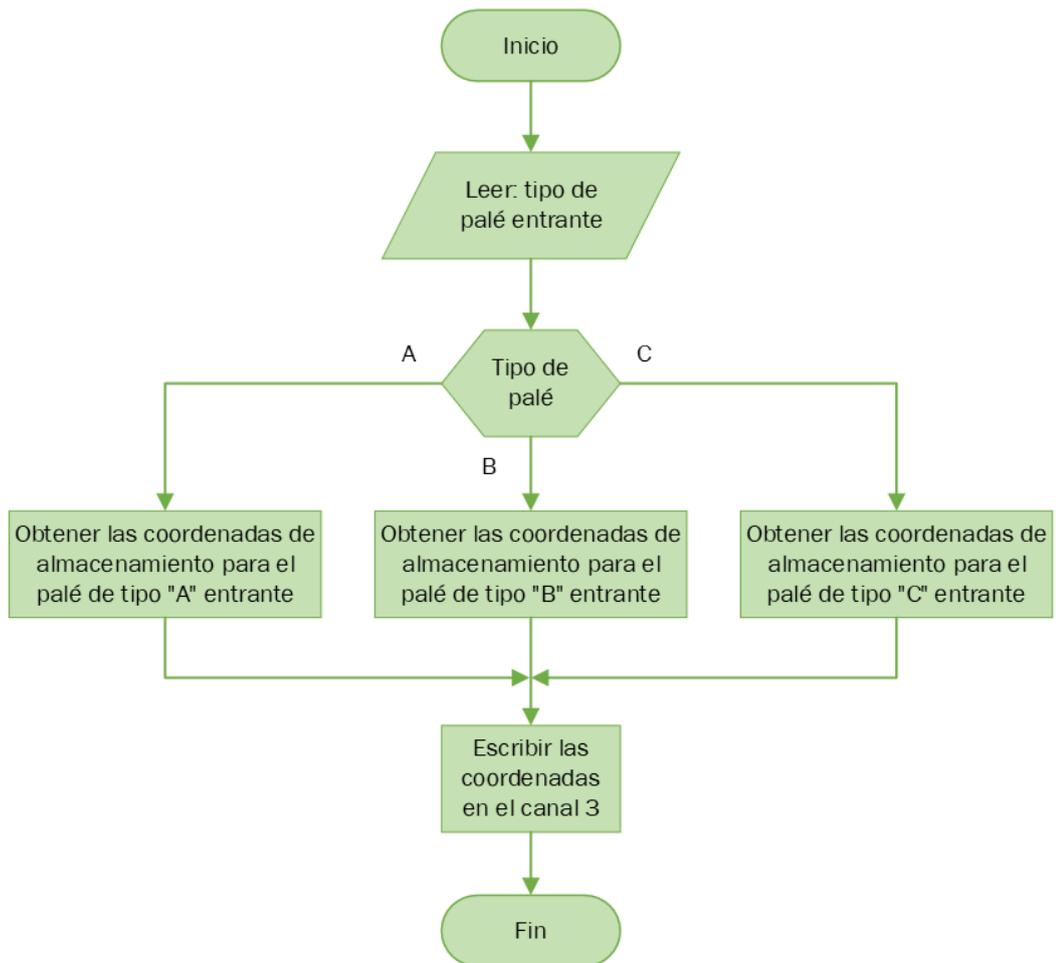


Figura 53. Subrutina LugarAlmacén



4.4.5. RecepciónDeLotesDePalés y EnvíoDeLotesDePalés

Antes de utilizar el sistema de control de inventario desarrollado en este trabajo, es importante conocer el funcionamiento de estas dos subrutinas y la hoja de Planificación y resultados, la cual ha sido explicada en la guía de uso.

En primer lugar, la macro RecepciónDeLotesDePalés (ver Figura 54) considera un lote de palés al conjunto de palés del mismo tipo de productos consecutivos (A, B o C) que hayan entrado o salido del almacén en un intervalo de tiempo similar. Es decir, la subrutina empieza contando el número de palés desde la parte superior de la tabla de Entradas y cada vez que el tipo de palés varía o el salto temporal de las entradas es superior a una hora comienza a contar otra vez, pero considerándolo como un segundo lote, y así sucesivamente.

Después, en la hoja de Planificación y resultados, se considera a cada fila como un lote o movimiento, por lo que el programa asignará la cantidad de palés del enésimo lote a la enésima fila en la que el movimiento sea “Entrar”. Será el usuario quién se debe asegurar de introducir correctamente y en orden cronológico las fechas de cada movimiento.

Esta subrutina se ejecuta al presionar el botón Actualizar en la hoja Planificación y resultados y llama a la siguiente función EnvíoDeLotesDePalés. La planificación debe realizarse previamente.

En segundo lugar, EnvíoDeLotesDePalés (ver Figura 55) funciona del mismo modo que RecepciónDeLotesDePalés pero para las salidas, que deberán ser planificadas cronológicamente teniendo en cuenta también los movimientos de entrada. De este modo funcionará correctamente la siguiente macro, InventarioDiario, que será llamada antes de finalizar esta subrutina para mantener el registro de los palés almacenados diarios actualizado.

