

**Universidad de La Laguna,  
Escuela Superior de Ingeniería  
y Tecnología**

Diseño e implementación de un  
sistema autónomo para la  
simulación de fugas en  
depósitos.

**Autor:**

Luis Arriaga Campos

**Tutores:**

Pedro Antonio Toledo Delgado

Marta Sigut Saavedra

# ÍNDICE

Resumen.....	1
Abstract.....	1

## CAPÍTULO I: CONSIDERACIONES INICIALES

1. Preámbulo y objetivos del proyecto.....	3
2. Reglamentos y normas de aplicación.....	5
3. Adecuación a la situación actual (Crisis COVID-19).....	6

## CAPÍTULO II: EL PROBLEMA DE LAS FUGAS DE COMBUSTIBLE EN LAS ESTACIONES DE SERVICIO

1. Funcionamiento de las estaciones de servicio reales.....	8
2. Sistemas de detección de fugas en estaciones de servicio.....	12
3. Impacto medioambiental generado por estas fugas.....	14
4. Antecedentes sobre casos de fugas de combustible.....	15

## CAPÍTULO III: PROTOTIPO DE LA PLANTA

1. Diseño de la planta.....	20
2. Materiales y componentes del prototipo físico.....	22

## CAPÍTULO IV: SIMULADOR DE LA PLANTA

1. Idea general.....	29
2. Funcionamiento del simulador de la planta.....	30
3. Electrónica para el acondicionamiento de señales.....	31

## CAPÍTULO V: CONTROLADORA

1. Funcionamiento e idea general.....	36
2. Estados que contempla la controladora.....	38
3. Acondicionamiento de las señales de entrada.....	42

## CAPÍTULO VI: SIMULACIONES Y CONCLUSIONES OBTENIDAS

1. Simulador utilizado y situaciones simuladas.....	45
2. Validación del simulador de la planta y la controladora.....	51
3. Conclusiones.....	56
4. Líneas de trabajo futuras.....	58

## ANEXOS

Anexo 1: Presupuesto.....	63
Anexo 2: Guía del desarrollador: La controladora.....	65
Anexo 3: Guía del desarrollador: Simulador de la planta.....	70

## REFERENCIAS

Referencias.....	74
------------------	----

## ÍNDICE DE FIGURAS

Figura 1.1: Estación de servicio de Montmeló causante de una gran fuga.....	3
Figura 2.1: Tanque de simple pared y cubeta de hormigón.....	8
Figura 2.2: Colocación de un tanque de doble pared.....	9
Figura 2.3: Proceso de llenado del tanque de almacenamiento de gasolina.....	10
Figura 2.4: Surtidor de combustible.....	11
Figura 2.5: Controlador de un sistema de detección de fugas del Grupo “inpro”.....	13
Figura 2.6: Contaminación por filtrado de combustible en el tanque de almacenamiento.....	14
Figura 2.7: Estación de servicio de Montmeló causante de una gran fuga.....	16
Figura 2.8: Río Jarama, afectado por una gran fuga de vertidos y combustible en la Comunidad de Madrid.....	17
Figura 3.1: Esquema de la planta diseñada.....	21
Figura 3.2: Boquilla de riego por goteo.....	23
Figura 3.3: Nebulizador de agua.....	24
Figura 3.4: Perfusor de anestesia.....	24
Figura 4.1: Esquema del filtro implementado.....	32
Figura 4.2: Comparación de la entrada y la salida del filtro para una señal PWM equivalente a 2.5 voltios.....	33
Figura 5.1: Evolución ideal del volumen de los depósitos.....	37
Figura 5.2: Máquina de estados de la controladora.....	38
Figura 5.3: Panel de control de la controladora.....	40
Figura 5.4: Máquina de estados de los depósitos.....	41
Figura 5.5: Función de rizado en la salida del filtro.....	43

Figura 6.1: Simulación de un depósito.....	46
Figura 6.2: Simulación de la planta sin el depósito de fugas.....	47
Figura 6.3: Simulación de la controladora sin el depósito de fugas.....	48
Figura 6.4: Simulación completa de la planta.....	48
Figura 6.5: Simulación completa de la controladora.....	49
Figura 6.6: Simulación conjunta sin fugas.....	50
Figura 6.7: Simulación completa.....	50
Figura 6.8: Datos referentes al volumen del tanque de recarga.....	52
Figura 6.9: Datos referentes al volumen del tanque de almacenamiento.....	53
Figura 6.10: Datos referentes al volumen del dispensador 1 .....	54
Figura 6.11: Datos referentes al volumen del tanque de fugas.....	55
Figura 6.12: Dispensadores de combustible con varios productos.....	59
Figura 6.13: Microcontrolador Raspberry Pi.....	60