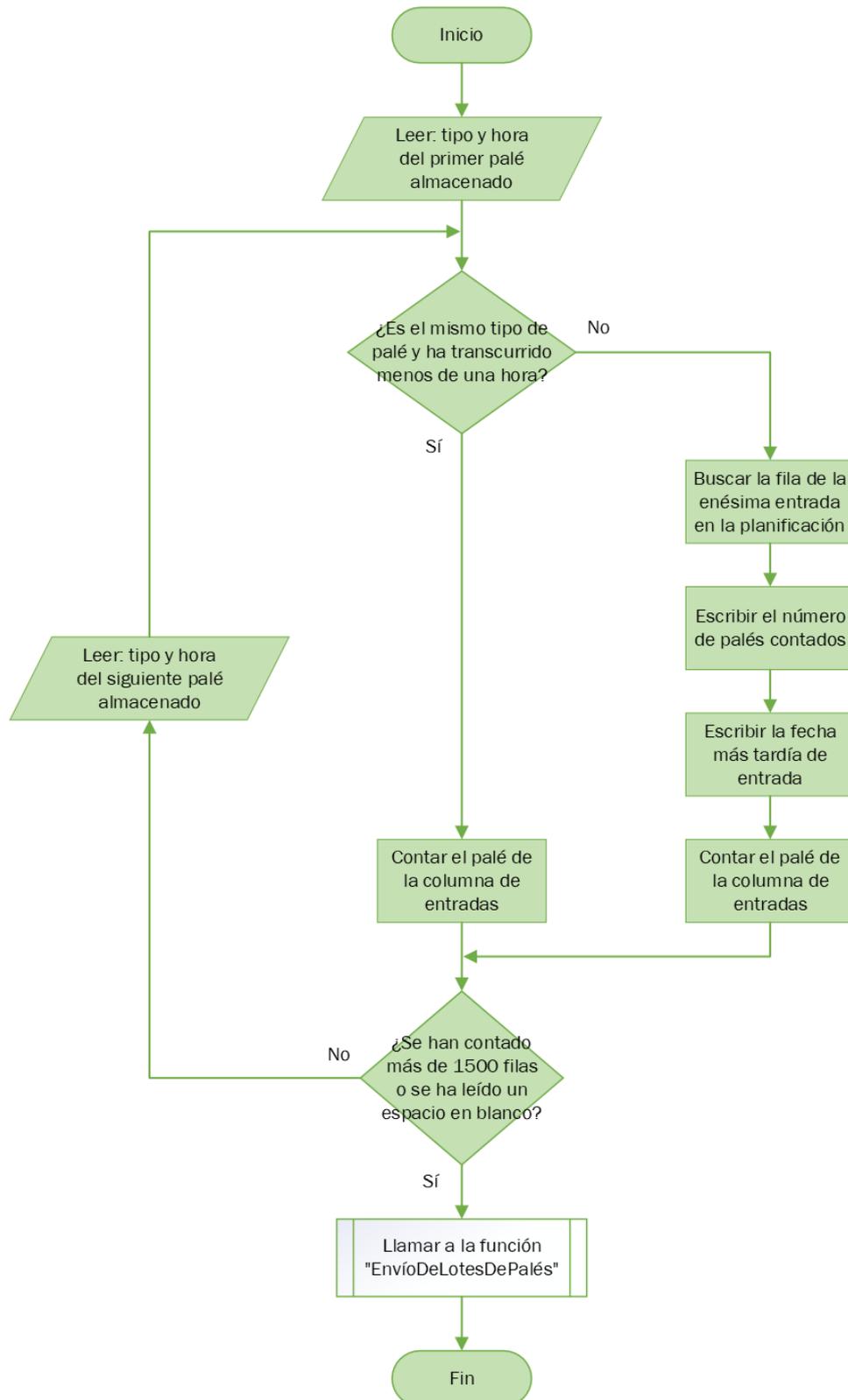


Figura 54. Subrutina RecepciónDeLotesDePalés

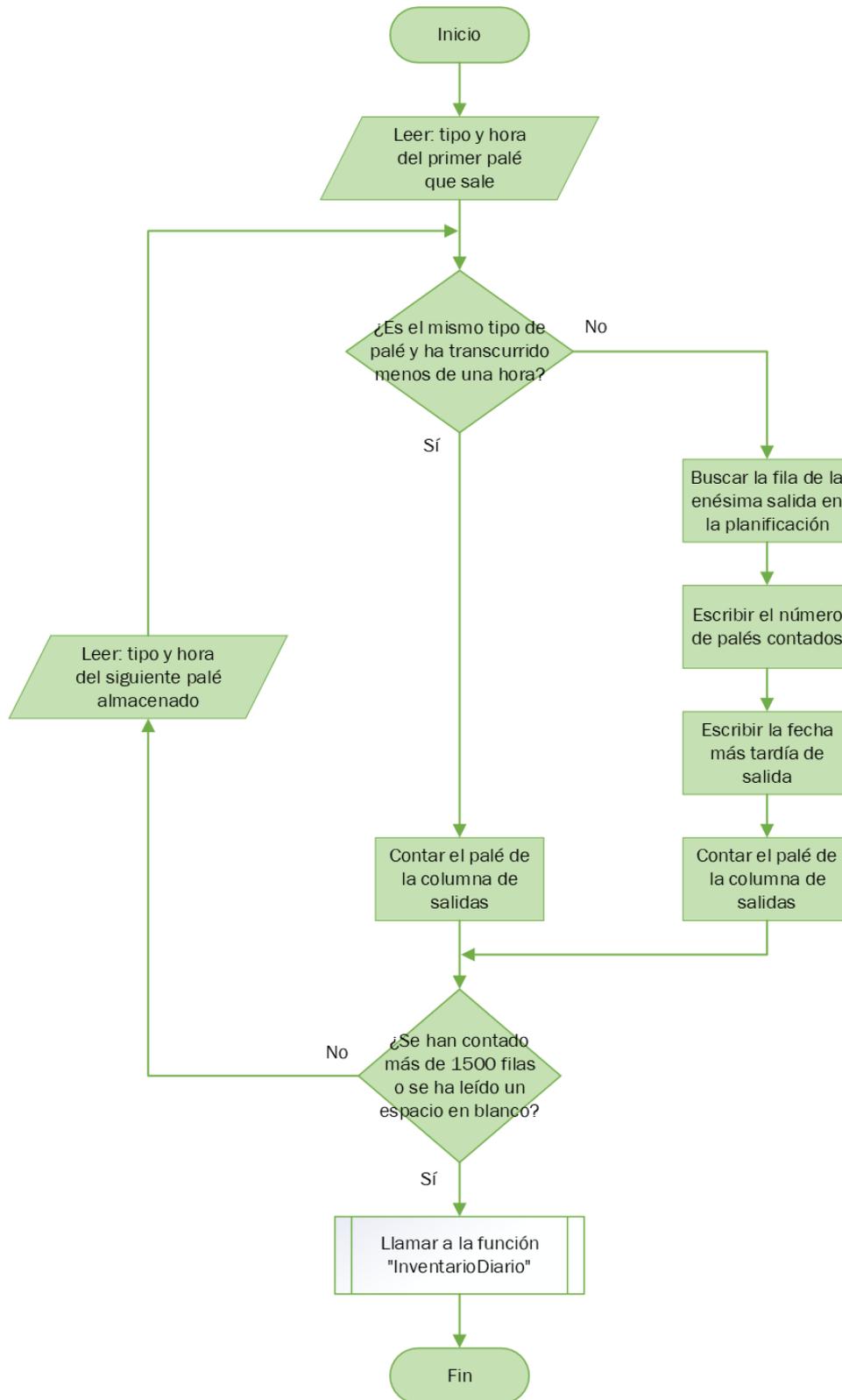


Figura 55. Subrutina EnvíoDeLotesDePalés



4.4.6. Subrutina InventarioDiario

Con el propósito de mantener un registro de la evolución del inventario del almacén a lo largo del tiempo, se dedica la hoja de la aplicación de Excel Inventario diario para anotar la cantidad de palés de cada tipo que se encuentran almacenados al final de cada jornada. La anotación es realizada de manera automática por esta subrutina cada vez que se pulsa el botón “Actualizar” en las hojas Planificación y resultados o Inventario diario. Después, estos datos se emplean en la gráfica Inventario diario de la hoja Cuadro de control y para la obtención de los KPIs de rotación, ya que es necesario conocer el inventario promedio. El funcionamiento es el siguiente:

En la hoja Planificación y resultados, después de la ejecución de la subrutina RecepciónDeLotesDePalés se tiene constancia de todos los movimientos realizados en el almacén. Conociendo de cada uno de ellos la cantidad de palés, la fecha, el tipo de producto en concreto y si dichas unidades llegan o abandonan el almacén. Haciendo uso de esta información, InventarioDiario comienza desde la fecha más temprana o principio de la tabla de Planificación y resultados sumando o restando la cantidad real de palés de la cantidad almacenada. Por ejemplo (Tabla 19): el día 08/05/2024 se sumaron las cantidades reales de productos C, A, B y A que llegaron al almacén y se restó la salida de A de tal forma que las variables cantidadDeA, cantidadDeB Y cantidadDeC contenían los valores 11, 8 y 10, respectivamente. Al detectar una fecha diferente, el programa anotó el resultado en el día 08/05/2024 de la tabla Inventario diario (Tabla 20) para cada producto y continuó contando las cantidades del día 09/05/2024. Al detectar una fila en blanco, anotó los resultados del día 9 y se finalizó la ejecución (Figura 56).

Fecha	Movimiento	Producto	Cantidad Planificada	Cantidad real
08/05/2024	Entrar	C	10	10
08/05/2024	Entrar	A	8	8
08/05/2024	Entrar	B	9	8
08/05/2024	Entrar	A	8	8
08/05/2024	Salir	A	5	5
09/05/2024	Entrar	B	9	9

Tabla 19. Ejemplo de tabla de Planificación y resultados

INVENTARIO DIARIO			
Fecha	Producto A	Producto B	Producto C
08/05/2024	11	8	10
09/05/2024	11	17	10

Tabla 20. Ejemplo de tabla de Inventario diario

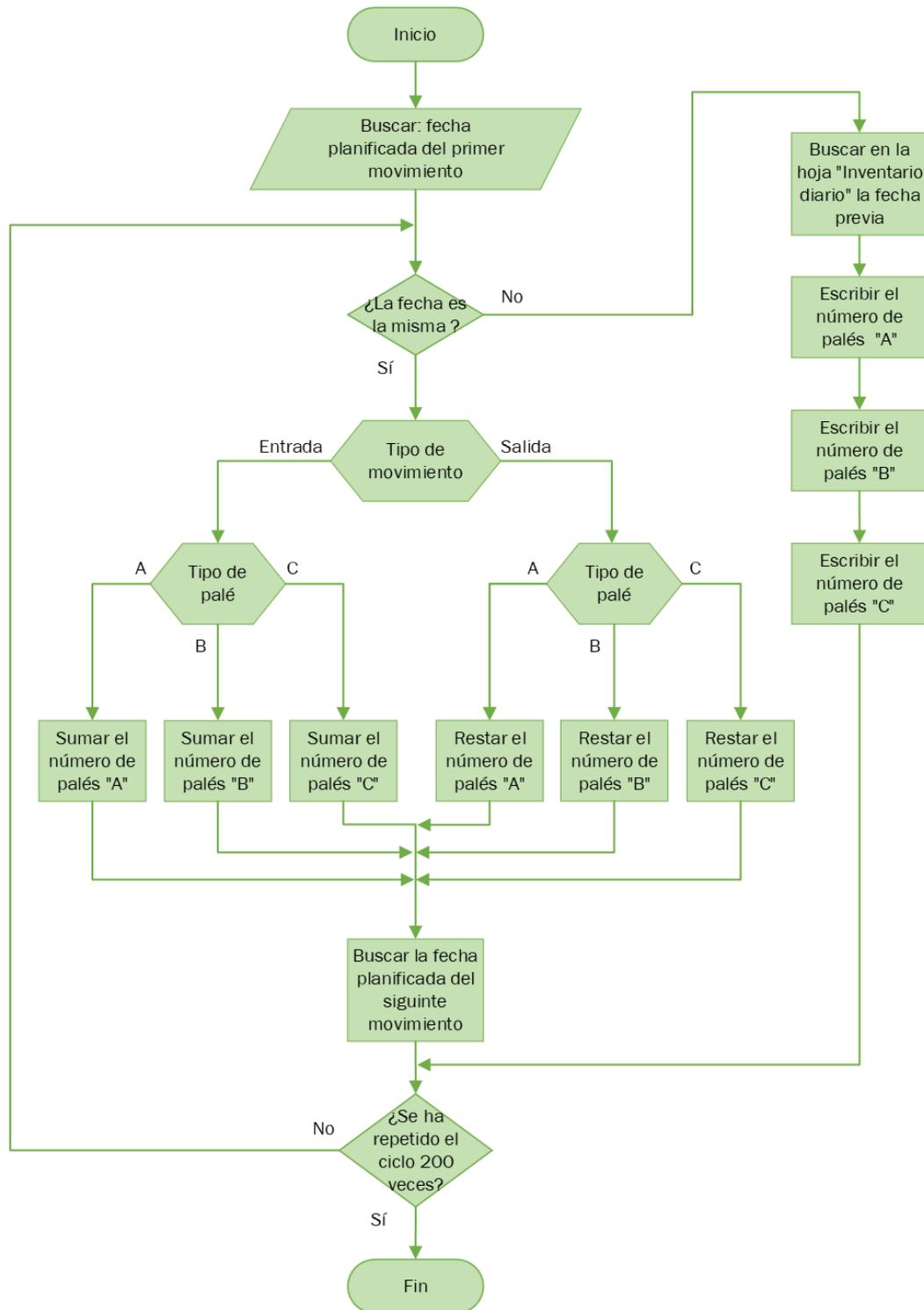


Figura 56. Subrutina InventarioDiario

4.4.7. Subrutina AjustarEjeXGráfica

La gráfica del inventario diario (en la hoja Cuadro de control) muestra la cantidad de productos A, B y C almacenados al final de cada jornada, según avanzan los días, se amplía la cantidad de datos en el eje X. Ajustarlo resulta una tarea repetitiva y además poco intuitiva, ya que, las unidades de este eje son fechas. Por lo que esta macro se encarga de realizar el ajuste de manera automática (ver Figura 57). Se activa con el botón Ajustar, situado a la derecha de la gráfica.

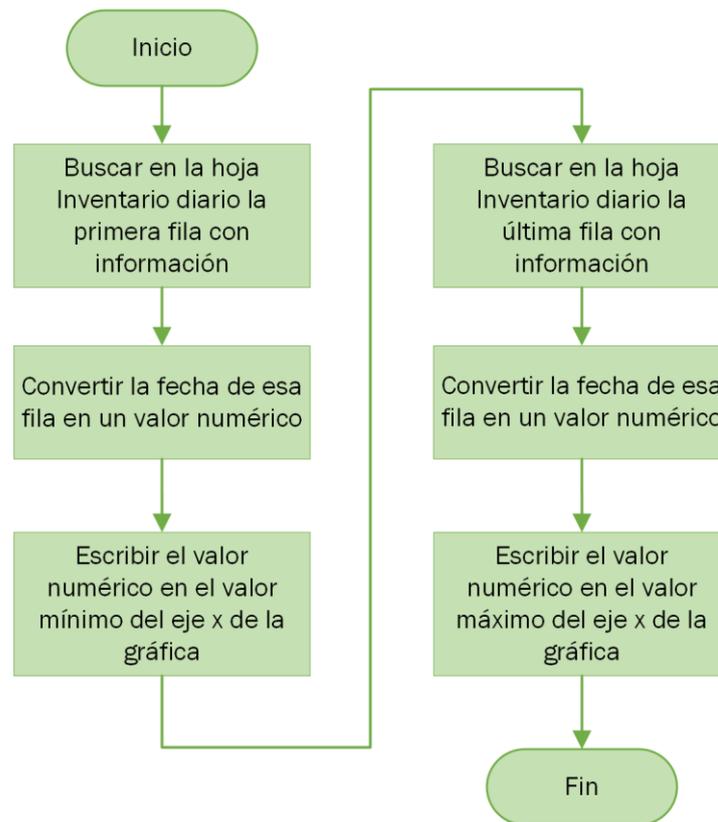


Figura 57. Subrutina AjustarEjeXGráfica

4.4.8. Subrutina EliminarDatos

Elimina toda la información relativa a las entradas, salidas y productos almacenados. Se ejecuta al presionar el botón Borrar datos situado junto a la tabla de entradas en la hoja Datos.



CAPÍTULO 5. PRESENTACIÓN DE RESULTADOS



Desarrollo de un sistema de control de inventario en tiempo real mediante tecnología RFID





5.1. Introducción

En este capítulo se mostrará el funcionamiento del sistema para el control de almacenes en tiempo real paso por paso. Para ello, primero se establecerán las hipótesis necesarias y luego se realizará una simulación.

La simulación partirá de una situación inicial en la que el almacén ha estado operando durante tres días y se tiene una cantidad de productos almacenados de los tres tipos.

En primer lugar, se planteará la necesidad de realizar dos envíos a un cliente para los cuales es necesario recibir varios palés previamente. Conociendo las cantidades almacenadas en el punto de partida, se procederá a planificar los movimientos de entrada y de salida de tal forma que la cantidad de productos se encuentre siempre por encima del stock de seguridad.

Luego, se mostrará cómo se registra la entrada y salida de productos, junto con las coordenadas mostradas por la pantalla TFT, para saber dónde almacenar los palés entrantes. Finalmente, se mostrará el desempeño del almacén con la información ofrecida en el cuadro de control y se extraerán unas conclusiones.

El objetivo de este capítulo se limita a mostrar el funcionamiento del sistema, que consiste principalmente en mostrar en tiempo real las existencias del almacén, realizar la planificación, registrar entradas y salidas, indicar ubicaciones de entrada y de salida y ofrecer KPIs. Funciones como la optimización del tamaño de los lotes, análisis y reducción de costes y otras funciones para mejorar la eficiencia, no se encuentran dentro del alcance de este trabajo.

5.2. Hipótesis

El dispositivo de control de almacenes desarrollado en este trabajo está adaptado a un caso particular que está condicionado por las siguientes hipótesis:

- Cada tipo de producto es suministrado por un proveedor diferente.
- El sensor de entrada, el LED verde y la pantalla se encuentran en la entrada del almacén, el sensor de salida y el LED amarillo en la zona de expedición y el dispositivo Arduino junto a un ordenador, fuera del almacén, en una oficina, aunque en el prototipo se muestran todos los componentes agrupados en la protoboard.
- Los sensores de entrada y de salida se reutilizan en el prototipo para cumplir con la función de escritura de las tarjetas/etiquetas RFID, sin



embargo, en la aplicación real, será necesario un tercer lector RFID para la grabación de datos en las mismas.

- Los envíos que se realizan o se reciben siempre constarán de varios palés y serán de un solo tipo de producto.
- Se trabajará con productos perecederos que deben ser gestionados con el método FIFO para enviar primero aquellos que caducan antes.
- Cada estantería está destinada a un tipo de producto.
- El almacén puede albergar 90 palés por estantería y hay 3 estanterías.
- El almacén opera los 7 días de la semana.
- La simulación se realiza con 3 tarjetas RFID solamente. Por lo que cada vez que una tarjeta pase por el lector de entrada considerará que es un producto diferente. Del mismo modo, cada vez que la tarjeta sea detectada por el lector de salida se considerará como que otro palé está abandonando el almacén y la elección del palé se realizará mediante el método FIFO.

5.3. Simulación

5.3.1. Situación inicial

El almacén ha estado operando durante tres días y la fecha de inicio de la simulación es el 20/06/2024. El método de gestión de almacén es FIFO. Se requiere realizar dos envíos de productos a un cliente al día siguiente. El primero consiste en 20 palés de producto A y el segundo en 24 palés de producto C. El envío debe realizarse lo antes posible. Para conseguirlo, es necesario haber planeado con antelación la recepción y expedición de productos de tal modo que cuando se haga el envío, la cantidad de palés almacenados no sea inferior al stock de seguridad. Tras las entregas y recepciones de varios días, la Figura 58 muestra la situación inicial a las 09:00 horas del 20/06/2024 y la Tabla 21 los datos de cada palé en stock. En la Figura 58 se muestran las ubicaciones ocupadas al inicio de la simulación.



Desarrollo de un sistema de control de inventario en tiempo real mediante tecnología RFID

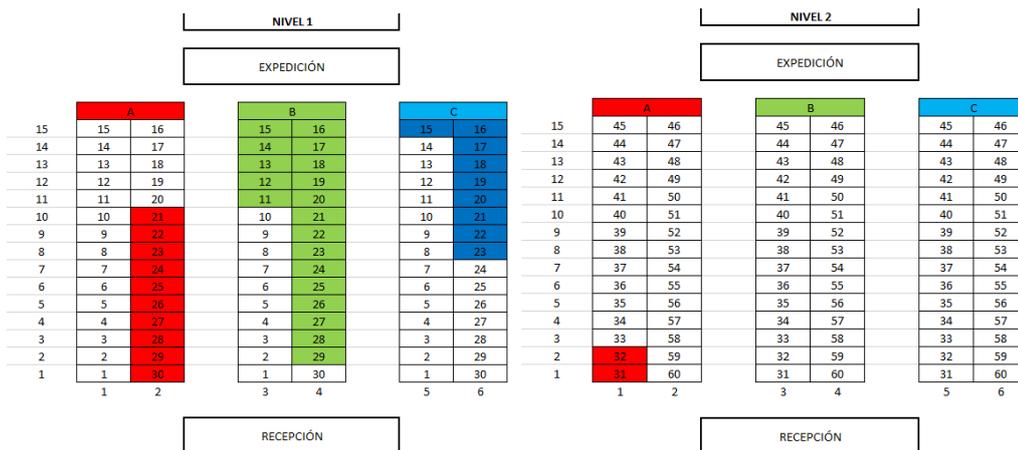


Figura 58. Estado del almacén al comienzo del 20/06/2024 de la simulación

La Tabla 21 muestra los datos de cada palé almacenado.

ALMACENADO							Total
							40
Tipo de product	SKU	Fecha de caducid	x	y	Nivel	Ubicació	
B	39	15/09/2024	3	11	1	11	11
B	40	15/09/2024	3	12	1	12	12
B	41	15/09/2024	3	13	1	13	13
B	42	15/09/2024	3	14	1	14	14
A	47	18/09/2024	2	10	1	21	21
A	48	18/09/2024	2	9	1	22	22
A	49	18/09/2024	2	8	1	23	23
A	50	18/09/2024	2	7	1	24	24
A	51	18/09/2024	2	6	1	25	25
A	52	18/09/2024	2	5	1	26	26
A	53	18/09/2024	2	4	1	27	27
A	54	18/09/2024	2	3	1	28	28
A	55	18/09/2024	2	2	1	29	29
A	56	18/09/2024	2	1	1	30	30
A	57	18/09/2024	1	1	2	31	31

Tabla 21. Registro de productos almacenados al comienzo del 20/06/2024

En el cuadro de mando figura información como el stock de seguridad para cada producto, que en este caso es de siete unidades para cada tipo. También se puede observar la cantidad de palés almacenados en tiempo real y la evolución del inventario a lo largo de los días e instantes previos al inicio simulación (Figura 59).