## ÍNDICE DE TABLAS Y ECUACIONES

Tabla 3.1: Comparativa entre Arduino UNO, Arduino MEGA y Raspberry Pi 3.....	26
Tabla A.1: Presupuesto correspondiente a los materiales para la construcción del prototipo de la planta.....	63
Tabla A.2: Presupuesto correspondiente al trabajo de desarrollo del proyecto.....	64
Ecuación 4.1: Expresión de la frecuencia de corte de un filtro por condensador de primer orden.....	32
Ecuación 4.2: Expresión de la constante de tiempo en un filtro por condensador.....	33
Ecuación 5.1: Expresión que sigue el rizado en las señales de salida del filtro.....	43
Ecuación A.1: Expresión de conversión de tensión a volumen en la controladora.....	68
Ecuación A.2: Expresión de conversión de volumen a tensión en el simulador de la planta.....	73

## Resumen

Este proyecto está dedicado al desarrollo e implementación de un prototipo físico y de bajo coste de una planta que simula el llenado y vaciado de depósitos de una estación de servicio. La implementación software de su controladora soportada mediante un dispositivo de bajo coste (Arduino). Una simulación sobre un segundo prototipo embebido de la propia planta, con el objeto tanto de facilitar tanto el uso académico de la planta, como la investigación en la construcción de sistemas automáticos de detección de fugas. Cabe recalcar que este proyecto solamente abarca la simulación de estas fugas. La detección y eliminación de estas no se contempla en este trabajo. Dado que se pretende que el prototipo físico forme parte de un laboratorio de docencia, se ha intentado que su coste sea lo más bajo posible. Esto nos ha obligado a llevar a cabo una ardua labor de búsqueda de los componentes de la planta que, cumpliendo con los requerimientos técnicos necesarios, nos permitan alcanzar el objetivo de bajo coste.

A lo largo de este documento, se explica cómo es el funcionamiento de una estación de servicio real, así como los sistemas de detección de fugas que pueden implementarse y, además, se aportan datos sobre el impacto ambiental que estas fugas pueden suponer. Se plantea también cuál era la idea inicial del proyecto y cómo ha variado debido a la situación en la que nos encontramos actualmente (Crisis COVID-19).

Se aportan otros datos como presupuestos, opciones de diseño y códigos realizados para el control de la planta, además de las simulaciones realizadas para comprobar el funcionamiento de esta.

## Abstract

This project is dedicated to the development and implementation of a low-cost, physical prototype of a plant that simulates the filling and emptying of tanks at a service station. The software implementation of its controller supported by a low cost device (Arduino). A simulation of a second prototype embedded in the plant itself, in order to facilitate both the academic use of the plant and research into the construction of automatic leak detection systems. It must be said that this project will only encompass the leaks simulation. Leaks detection and elimination does not enter in the contents of this elaboration. Since the physical prototype is intended to be part of a teaching laboratory, it has been tried to make its cost as low as possible. This has forced us to carry out an arduous search for the components of the plant that, complying with the necessary technical requirements, allows us to achieve the objective of being low cost.

Throughout this document, it will be explained how a real service station works, as well as how a leak detection system may be implemented. In addition, data on the environmental impact that these leaks may cause. It will also be considered what the initial idea of the project was and how it has changed due to the situation we are currently in (COVID-19).

Other data such as budgets, design options and codes made for the control of the plant are provided, in addition to the simulations carried out to check the operation of the plant.



# CAPÍTULO I: CONSIDERACIONES INICIALES

## 1. Preámbulo y objetivos del proyecto

Actualmente, una de las fuentes de energía más usadas sigue siendo los hidrocarburos. Pese a que en un futuro próximo posiblemente dejen de utilizarse, hoy en día siguen siendo motivo de estudio, no solo por sus propiedades energéticas, sino por el impacto ambiental que generan.

El almacenamiento subterráneo de hidrocarburos es una práctica habitual en instalaciones de suministro como Estaciones de Servicios a vehículos. Sin embargo, esta actividad no está exenta de riesgos, como por ejemplo la potencial filtración de hidrocarburos al subsuelo debido a la corrosión de los depósitos de almacenamiento. Existen casos de estaciones de servicio que han sido investigadas por incidentes que han generado serios problemas medioambientales. Por ejemplo, dos gasolineras de Montmeló, Cataluña, estuvieron filtrando inconscientemente combustible a los acuíferos de la zona en 2011, [2].



Figura 1.1: Estación de servicio de Montmeló causante de una gran fuga.

A partir de esta realidad, se planteó llevar a cabo este proyecto, creando un modelo reducido de una estación de servicio para simular dichas fugas. Teniendo esto en cuenta,

se plantearon los siguientes objetivos como metas a cumplir durante el desarrollo del proyecto:

- Plantear cuál sería el diseño y composición que tendría el sistema que se desea construir.
- Construir un prototipo físico del sistema que se ha diseñado.
- Conseguir que el prototipo fuera construido con materiales lo más económicos posible.
- Implementar una controladora que gestione el funcionamiento del prototipo.
- Desarrollar un simulador que se comporte como el prototipo de la planta.
- Validación de todos los elementos nombrados anteriormente mediante una simulación de su funcionamiento.

Este Trabajo de Fin de Grado surgió con un doble propósito. Por un lado, tiene una vertiente académica ya que tanto el prototipo físico que se pensaba construir en un principio como el sistema simulado que se ha implementado en su lugar formarán parte de un laboratorio de docencia. Por otro lado, un grupo de investigadores del departamento de Ingeniería Informática y de Sistemas, entre los cuales se encuentran los tutores de este trabajo, llevan algunos años trabajando en la detección de fugas de combustible en estaciones de servicio. Fruto de este trabajo se han obtenido dos publicaciones en revistas del primer cuartil del JCR (Journal Citations Report), [3] y [10]. En estos artículos los autores aplican técnicas de clasificación para la detección de fugas, unas técnicas que requieren disponer de un gran volumen de datos. Hasta ahora se ha trabajado con datos provenientes de estaciones de servicio reales, concretamente de los libros de inventario. Estos libros son registros donde quedan reflejados, para cada día de funcionamiento de la estación, todos los movimientos de combustible que se producen: descargas de la cuba en el depósito de almacenamiento, ventas a través de los surtidores, etc. Esto genera ciertos problemas como los que se comentan a continuación:

- No es sencillo conseguir datos provenientes de libros de inventario de estaciones de servicio reales, ya que las empresas son reticentes a facilitar este tipo de información.

- En el caso de los datos con los que se ha trabajado hasta ahora los libros de inventario se actualizan una vez al día, lo cual supone una limitación importante.
- A veces se cometen errores al anotar en el libro de inventario los movimientos de combustible en la estación y se reflejan en un día distinto del que se producen (por ejemplo, se anota la descarga de la cuba el día siguiente de cuando realmente tuvo lugar). Esto provoca desfases en los datos que obligan a que se tengan que ‘limpiar’ antes de utilizarlos para la detección de fugas. De no hacerlo se provocarían falsas alarmas. El hecho de disponer de un simulador para generar datos sintéticos permite resolver estos problemas:
  - En primer lugar, se podrá disponer de un número ilimitado de datos correspondientes tanto a días sin fuga como a días con fuga y, en este último caso, para distintos tipos y ratios de fuga. Asimismo, la fuga se podría provocar en el momento que se desee.
  - Además, se podrá generar datos con la periodicidad deseada (X datos al día), lo que resulta de gran utilidad a la hora de analizar el cumplimiento de la normativa vigente sobre detección de fugas [1].
  - Por último, los libros de inventario generados por el simulador estarán exentos de los errores y desfases que caracterizan los registros de estaciones reales y, por tanto, será innecesaria la etapa de filtrado o limpieza de los datos.

## 2. Reglamentos y normas de aplicación

Durante la realización del proyecto, se ha consultado la normativa vigente relativa a la detección de fugas de combustible en estaciones de servicio. Concretamente:

[1] UNE-EN 13160-5, Sistemas de detección de fugas. Parte 5: Requisitos y métodos de ensayo/evaluación de los sistemas de detección de fugas en tanques con indicador de nivel y en los sistemas de tuberías a presión (marzo 2017).  
<<https://www.en.aenor.com/normas-y-libros/buscador-de-normas/une?c=N0058060>>.