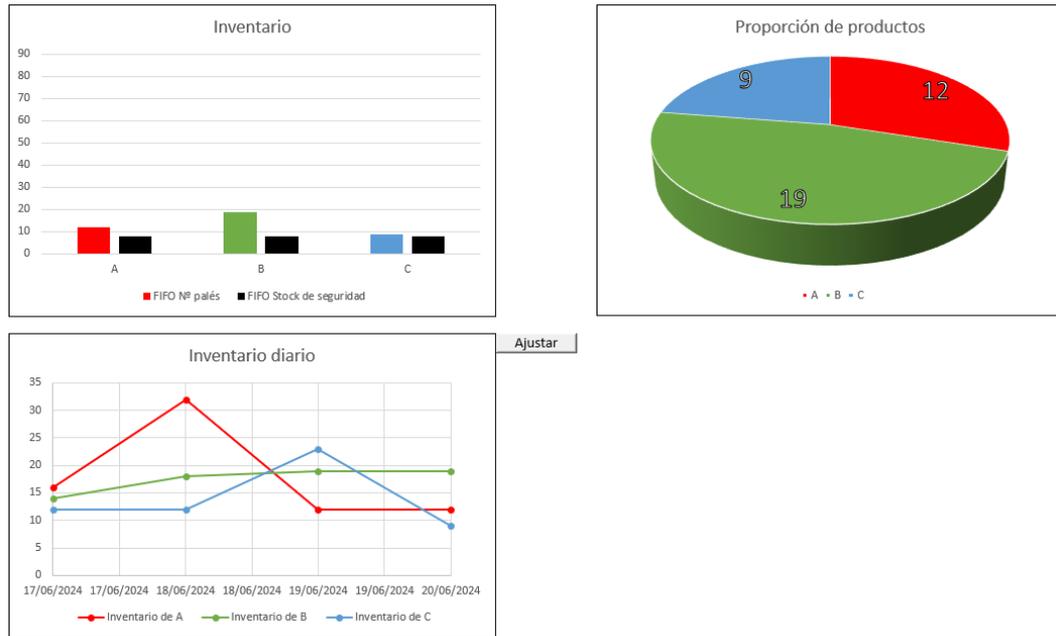


Figura 59. Inventario al comienzo del día 21/06/2024

5.3.2. Planificación de movimientos

Antes de iniciar la simulación se había planificado el abastecimiento correspondiente de productos A y B. Se realizó un pedido de 16 palés de producto A con una llegada al almacén prevista el día 20/06/2024 a las 10:00h y otro de 24 palés de producto C, para el mismo día, una hora después. Además, en la planificación también figuran los envíos que deben hacerse el día 21. Con las columnas de proyección, según se va realizando la planificación, se puede ver la cantidad de cada producto que se encontraría en stock si se realizan los movimientos con las cantidades previstas. De este modo, se puede ver rápidamente si se requiere un mayor abastecimiento para evitar consumir el inventario de seguridad (Tabla 22).

Proyección A	Proyección B	Proyección C
28	19	9
28	19	33
8	19	33
8	19	9

Tabla 22. Columnas de proyección de productos

En la hoja de Planificación y resultados (ver Tabla 23) se puede ver la situación actual de cada movimiento con aquellos que ya se han procesado y los que están pendientes cuyas celdas de cantidades reales, diferencia de cantidades y hora real están en blanco.

Fecha	Movimiento	Producto	Cantidad Planificada	Cantidad real	Diferencia de cantidad	Hora planificada	Hora real
17/06/2024	Entrar	C	12	12	0	9:00:00 AM	9:00:00 AM
17/06/2024	Entrar	A	16	16	0	12:00:00 PM	12:20:00 PM
17/06/2024	Entrar	B	14	14	0	5:00:00 PM	5:00:00 PM
18/06/2024	Salir	B	10	10	0	9:00:00 AM	9:00:00 AM
18/06/2024	Entrar	A	16	16	0	12:00:00 PM	12:00:00 PM
18/06/2024	Entrar	B	14	14	0	5:00:00 PM	5:15:00 PM
19/06/2024	Salir	A	10	10	0	9:00:00 AM	9:00:00 AM
19/06/2024	Entrar	B	1	1	0	11:00:00 AM	11:00:00 AM
19/06/2024	Salir	A	10	10	0	2:00:00 PM	2:00:00 PM
19/06/2024	Entrar	C	12	11	-1	5:00:00 PM	5:00:00 PM
20/06/2024	Salir	C	6	14	8	9:00:00 AM	8:51:00 AM
20/06/2024	Entrar	A	16			10:00:00 AM	
20/06/2024	Entrar	C	24			11:00:00 AM	
21/06/2024	Salir	A	20			9:00:00 AM	
21/06/2024	Salir	C	24			10:00:00 AM	

Tabla 23. Planificación de recepciones y envíos

5.3.3. Recepción y envío de pedidos

Después de realizar la planificación llegan los pedidos al almacén. Será necesario pegar, en cada palé, una etiqueta RFID que almacene el tipo de producto que entra y la fecha de caducidad de los mismos. Al entrar al almacén, el lector de entrada lo detecta y se muestran por pantalla las coordenadas donde el palé debe ser colocado. Por ejemplo, al entrar el primer palé del pedido de productos C, la pantalla mostraría la información que se puede observar en la Figura 60.

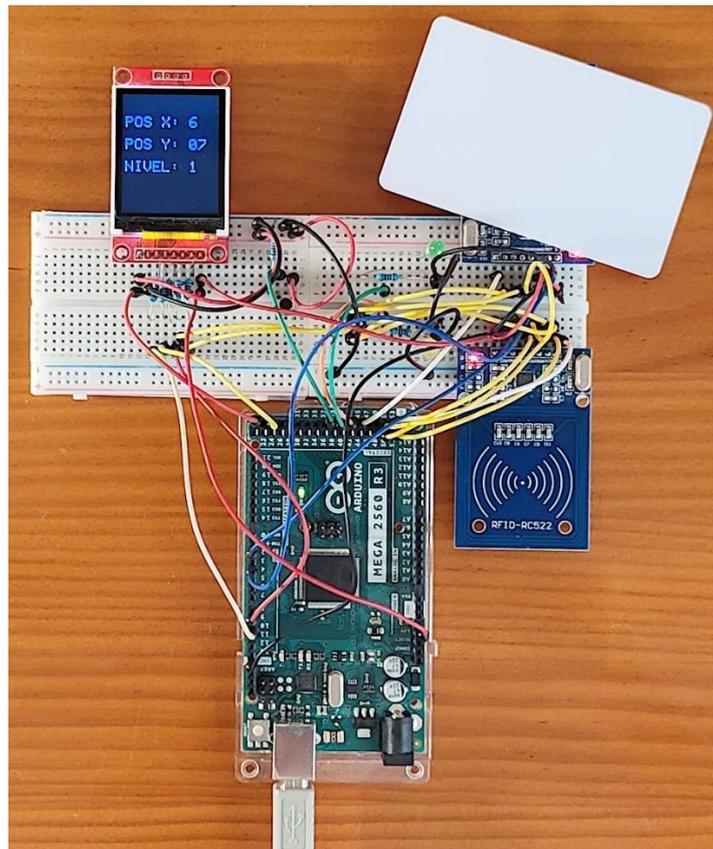


Figura 60. Ejemplo de entrada de un producto al almacén



Como se puede ver en la imagen, la pantalla indica que el primer palé de producto C que entra debe ser situado en las coordenadas $x = 6$, $y = 7$ y $z = 1$ o ubicación 24, ya que, como se ha visto antes (ver Figura 58), la última ubicación ocupada por un palé de tipo C era la 23 ($x = 6$, $y = 8$ y $z = 1$).

Entonces, a las 10:00 horas pasa por el lector RFID de entrada el último palé A del lote y a las 11:00 horas el último del lote de productos C. Toda la información queda registrada en la tabla de entradas y de productos almacenados. Luego, al pulsar el botón de actualizar, en la hoja de Planificación y resultados, vemos que las cantidades y horas de llegada previstas para los pedidos de A y de C se han cumplido (ver Tabla 24).

20/06/2024	Entrar	A	16	16	0	10:00:00 AM	10:00:00 AM
20/06/2024	Entrar	C	24	24	0	11:00:00 AM	11:00:00 AM
21/06/2024	Salir	A	20			9:00:00 AM	
21/06/2024	Salir	C	24			10:00:00 AM	

Tabla 24. Planificación y resultados al final del 20/06/2024

Al día siguiente, según lo programado, se realizan las salidas correspondientes acorde con las indicaciones del programa, que decidirá de qué ubicaciones y en qué orden se extraen los palés que se van a enviar. Toda esta información queda registrada en la tabla de salidas. Por otra parte, se ve que al pulsar de nuevo el botón de actualizar se rellenan las celdas restantes y se comprueba que tampoco se han dado desviaciones en la cantidad de palés ni en los tiempos de salida de manera significativa (ver Tabla 25).

20/06/2024	Entrar	A	16	16	0	10:00:00 AM	10:00:00 AM
20/06/2024	Entrar	C	24	24	0	11:00:00 AM	11:00:00 AM
21/06/2024	Salir	A	20	20	0	9:00:00 AM	9:03:00 AM
21/06/2024	Salir	C	24	24	0	10:00:00 AM	10:06:00 AM

Tabla 25. Planificación y resultados al final del 21/06/2024

5.3.4. Evaluación del desempeño

Aunque se tiene a disposición una gran cantidad de información con los registros de entradas, salidas y planificación, los KPI resultan útiles para ver fácil y cuantitativamente información de interés como puede ser la rapidez con la que salen ciertos productos del almacén (rotación de inventario).

Si se observan los KPIs de rotación (ver Tabla 26), los palés de tipo B entran y salen a un ritmo significativamente inferior que los palés A y C. Los siguientes indicadores son las rupturas de stock y la tasa de devoluciones que se calculan contando la cantidad de pedidos en los que no se ha podido satisfacer a tiempo la demanda del cliente y dividiendo la cantidad de palés devueltos entre el número de palés enviados, respectivamente.



KPIs	
Rotación de inventario de A	2,083
Rotación de inventario de B	0,562
Rotación de inventario de C	2,135
Rupturas de stock	0
Ocupación del almacén	13,333%
Tasa de devoluciones	1,136%
Recepciones incorrectas de A	0%
Recepciones incorrectas de B	0%
Recepciones incorrectas de C	33%
Envíos incorrectos	17%
Recepciones tardías de A	33%
Recepciones tardías de B	0%
Recepciones tardías de C	0%
Disponibilidad para la entrega	100,000%

Tabla 26. KPIs de la simulación

Por otro lado, se puede ver claramente en el gráfico del inventario (Figura 61) que las existencias son escasas o visto con un KPI, el almacén tiene un 13,33% de ocupación.