[2] Real Decreto 706/2017, de 7 de julio, por el que se aprueba la instrucción técnica complementaria MI-IP 04 "Instalaciones para suministro a vehículos" y se regulan

determinados aspectos de la reglamentación de instalaciones petrolíferas. <[https://www.boe.es/diario\\_boe/txt.php?id=BOE-A-2017-9188](https://www.boe.es/diario_boe/txt.php?id=BOE-A-2017-9188)>.

[3] Real Decreto 2085/1994, de 20 de octubre, por el que se aprueba el Reglamento de Instalaciones Petrolíferas. <<https://www.boe.es/buscar/doc.php?id=BOE-A-1995-2122>>.

Además de esta normativa, se han seguido las normas definidas por la asignatura “Trabajo de Fin de Grado” de la Universidad de La Laguna, así como la guía docente asociada a esta.

### **3. Adecuación a la situación actual (Crisis COVID-19)**

En un principio, el proyecto tenía como objetivo final la construcción del modelo físico de la planta donde se querían simular las fugas, y a partir de él, obtener los datos de la variación del volumen del depósito de almacenamiento para, a partir de ellos, llevar a cabo alguna técnica de detección de fugas. Todo esto implica un proceso de presupuestado, selección de componentes y de fabricación, así como las pertinentes pruebas y testeos para asegurar el funcionamiento del modelo físico. Sin embargo, debido a la crisis sanitaria que se está viviendo actualmente, no se consiguieron los materiales para que esta fase se llevara a cabo. Los distribuidores de dichos materiales cancelaron los pedidos porque su país, China, fue uno de los primeros en sufrir esta crisis. Pese a estas circunstancias, se va a mostrar toda la documentación referente al presupuesto y a los materiales, ya que se considera que, aunque no se haya podido llevar a cabo esta etapa, sigue siendo parte del proyecto. Además, es un trabajo al que se dedicó bastante tiempo y esfuerzo y el punto de partida de la futura construcción del prototipo físico.

Por lo que se acaba de comentar, este Trabajo de Fin de Grado se ha tenido que adaptar a las circunstancias en las que nos encontramos. Al no poder obtener los materiales, se ha decidido priorizar la construcción del simulador frente al prototipo físico y así obtener directamente los datos de él. Éste envía la información de los sensores simulados a otro microcontrolador en el que se ha implementado la controladora, la cual, en función de dicha información y de la lógica programada, envía las órdenes apropiadas a los actuadores del sistema.



## CAPÍTULO II: EL PROBLEMA DE LAS FUGAS DE COMBUSTIBLE EN LAS ESTACIONES DE SERVICIO

### 1. Funcionamiento de las estaciones de servicio reales

Uno de los elementos fundamentales en las estaciones de servicio son los depósitos de almacenamiento de combustible. Estos depósitos presentan dos tipologías:

- De simple pared: de plástico reforzado o acero como material de fabricación.
- De doble pared: presentando entre ellas un espacio vacío.

Los tanques de simple pared se entierran dentro de un foso común o cubeto estanco de hormigón armado. Esta es una técnica utilizada para que, si se produce alguna fuga, esta quede contenida en la cubeta de hormigón y no se exponga al medio. A esto hay que añadirle la adición de un tubo vertical que hace de visor para comprobar que no se presentan fugas en el fondo de la cubeta.



Figura 2.1: Tanque de simple pared y cubeta de hormigón.

En los depósitos de doble pared, las posibilidades de fugas se reducen debido a su morfología, pero debido a eso, la normativa les exige que se instale un sistema de detección de fugas para monitorear que no se han producido grietas en ambas paredes y, por tanto, fugas de combustible. Además, la cámara que queda entre las dos paredes permite detectar inmediatamente si hay un escape. Esta es una manera eficiente de eliminar las fugas de combustible al suelo y las pérdidas potenciales de combustible.



Figura 2.2: Colocación de un tanque de doble pared.

Las estaciones de servicio evitan en todo momento que se produzca contacto directo entre los productos que van a vender. La solución para esto es disponer de un tanque de almacenamiento por producto que ofrezcan a la venta. Debido a esto, el proceso de recarga de estos depósitos también debe realizarse teniendo en cuenta que producto se va a descargar en estos tanques, [8].