Después, se considera como recepción incorrecta aquella en la que la cantidad de productos difiere de la planeada. El ratio de recepciones incorrectas se calcula dividiendo los movimientos (lotes de palés) de entrada cuya cantidad de palés sea incorrecta entre los de entrada totales para un tipo de producto. El cálculo para los envíos incorrectos es el mismo pero para las salidas y agrupando los 3 tipos de productos.

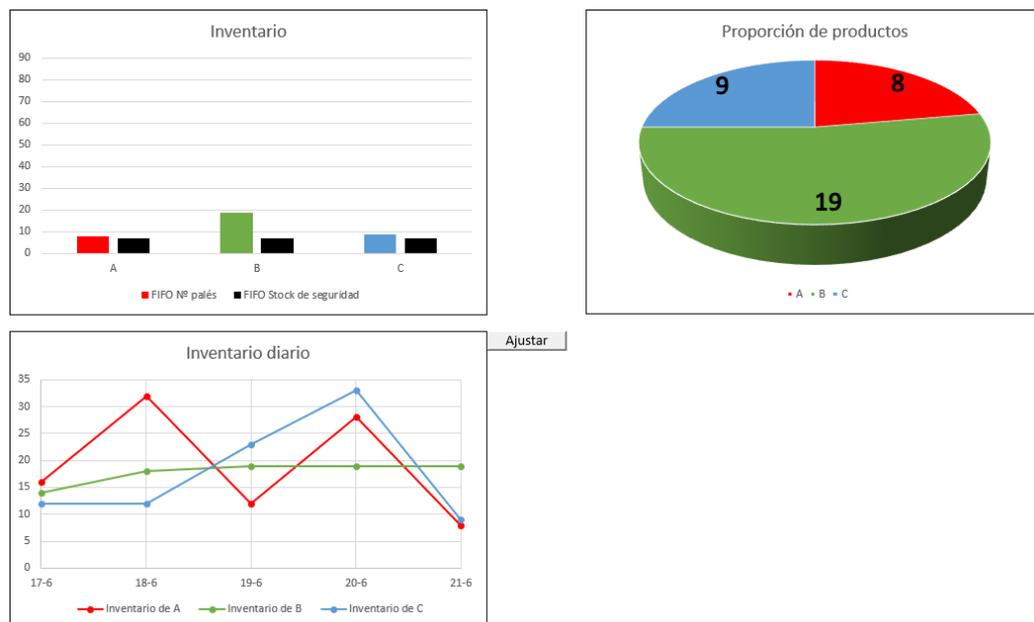


Figura 61. Gráficas del cuadro de control al final de la simulación



Las recepciones tardías se calculan de manera similar a las recepciones incorrectas. Las recepciones tardías calculan la cantidad de movimientos de entrada que se hayan retrasado más de cuarto de hora y los divide entre las entradas totales para un tipo de producto. Finalmente, la disponibilidad para la entrega resta al número total de movimientos de salida aquellos que se han retrasado y este resultado es dividido entre el número total de movimientos de salida. Las rupturas de stock también se consideran como movimientos que se han atrasado. Expresado de una manera más sencilla, la disponibilidad para la entrega es el número de pedidos entregados dentro del plazo establecido entre el total de pedidos enviados. Estos datos se obtienen de la hoja Planificación y resultados.

La utilidad de estos indicadores radica en mostrar el avance hacia los objetivos y en identificar áreas de mejora. Para el caso en particular de este almacén, aunque no se han establecido objetivos específicos, se observan ciertos aspectos a mejorar. Se debería conseguir que el proveedor de productos C se ajustase a la cantidad de productos solicitada y que el proveedor de productos A que se ajustase mejor a los tiempos acordados. Estas áreas de mejora son un ejemplo de las conclusiones que se pueden extraer de la tabla de KPIs, sin embargo, para obtener resultados más representativos sería necesario tener datos de un periodo temporal más extenso.



CAPÍTULO 6. ESTUDIO ECONÓMICO



Desarrollo de un sistema de control de inventario en tiempo real mediante tecnología RFID





6.1. Introducción

El estudio económico tiene como objetivo estimar los costes incurridos en la realización de este trabajo, de este modo se puede evaluar la viabilidad económica para su implantación.

Este proyecto ha sido desarrollado de principio a fin por un ingeniero en organización industrial, el cual ha tenido que llevar a cabo las cinco tareas en las que se divide el proyecto. Estas tareas son: la recopilación de información, el diseño y montaje del circuito, la programación del microcontrolador de Arduino, la programación del libro Excel y la redacción de la memoria.

Este capítulo consta de dos apartados, en el primero se analizan los diferentes costes por separado y en el segundo se presenta el presupuesto final desglosado tanto en actividades como en partidas de gasto.

6.2. Costes

En este apartado se cuantificarán los costes que supone la realización de este proyecto. En primer lugar, se analizarán los costes de personal, luego las amortizaciones, seguidas de los costes materiales, otros costes directos y costes indirectos. Los costes se han calculado sin IVA y además se muestran las horas dedicadas a cada sección del proyecto.

6.2.1. Costes de personal y reparto de tiempos

El proyecto ha sido realizado por un ingeniero en Organización Industrial, bajo la supervisión de un director de proyecto y ha tenido que realizar cinco tareas principales para finalizarlo:

- Recopilación de información
- Diseño y montaje del circuito
- Programación del dispositivo Arduino
- Programación del archivo Excel
- Redacción de la memoria

En total, se han necesitado 320 horas para completarlas (ver Tabla 27), suponiendo la redacción de la memoria prácticamente la mitad del tiempo empleado. Junto con la redacción, se realizó la recopilación de la información, que además incluye el proceso de lectura y comprensión. La programación del archivo Excel ha sido la segunda tarea más extensa ya que requería



programación en VBA, la escritura de fórmulas en las celdas, gestionar la presentación de la información y la gestión del transmisor de datos.

Programar el funcionamiento de Arduino resultó una tarea más sencilla que la programación en Excel al tener una funcionalidad más reducida, ya que se limita a leer y escribir en las tarjetas RFID, controlar la pantalla y a realizar la comunicación con el puerto serie.

Finalmente, el diseño y montaje del circuito incluye los diagramas realizados con Proteus para mostrar las conexiones de Arduino con sus periféricos y el montaje de cada componente y sus conexiones en la protoboard (ver Figura 62).

Fase del proyecto	Horas	Porcentaje
Redacción de la memoria	150	47%
Programación Excel	60	19%
Recopilación de información	45	14%
Programación Arduino	40	13%
Diseño y montaje del circuito	25	8%
Total	320	100%

Tabla 27. Horas dedicadas a cada actividad

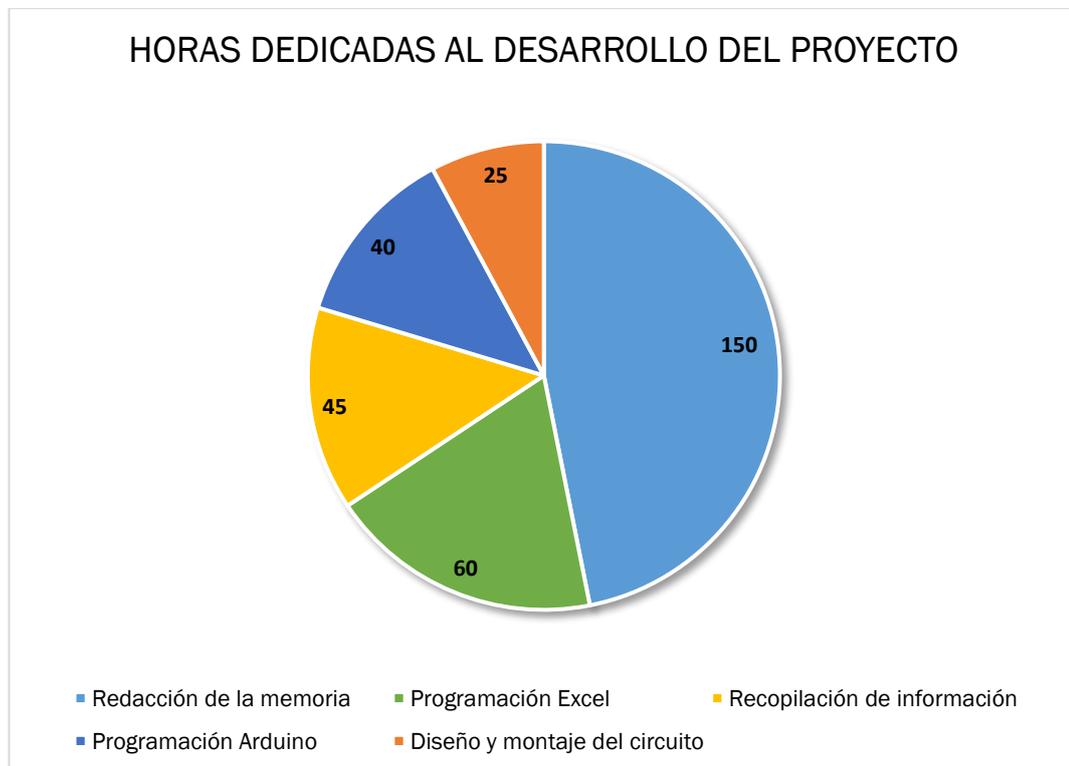


Figura 62. Gráfico circular de las horas dedicadas al desarrollo del proyecto



Este periodo de 320 horas equivale a 2 meses suponiendo que cada mes tiene 20 días laborables y la jornada es de 8 horas. El sueldo anual del ingeniero es de 28.000 € brutos anuales repartido en 14 pagas, lo cual supone un sueldo bruto mensual de 2.000 € y dos pagas extra, una en verano y otra en Navidad de 2.000 € cada una. Para hallar el coste total de este empleado se ha multiplicado el sueldo mensual por los meses trabajados y a este valor se le ha añadido la parte proporcional de la paga extra de verano y la seguridad social que es el 30% del sueldo. El coste total para la empresa es de 6.066,67 € o de 18,96 € la hora (ver Tabla 28).

Ingeniero en Organización Industrial	
Salario bruto mensual (€)	2.000,00
Salario bruto para el periodo trabajado (€)	4.000,00
Parte proporcional de la paga extra (€)	666,67
Seguridad social 30% (€)	1.400,00
Coste para la empresa (€)	6.066,67
Coste semanal (€)	758,33
Coste por hora (€)	18,96

Tabla 28. Coste del ingeniero en organización industrial

6.2.2. Amortizaciones

En las amortizaciones se han tenido en cuenta el ordenador personal y las licencias de los programas empleados con la excepción de la IDE de Arduino, ya que es un software gratuito. Tampoco se ha tenido en cuenta la mensualidad de Microsoft 365 al no ser un activo de la empresa sino un servicio contratado.

Se ha realizado una amortización lineal dividiendo la diferencia del valor del activo y su valor residual entre los años útiles del activo. Los valores de los activos, sus valores residuales y los años útiles se muestran en la Tabla 29.

Después de calcular las cuotas de amortización anuales se imputa la parte proporcional correspondiente a la duración del proyecto en la última columna, que como se ha indicado previamente ha sido de 2 meses.

Concepto	valor	valor residual	años	amortización anual	amortización periodo
Ordenador personal	826,45	80	6	124,41	20,73
Licencia Proteus	5.620,00	-	5	1.124,00	187,33
Licencia Windows 11	119,83	-	6	19,97	3,33
TOTAL				1.268,38	211,40

Tabla 29. Amortizaciones.



6.2.3. Costes materiales

Los costes materiales del prototipo incluyen los precios de cada componente y al igual que el resto de costes, no tienen el IVA incluido. Destaca como elemento más caro el dispositivo Arduino Mega 2560 R3, seguido de la pantalla TFT y los sensores RFID (ver Tabla 30).

Concepto	Cantidad	Coste/u (€)	Coste total (€)
Sensor RFID	2	1,49	2,98
Pantalla TFT	1	7,40	7,40
LEDs	2	0,05	0,10
Resistencias 1 kΩ	8	0,07	0,56
Pulsador	1	0,17	0,17
Cables (macho - macho)	33	0,05	1,70
Protoboard	1	2,06	2,06
Arduino Mega 2560 R3	1	42,00	42,00
Cable USB tipo A a USB tipo B	1	5,36	5,36
Tarjetas/llaveros MIFARE S50	4	0,82	3,28
TOTAL			65,61

Tabla 30. Costes materiales del prototipo

6.2.4. Otros costes directos

Los últimos costes directos que se deben considerar son la línea telefónica y la licencia de Microsoft 365. El coste de la licencia, supone al mes un coste de 5.6 € y solamente puede ser utilizada por un usuario, por este motivo es considerada un coste directo. Lo mismo sucede con el teléfono. En el importe total se muestra el coste de ambos conceptos para el periodo antes mencionado (Tabla 31).

Concepto	importe mensual	importe total
Teléfono	20,00	40,00
Licencia de Microsoft 365	5,60	11,20
TOTAL	25,60	51,20

Tabla 31. Otros costes directos

6.2.5. Costes indirectos

Primero se hallaron los costes indirectos mensuales y después se multiplicaron por 2 meses. Para el cálculo de los costes indirectos hay que tener en consideración que el proyecto se ha llevado a cabo en una oficina en la que trabajan 15 empleados, por lo que el criterio de reparto de estos costes ha sido



el número de trabajadores. Como de 15 solamente 1 de ellos ha formado parte de este trabajo, se imputa el 7% de los costes indirectos de la oficina a este proyecto (Tabla 32).

Concepto	importe mensual	importe total	imputado	coste
Electricidad	95,00	190,00	7%	12,67
Servicio de limpieza	225,00	450,00	7%	30,00
Internet	60,00	120,00	7%	8,00
Agua	20,00	40,00	7%	2,67
Calefacción	110,00	220,00	7%	14,67
Alquiler oficinas	1.750,00	3.500,00	7%	233,33
TOTAL	2.260,00	4.520,00		301,33

Tabla 32. Costes indirectos

6.3. Presupuesto final

6.3.1. Coste total desglosado por tareas

Una vez calculados los costes de personal, amortizaciones, materiales, otros costes directos y los indirectos se reparten en tareas. La suma del coste de todas las tareas es el presupuesto final.

El coste de personal que corresponde a cada actividad es proporcional a la cantidad de horas que han sido necesarias para completarla. Del mismo modo, las amortizaciones, otros costes directos y los costes indirectos se reparten según el mismo criterio. Los costes materiales solamente se corresponden con los componentes del prototipo por lo que se asignan en su totalidad a la tarea de diseño y montaje del circuito. También se asigna el 100% de la amortización de la licencia de Proteus a la actividad de diseño y montaje del circuito. La Tabla 33 expone los costes de redacción de la memoria.

Concepto	horas	coste/hora	coste total
Personal	150	18,96	2.843,75
Amortización	150	-	11,28
Ordenador personal	150	-	9,72
Licencia Windows 11	150	-	1,56
Otros costes directos	150	-	24,00
Costes indirectos	150	-	141,25
Total			3.020,28

Tabla 33. Costes de redacción de la memoria



En la Tabla 34 se muestran los costes de programación del archivo Excel.

Concepto	horas	coste/hora	coste total
Personal	60	18,96	1.137,50
Amortización	60	-	4,51
Ordenador personal	60	-	3,89
Licencia Windows 11	60	-	0,62
Otros costes directos	60	-	9,60
Costes indirectos	60	-	56,50
Total			1.208,11

Tabla 34. Costes de programación del archivo Excel

La Tabla 35 presenta los costes de la recopilación de información.

Concepto	horas	coste/hora	coste total
Personal	45	18,96	853,13
Amortización	45	-	3,38
Ordenador personal	45	-	2,92
Licencia Windows 11	45	-	0,47
Otros costes directos	45	-	7,20
Costes indirectos	45	-	42,38
Total			906,08

Tabla 35. Costes de la recopilación de información

En la Tabla 36 se muestran los costes de programación del dispositivo Arduino.

Concepto	horas	coste/hora	coste total
Personal	40	18,96	758,33
Amortización	40	-	3,01
Ordenador personal	40	-	2,59
Licencia Windows 11	40	-	0,42
Otros costes directos	40	-	6,40
Costes indirectos	40	-	37,67
Total			805,41

Tabla 36. Costes de programación Arduino

Finalmente, los costes de la última actividad se reflejan en la Tabla 37.



Concepto	horas	coste/hora	coste total
Personal	25	18,96	473,96
Amortización	25	-	189,21
Ordenador personal	25	-	1,62
Licencia Proteus	-	-	187,33
Licencia Windows 11	25	-	0,26
Costes materiales	-	-	65,61
Otros costes directos	25	-	4,00
Costes indirectos	25	-	23,54
Total			756,33

Tabla 37. Costes de diseño y montaje del circuito

Finalmente, se suman los costes de todas las actividades y se obtiene el presupuesto final, con las actividades ordenadas de la más costosa a la menos costosa (Tabla 38).

Actividad	horas	coste (€)
Costes de redacción	150	3.020,28
Costes de programación Excel	60	1.208,11
Costes de recopilación de la información	45	906,08
Costes de programación Arduino	40	805,41
Costes de diseño y montaje	25	756,33
TOTAL	320	6.696,21

Tabla 38. Coste total desglosado por actividades

Se puede observar en la Figura 63 que los costes se distribuyen entre las actividades de un modo muy similar a las horas, ya que el criterio de reparto de los costes ha sido el porcentaje de horas dedicadas. Sin embargo, existe una diferencia significativa en la proporción de la actividad diseño y montaje en la gráfica de las horas respecto de su proporción en la gráfica del coste (Figura 63). Esto se debe a que se le imputan los costes materiales y la amortización de Proteus como se ha indicado previamente.

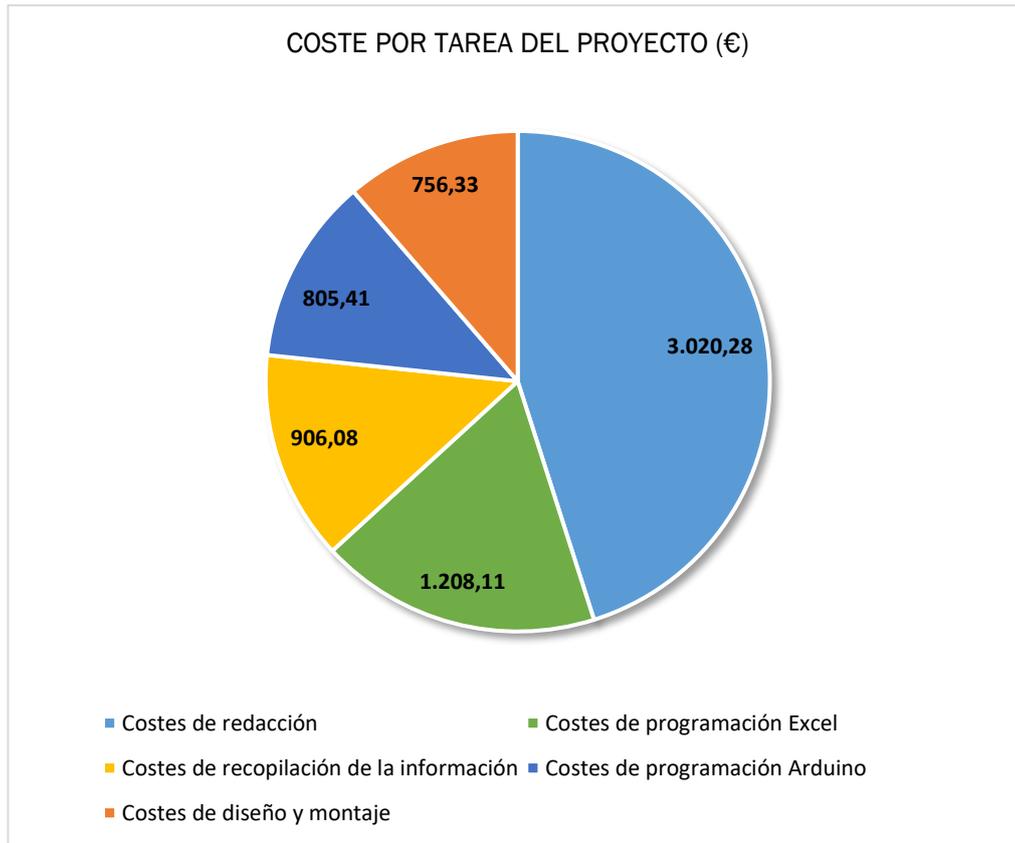


Figura 63. Gráfico circular del coste por tarea del proyecto

6.3.2. Coste total desglosado por partidas de gasto

Finalmente, estos son los costes incurridos por la realización del proyecto, pero ahora desglosados según la partida de gasto. El coste del ingeniero ha supuesto el 90,6 % del presupuesto, seguido por los costes indirectos que representan el 4,5 %, las amortizaciones el 3,2 %, los costes materiales el 1 % y el resto de costes directos el 0,8 % (Tabla 39 y Figura 64).