En la Figura 2.3 podemos observar el proceso que se sigue para recargar el tanque de almacenamiento desde un camión cisterna.



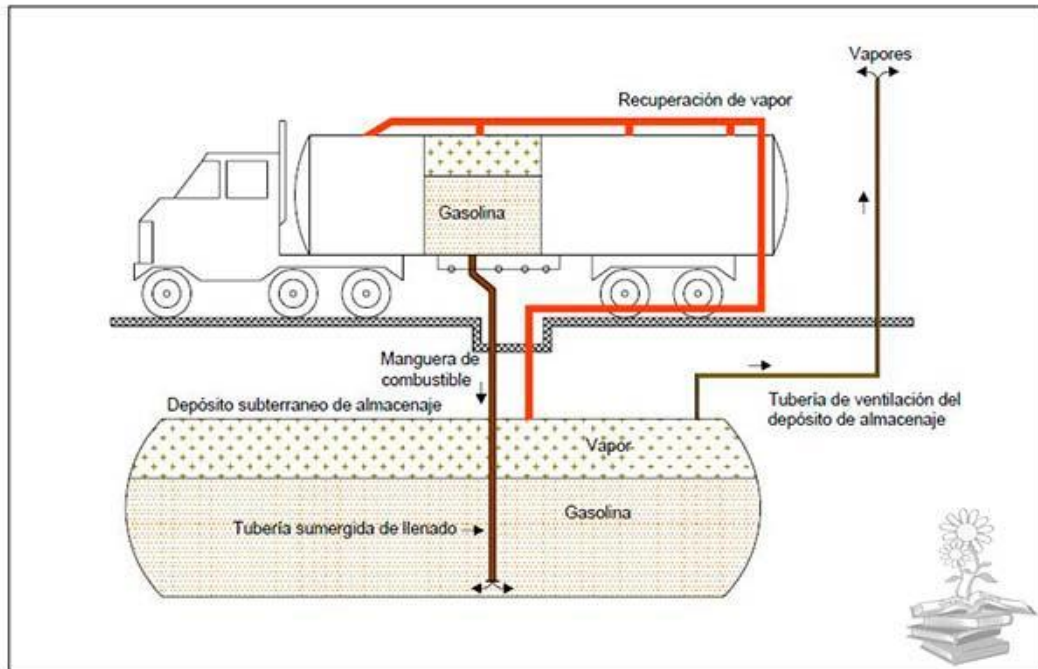


Figura 2.3: Proceso de llenado del tanque de almacenamiento de gasolina.

Otro elemento esencial en las estaciones de servicio son los surtidores. Estos se pueden clasificar en función del caudal, de la posición del cabezal y del número de combustibles que suministran.

El almacenamiento del combustible se lleva a cabo en tanques subterráneos en la estación de servicio, por lo que para llevar el combustible al vehículo que está repostando, la clave de este proceso se encuentra en las bombas con las que cuentan estas estaciones. Estas pueden ser sumergibles o de succión:

- Las bombas sumergibles están, claramente, sumergidas en el fondo depósito de combustible y se basan en una hélice que permite que el líquido suba hasta el surtidor que demande el combustible.
- Las bombas de succión usan el principio de la presión desigual. Lo que ocurre es que el motor que se encuentra por encima del combustible y, a través de la eliminación de aire en la tubería, logra el ascenso de este haciendo que cuando la presión del aire llega a su



punto más bajo, el combustible comienza a subir a la superficie. La válvula de retención es la responsable de impedir que, al apagarse el motor de la bomba, el combustible regrese al tanque subterráneo, todo esto mediante la creación de un cierre hermético.

Finalmente, el medidor de flujo entra en juego mientras el combustible está en las tuberías. Este es el elemento que nos permite repostar la cantidad que hemos solicitado. El medidor de flujo cuenta con una membrana de plástico, que comenzará a comprimirse a medida que el límite de la cantidad de combustible a repostar se aproxime. Cuando el medidor de flujo de un surtidor de combustible se desgasta, la eficiencia durante el repostaje disminuye de forma considerable, [7].



Figura 2.4: Surtidor de combustible.