Partida de gasto	coste (€)
Personal	6.066,67
Costes indirectos	301,33
Amortizaciones	211,40
Costes materiales	65,61
Otros costes directos	51,20
TOTAL	6.696,21

Tabla 39. Coste total desglosado por partidas de gasto

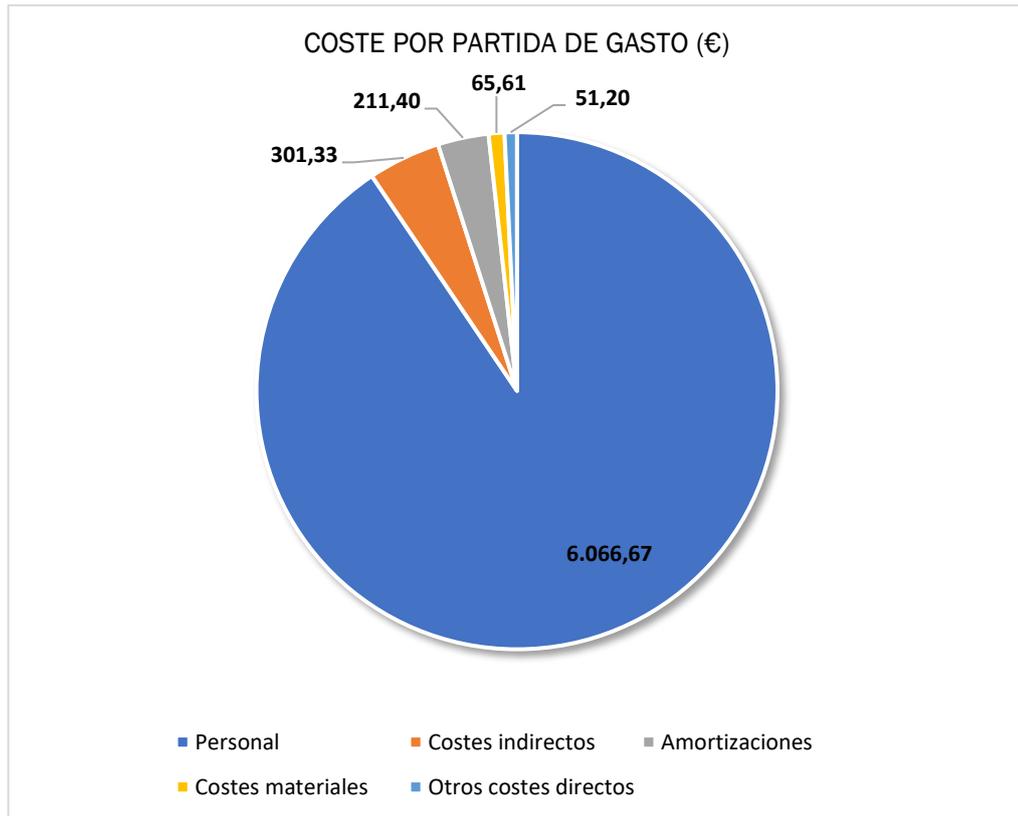


Figura 64. Gráfico circular sobre el coste por partida de gasto



Desarrollo de un sistema de control de inventario en tiempo real mediante tecnología RFID





CAPÍTULO 7. CONCLUSIONES Y LÍNEAS DE MEJORA



Desarrollo de un sistema de control de inventario en tiempo real mediante tecnología RFID





7.1. Conclusiones

Tras la finalización del trabajo se puede comprobar la consecución de todos los objetivos marcados en el primer capítulo, tanto del objetivo principal como de los secundarios. En el capítulo de presentación de resultados se mostró cómo se ha desarrollado un sistema de control de almacenes en tiempo real mediante tecnología RFID. Este sistema, aunque presente unas funcionalidades limitadas y adaptadas a un escenario particular, cumple con su cometido de una forma correcta y consistente.

Cabe destacar una utilización mínima de materiales para el desarrollo del prototipo, permitiendo un margen de mejora económico relativamente reducido, ya que, el presupuesto consiste principalmente en los costes de personal y los costes materiales suponen el 1 % del total. Con un presupuesto final estimado de 6.696,21 €, el prototipo supone un coste asumible para la mayoría de empresas, sin embargo, a este coste inicial se le debe agregar el coste de la instalación y de operación.

Por otra parte, la relevancia de la tecnología RFID está en alza debido a sus ventajas frente a otras alternativas como pueden ser los códigos de barras, las cuales, como la automatización y la reducción de errores y tiempos de escaneo, son características clave para el mantenimiento de la ventaja competitiva de las empresas. Por este motivo, se destaca la importancia del tema seleccionado para la realización de este trabajo.

En resumen, el resultado de este trabajo ha sido un sistema que mediante el empleo de hardware y software sencillo y barato ha permitido la gestión de inventarios mediante una tecnología que destaca por ser emergente en la industria actual.

Finalmente, aunque se hayan alcanzado con éxito los objetivos establecidos en este trabajo, no se ha diseñado el sistema final que se instalaría en el almacén. Por otro lado, tampoco se ha calculado cuál podría ser el beneficio o la mejora del desempeño de este sistema frente a una gestión tradicional de un almacén empleando papel o códigos de barras. Por lo tanto, el diseño de la instalación y el análisis de la mejora se podrían realizar en trabajos futuros junto con modificaciones del sistema que se proponen a continuación.

7.2. Líneas de mejora

En este apartado se proponen algunas mejoras para el sistema de control de almacenes desarrollado en este Trabajo de Fin de Grado, las cuales, podrían ser implementadas en trabajos posteriores. Estas líneas de mejora se dividen



en modificaciones del sistema de información o aplicación de Excel y en modificaciones del prototipo.

La primera línea que se sugiere consiste en permitir al usuario realizar modificaciones en la distribución en planta del almacén, pudiendo incrementar o reducir el número de ubicaciones en cualquiera de los tres ejes. Además, se debería ofrecer al usuario la posibilidad de seleccionar la cantidad de estanterías y sus respectivas disposiciones. También, se podría incluir la opción de agregar diferentes tipos de estantes, entre los que se incluyen las estanterías FIFO o LIFO mencionadas en el capítulo del estado del arte, en el apartado de almacenes (ver página 42). Esta línea de mejora está orientada a ofrecer al usuario un sistema que se ajuste a sus necesidades con la mayor precisión posible, pudiendo adaptarse a un mayor número de almacenes.

La segunda línea consiste en permitir la reubicación de las unidades de almacenamiento y en desarrollar un sistema más elaborado que también permita la expedición de productos en base a unas indicaciones más detalladas del cliente, en lugar de solo realizar envíos de productos en base a un método FIFO o LIFO. Es decir, enviar los productos que se ajusten a unos requisitos como pueden ser un color, un modelo o un tamaño específicos. Por poner un ejemplo, un tipo de producto podría ser refrescos y dentro de este tipo habría envases de diferentes tamaños.

Para conseguirlo, sería necesario emplear el identificador único de cada tarjeta o etiqueta como SKU, en lugar de asignarlo desde Excel con la entrada de productos. Sabiendo la ubicación en el almacén, un empleado podría encontrarlo y llevarlo a la zona de expedición, en la cual el lector RFID de salida captaría el identificador de ese producto permitiendo al programa saber qué producto ha salido.

Además, esta mejora también es interesante porque permite identificar cada producto de manera inequívoca en cualquier instante, ya sea a la hora de hacer inventario o de comprobar que salen del almacén exactamente los productos planificados. Esta característica no ha podido ser implementada en el prototipo del proyecto, ya que se requerirían tantas tarjetas como productos almacenados y era un gasto que no estaba justificado para solamente mostrar la funcionalidad del sistema de control de almacenes. Para poder hacer el inventario, sería necesario desarrollar un lector RFID portable que pudiesen utilizar los empleados con comodidad, además de una extensión de los programas de Arduino y Excel.

La tercera línea trata sobre la mejora del prototipo. Se ha podido comprobar que la zona de interrogación de los lectores RFID es inferior a los tres



centímetros, aproximadamente. Cambiando el lector RFID sería posible incrementar la zona de interrogación del lector para facilitar la lectura.

Luego, se podría ampliar el abanico de funcionalidades del sistema mejorando las etiquetas RFID. Con el uso de etiquetas activas (transpondedores activos, ver Figura 5) se podría ampliar el rango de lectura. Esto es de especial utilidad para permitir a los empleados hacer el inventario de manera rápida, fácil y a distancia. De este modo se reduciría el tiempo y los errores en comparación con otros métodos para hacer inventario.