## 2. Sistemas de detección de fugas en estaciones de servicio

A pesar de que los tanques de doble pared son muy seguros, es preciso instalar un sistema de detección de fugas de combustible. Estos sistemas pueden ser de varios tipos:

- Sistema de detección de fugas por líquido eléctricamente conductor situado entre las dos paredes del depósito: En esta modalidad, mediante la introducción de un electrodo, se controla la cantidad de líquido intersticial que presenta el tanque. Con este sistema somos capaces de detectar inmediatamente una fuga, pero un gran inconveniente es que se transmite electricidad al depósito. Normalmente se descarta la instalación de este tipo de sistema de detección por transmitir electricidad a los tanques.
- Sistema de detección de fugas por vacío: Es un método que nos permite controlar el vacío que se presenta entre las dos paredes del depósito. En comparación con el anterior, este sistema nos permite inspeccionar la cámara sin tener que abrir el depósito y no transmite electricidad a los depósitos.
- Sistema de detección de fugas por vacío con regeneración: Este sistema es idéntico al anterior, pero con un microprocesador que diferencia entre una pérdida de vacío ocasional o una fuga.
- Sistema de detección de fugas y control de existencias: Este equipo, además de detectar las posibles fugas de combustible, controla todo tipo de información sobre el combustible almacenado, como pueden ser existencias de cada producto, densidades, nivel de agua, consumo por hora, datos sobre las últimas operaciones de carga, prueba para detección de fugas, informes de gestión, etc. Las mediciones se basan en el principio hidrostático (medición de vacío o presión) con unos sensores en el interior del tanque y en la cámara intermedia, con la ventaja de no transmitir electricidad a los tanques. También se puede visualizar la cámara intersticial sin tener que abrir el depósito. Todos estos sistemas tienen un equipo, situado en un lugar a la vista del personal de la estación de servicio, que daría señales luminosas y acústicas en caso de detectar algún escape de combustible.

Uno de los principales motivos por los que hay que realizar reparaciones o verificaciones de este tipo de depósitos es la aparición de corrosiones o picaduras en el acero que lo forman. Los depósitos llevan una capa de protección contra la corrosión, denominada protección pasiva. Pero esta solución no es suficiente porque cualquier defecto en este aislamiento conduce a un proceso muy local, pero muy virulento, generando picaduras con velocidades de penetración muy rápidas. Además, el proceso de transporte e instalación de estos depósitos hace que sea muy fácil la aparición de estos deterioros en las capas de protección. También se ha de tener en cuenta que esta capa, por sí sola, tiene eficacia durante, aproximadamente, 10 años. Para alargar la vida de estos tanques debe implementarse la denominada protección activa. Esta protección está basada en la introducción de un monitoreo de la corrosión ya que, en un futuro, esta corrosión puede ser la causa de una futura fuga de combustible, [8]. En la siguiente figura podemos observar un dispositivo de detección de fugas, el cual posee un sistema de protección activa contra corrosión. Este dispositivo corre cuenta del Grupo “inpro”.



Figura 2.5: Controlador de un sistema de detección de fugas del Grupo ‘inpro’.

### 3. Impacto medioambiental generado por estas fugas

Las fugas de combustible en tanques subterráneos pueden llegar a causar graves problemas ambientales debido a la toxicidad de los hidrocarburos. Una de las posibles situaciones es la filtración a través del subsuelo mientras estos combustibles son transportados, lo cual genera una contaminación en las aguas subterráneas que pueden derivar en peligrosos accidentes para la comunidad cercana, siempre y cuando utilicen este medio como fuente de consumo de agua. Por otro lado, el hecho de producir los combustibles también es otro factor de riesgo capaz producir una catástrofe medioambiental y que hay que vigilar con cuidado con el fin de evitar problemas mayores.

El proceso que siguen estos líquidos cuando se filtran es el siguiente: el primer paso es cuando entran en contacto con el suelo contaminando la propia tierra, el segundo es cuando penetra en el subsuelo y entra en contacto con las aguas subterránea o los acuíferos. Esto produce un impacto medioambiental gravísimo, pudiendo intoxicar a la población cercana.