Por otra parte, las etiquetas empleadas en este proyecto tienen 1 KB de almacenamiento de memoria de los cuales tan solo se emplean 32 bytes. Entonces, se podría aprovechar la capacidad restante de memoria para incluir una mayor cantidad de información del producto como puede ser la trazabilidad, historial de movimientos, fecha de entrada al almacén, fecha de salida, cantidad, peso, información sobre el proveedor y sobre el cliente, entre otros.

Además, se podría mejorar la estética y fiabilidad del hardware soldando las conexiones realizadas en la protoboard en un shield de Arduino y protegiendo los componentes dentro de una carcasa. Además, se podría instalar un arco en la entrada y otro en la salida del almacén donde se situarían los lectores RFID, de tal forma que los palés estarían forzados a atravesar estos arcos, garantizando y facilitando de este modo el registro de los movimientos de los palés.



Desarrollo de un sistema de control de inventario en tiempo real mediante tecnología RFID





BIBLIOGRAFÍA



Desarrollo de un sistema de control de inventario en tiempo real mediante tecnología RFID





Referencias

- Arbones Malisani, E. A. (1990). *Logística empresarial*. Marcombo.
- Arduino. (s.f.). *Arduino Mega 2560 Rev3*. Obtenido de Arduino.cc. <https://store.arduino.cc/products/arduino-mega-2560-rev3>
- Arduino. (s.f.). *Arduino Mega 2560 Rev3 [Fotografía]*. Arduino.cc. <https://store.arduino.cc/products/arduino-mega-2560-rev3?queryID=undefined>
- Atmel. (2014). *Atmel ATmega640/V-1280/V-1281/V-2560/V-2561/V*. Microchip Technology Inc. https://ww1.microchip.com/downloads/en/devicedoc/atmel-2549-8-bit-avr-microcontroller-atmega640-1280-1281-2560-2561_datasheet.pdf
- Axelsson, J. (2000). *Serial Port Complete*. Lakeview Research.
- Bartholdi, J. J. y Hackman, S. T. (2019). *Warehouse & Distribution Science*. The Supply Chain & Logistics Institute. <https://www.warehouse-science.com/book/index.html>
- Características generales de los microcontroladores. (2023). En M. I. del Valle (Comp.), *IOI: Sistemas Electrónicos Basados en Microcontrolador*. Universidad de Valladolid.
- Costa, F., Genovesi, S., Borgese, M., Michel, A., Dicandia, F. A. y Manara, G. (2021). A review of RFID sensors, the new frontier of internet of things. *Sensors*, 21(9), 3138. <https://doi.org/10.3390/s21093138>
- del Valle, L. (s.f.). *Cómo instalar una librería de Arduino en el entorno de desarrollo*. Programar fácil. https://programarfácil.com/blog/arduino-blog/instalar-una-libreria-de-arduino/#Como_instalar_una_libreria_de_Arduino
- Demir, M. F., Cankirli, A., Karabatak, B., Yavariabdi, A., Mendi, E. y Kusetogullari, H. (2018). Real-Time Resistor Color Code Recognition using Image Processing in Mobile Devices. *En 2018 International Conference on Intelligent Systems (IS)*, 26-30. IEEE. <https://doi.org/10.1109/IS.2018.8710533>
- Dhaker, P. (2018). Introduction to SPI Interface. *AnalogDialogue*, 52(3), 49-53.



- Finkenzeller, K. (2010). *RFID Handbook. Fundamentals and Applications in Contactless Smart Cards, Radio Frequency Identification and Near-Field Communication* (3ª ed.). Wiley.
- García, J. (15 de 09 de 2023). Así será comprar en Zara en 2025: sin pasar por caja casi ni para quitar la alarma. *Cinco Días*.
<https://cincodias.elpais.com/companias/2023-09-15/asi-sera-comprar-en-zara-en-2025-sin-pasar-por-caja-casi-ni-para-quitar-la-alarma.html#>
- Ghael, H. D., Solanki, L. y Sahu, G. (2020). A review paper on raspberry pi and its applications. *International Journal of Advances in Engineering and Management (IJAEM)*, 2(12), 4. <https://doi.org/10.35629/5252-0212225227>
- González, R. y Villapol, M. E. (2013). Un Estudio Comparativo de las Tecnologías para Redes de Área Personal. *ResearchGate*.
<https://doi.org/10.13140/2.1.4728.9601>
- Herrero, J. C. y Sánchez, J. (2015). Una mirada al mundo Arduino. *Tecnología y desarrollo*.
- Hompel, M. y Schmidt, T. (2006). *Warehouse Management. Automation and Organisation of Warehouse and Order Picking Systems*. Springer Science & Business Media.
- Introducción a los procesadores digitales. (2023). En M. I. del Valle (Comp.), *IOI: Sistemas Electrónicos Basados en Microcontrolador*. Universidad de Valladolid.
- Kanagachidambaresan, G. R., Anand, R., Balasubramanian, E. y Mahima, V. (2020). *Internet of Things for Industry 4.0: Design, Challenges and Solutions*. Springer. <https://doi.org/10.1007/978-3-030-32530-5>
- Khattab, A., Jeddi, Z., Amini, E. y Bayoumi, M. (2017). *RFID Security. A Lightweight Paradigm*. Springer. <https://doi.org/10.1007/978-3-319-47545-5>
- Kondaveeti, H. K., Kumaravelu, N. K., Vanambathina, S. D., Mathe, S. E. y Vappangi, S. (2021). A systematic literature review on prototyping with Arduino: Applications, challenges, advantages, and limitations. *Computer Science Review*, 40, 100364.
<https://doi.org/10.1016/j.cosrev.2021.100364>



- Lasi, H., Fettke, P., Kemper, H. G., Feld, T. y Hoffmann, M. (2014). Industry 4.0. *Business & information systems engineering*, 6, 239-242. <https://doi.org/10.1007/s12599-014-0334-4>
- Llamas, L. (2014). *Leer un pulsador con Arduino*. Obtenido de Luis Llamas. Ingeniería, informática y diseño. <https://www.luisllamas.es/leer-un-pulsador-con-arduino/>
- Mondal, S., Kumar, D. y Chahal, P. (2021). Recent Advances and Applications of Passive Harmonic RFID Systems: A Review. *Micromachines*, 12(4), 420. <https://doi.org/10.3390/mi12040420>
- NXP Semiconductors. (2007). *MF1 IC S50. Functional specification*. NXP Semiconductors. <https://cdn-shop.adafruit.com/datasheets/S50.pdf>
- Organización Internacional de Normalización. (s.f.). *Norma ISO 14443A*. Obtenido de NormasISO.org. <https://normasiso.org/norma-iso-14443a/>
- Peña, C. (2017). *Arduino de cero a experto. Proyectos prácticos. Electrónica, hardware y programación*. RedUsers.
- Peña, E. y Legaspi, M. G. (2020). Uart: A hardware communication protocol understanding universal asynchronous receiver/transmitter. *Visit Analog*, 54(4), 1-5.
- Piramuthu, S. (2011). RFID mutual authentication protocols. *Decision Support Systems*, 50(2), 387-393. <https://doi.org/10.1016/j.dss.2010.09.005>
- Qamarina, N., Jamaludin, A. A., Azizan, A. y Abas, H. (2018). Arduino vs Raspberry Pi vs Micro Bit: Platforms for Fast IoT Systems Prototyping. *Open International Journal of Informatics (OIJI)*, 6 (1).
- Reyes, F. y Cid, J. (2015). *Arduino. Aplicaciones en robótica, mecatrónica e ingenierías*. Marcombo. [https://books.google.es/books?hl=es&lr=&id=fExOEAAAQBAJ&oi=fnd&pg=PR1&dq=Reyes,+F.,+%26+Cid,+J.+\(2015\).+Arduino.+Aplicaciones+en+rob%C3%B3tica,+mecatr%C3%B3nica+e+ingenier%C3%ADas.+Marcombo.&ots=PZf6h5FjqD&sig=9u7FOwEvTAH_cEFAq9pXB1t4hlg#v=onepage&q&f=fal](https://books.google.es/books?hl=es&lr=&id=fExOEAAAQBAJ&oi=fnd&pg=PR1&dq=Reyes,+F.,+%26+Cid,+J.+(2015).+Arduino.+Aplicaciones+en+rob%C3%B3tica,+mecatr%C3%B3nica+e+ingenier%C3%ADas.+Marcombo.&ots=PZf6h5FjqD&sig=9u7FOwEvTAH_cEFAq9pXB1t4hlg#v=onepage&q&f=fal)
- Salazar, J. (s.f.). Clasificación de las redes inalámbricas [Fotografía]. TechPedia. https://upcommons.upc.edu/bitstream/handle/2117/100918/LM01_R_ES.pdf



- Strandhagen, J. W., Alfnes, E., Strandhagen, J. O. y Vallandingham, L. R. (2017). The fit of Industry 4.0 applications in manufacturing logistics: a multiple case study. *Advances in Manufacturing*, 5, 344-358.
<https://doi.org/10.1007/s40436-017-0200-y>
- Tubis, A. A. y Rohman, J. (2023). Intelligent Warehouse in Industry 4.0— Systematic Literature Review. *Sensors*, 23(8), 4105.
<https://doi.org/10.3390/s23084105>
- White, G., Gardiner, G., Prabhakar, G. P. y Abd Razak, A. (2007). A Comparison of Barcoding and RFID Technologies in Practice. *Journal of Information, Information Technology, and Organizations*, 2.
- Wyld, D. C. (2006). RFID 101: the next big thing for management. *Management Research News*, 29(4), 154-173.



ANEXOS



Desarrollo de un sistema de control de inventario en tiempo real mediante tecnología RFID





ÍNDICE DE ANEXOS

1. Programa del microcontrolador Arduino.
2. Archivo de Excel para el control del almacén.
3. Diagrama de las conexiones de Arduino con sus periféricos.