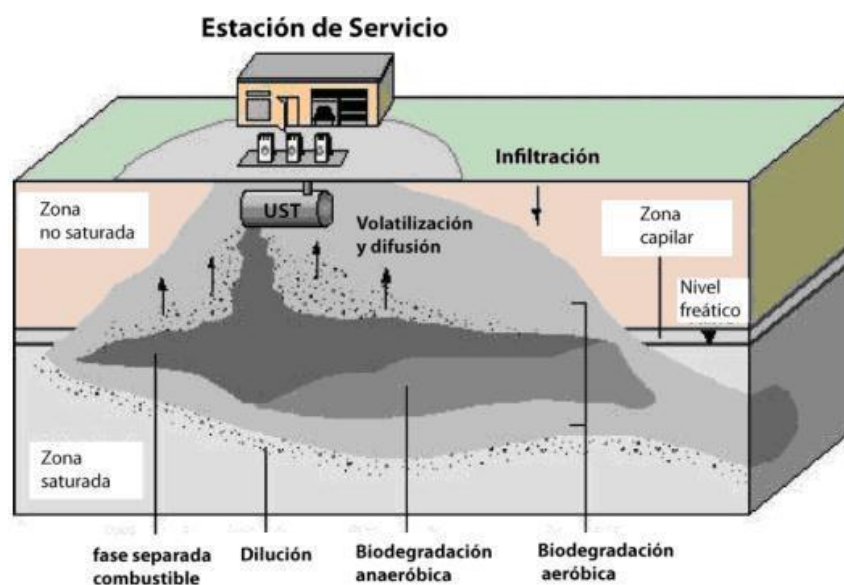


Figura 2.6: Contaminación por filtrado de combustible en el tanque de almacenamiento.

Pero estas fugas causan otro tipo de contaminación aparte de la ya nombrada. Recordemos que estos hidrocarburos generan gases tóxicos que son expulsados, contaminando así el aire cercano a la fuga. Estos gases son difíciles de detectar y son muy nocivos tanto para la atmósfera como para los seres vivos cercanos a la zona de la fuga. También es posible que se presenten otras emisiones evaporativas como las presentes en el diésel y Keroseno (destilación del petróleo) pero con menor porcentaje de presiones de vapor, ya que la mayor parte de las emisiones producidas en las estaciones de servicio es debido a la gasolina.

En el caso de contaminación producida en los dispensadores de combustible hay dos procesos por el cual las emisiones se identifican claramente: el primer proceso es por causa de los escapes de gases provenientes del propio dispensador, por la gasolina cargada hasta la salida de la estación y el otro proceso por los derrames de estos contaminantes. También pueden producirse derrames a la hora de llenar el depósito de un vehículo que esté repostando, pudiendo causar un accidente mayor si no se llega a retirar cuanto antes, [9].

Estos son algunos ejemplos de los problemas que puede generar una fuga y un mal uso de las instalaciones de una estación de servicio, y es importante tenerlos en cuenta para resaltar la importancia del proyecto que se está presentando.

#### **4. Antecedentes sobre casos de fugas de combustible**

Para entender mejor la importancia de evitar este tipo de contaminación por parte de las estaciones de servicio, se va a mostrar a continuación una recopilación de noticias que muestran la problemática que conllevan las fugas de combustible para nuestro entorno.

- **Caso de la estación de servicio de Montmeló:** En diciembre de 2011 salió publicado en el diario El País, el caso de cuando se produjo una filtración en una gasolinera de Repsol en Montmeló. Los vecinos observaron que el agua del grifo tenía “un olor diferente del habitual” y alertaron al Ayuntamiento. El problema se originó en el túnel de lavado de la estación de servicio, que captaba agua indistintamente de un acuífero y de la

red de suministro municipal. Ambas conexiones estaban reguladas por una válvula que debía impedir el trasvase de una a otra. Se produjo el error de filtrar todos estos desechos del lavado a la red suministro municipal. La empresa reparó rápidamente la instalación y desde entonces toma el agua de un depósito interno para garantizar la seguridad de los vecinos. La ACA (Asociación Catalana del Agua) considera que el riesgo para los vecinos fue mínimo, pero denunció el caso a la fiscalía, [5].



Figura 2.7: Estación de servicio de Montmeló causante de una gran fuga.

- **Caso de la estación de servicio de Santa Perpetua:** En marzo de 2011, el diario El País nos cuenta otro caso en la misma comunidad. El color del agua y el olor que provenía de los pozos, cada día más molesto fueron los que alertaron de que algo estaba sucediendo. El incidente ocurrió cuando la petrolera BP inició unas obras de reparación en la gasolinera del Pont Vell y agujereó por accidente una de las cañerías. El vertido, de cerca de 9.000 litros de gasolina, se extendió a lo largo de medio kilómetro por el subsuelo. Poco después se observó que también había alcanzado algunos pozos del acuífero cercano a la zona. El fuerte olor a disolvente y el cambio de color de las aguas en piscinas y pozos fueron pruebas suficientes para iniciar una investigación. Los afectados alertaron al Ayuntamiento, que abrió un expediente a la empresa y alertó a la ACA. Técnicos de la ACA tomaron muestras del agua y reclamaron a la petrolera que



actuara ante la grave situación. La empresa admitió su responsabilidad y afirmó que los trabajos estarían terminados en marzo de 2012. BP abrió 30 pozos en la zona para analizar las aguas y extraer el combustible, [5].

• **Caso de la estación de servicio de Madrid que soltó vertidos en el río Jarama:** El periódico *Europapress* nos informa que desde el año 1993, una gasolinera del kilómetro 24 de la N-I vertió aguas residuales con hidrocarburos directamente al colector sin tratamiento previo. Tanto la fiscalía como las acusaciones defienden que, además, los tanques donde se almacenaba la gasolina sufrían pérdidas que se fueron filtrando al subsuelo durante años como demuestra la presencia de gasolina sin plomo en las aguas analizadas en varios pozos afectados. El vertido contaminó el subsuelo y la capa freática de la tierra colindante y, por su volumen, ocasionó la contaminación del acuífero que se encontraba bajo la gasolinera y del que se abastecían las industrias y vecinos de la zona. El acuífero contaminado por la invasión de los hidrocarburos también abastecía a diversos establecimientos de hostelería abiertos al público, los que provocó un gran riesgo para la salud de la gente que frecuentaba la zona. Las afecciones ambientales y sanitarias que sufren esas aguas se han considerado finalmente como irreparables, y exigirían grandes esfuerzos económicos, [6].



Figura 2.8: Río Jarama, afectado por una gran fuga de vertidos y combustible en la Comunidad de Madrid.

Estos son solo algunos de los casos en los cuales una fuga de combustible ha generado un gran daño ambiental, y solo se han tenido en cuenta casos a nivel nacional. Se han incluido en esta memoria con el fin de hacer ver la importancia que tiene el evitar que se produzcan accidentes medioambientales como los nombrados anteriormente. Este proyecto puede ser de ayuda para este campo, ya que al poseer un modelo que simule estas fugas, puede hacer que otros prueben sus sistemas de detección de fugas en él.





## CAPÍTULO III: PROTOTIPO DE LA PLANTA

### 1. Diseño de la planta

A la hora de plantearnos la construcción de una planta que representase el funcionamiento de una estación de servicios, incluyendo la existencia de posibles fugas de combustible, lo hicimos en base a dos premisas. La primera, que contuviera todos los elementos básicos de una estación real de forma que los datos obtenidos sirvieran para el propósito que se establece este documento. La segunda premisa era que el prototipo implementado presentase unas características (tamaño, coste, etc.) que permitiese su instalación en un laboratorio de docencia. Teniendo en cuenta esto, el prototipo físico, que no se ha podido construir por las razones ya comentadas, responde al esquema que se muestra en la figura 3.1.

Como es lógico, el modelo no va a representar una estación de servicio en su totalidad, ya que la complejidad superara el alcance de este proyecto. Así que el trabajo se ha centrado en lo que nos interesa, que es la simulación de fugas en el depósito de almacenamiento. En particular, se ha decidido implementar un único tanque de almacenamiento, un depósito que simule el camión cisterna responsable de las descargas de combustible, dos dispensadores para de las ventas, un tanque encargado de recoger las fugas que se vayan generando y un depósito que almacena todas las descargas de los dispensadores y envía fluido al tanque equivalente al camión cisterna. Cada uno de los depósitos representados en la Figura 3.1 tiene una representación software.

A continuación, se muestra el esquema de la planta que se ha diseñado junto a la explicación de las partes que la conforman. Cabe destacar que los nombres utilizados en la Figura 3.1 son los que se han usado a nivel software y a lo largo de este documento se referirá a los depósitos por dichos nombres:

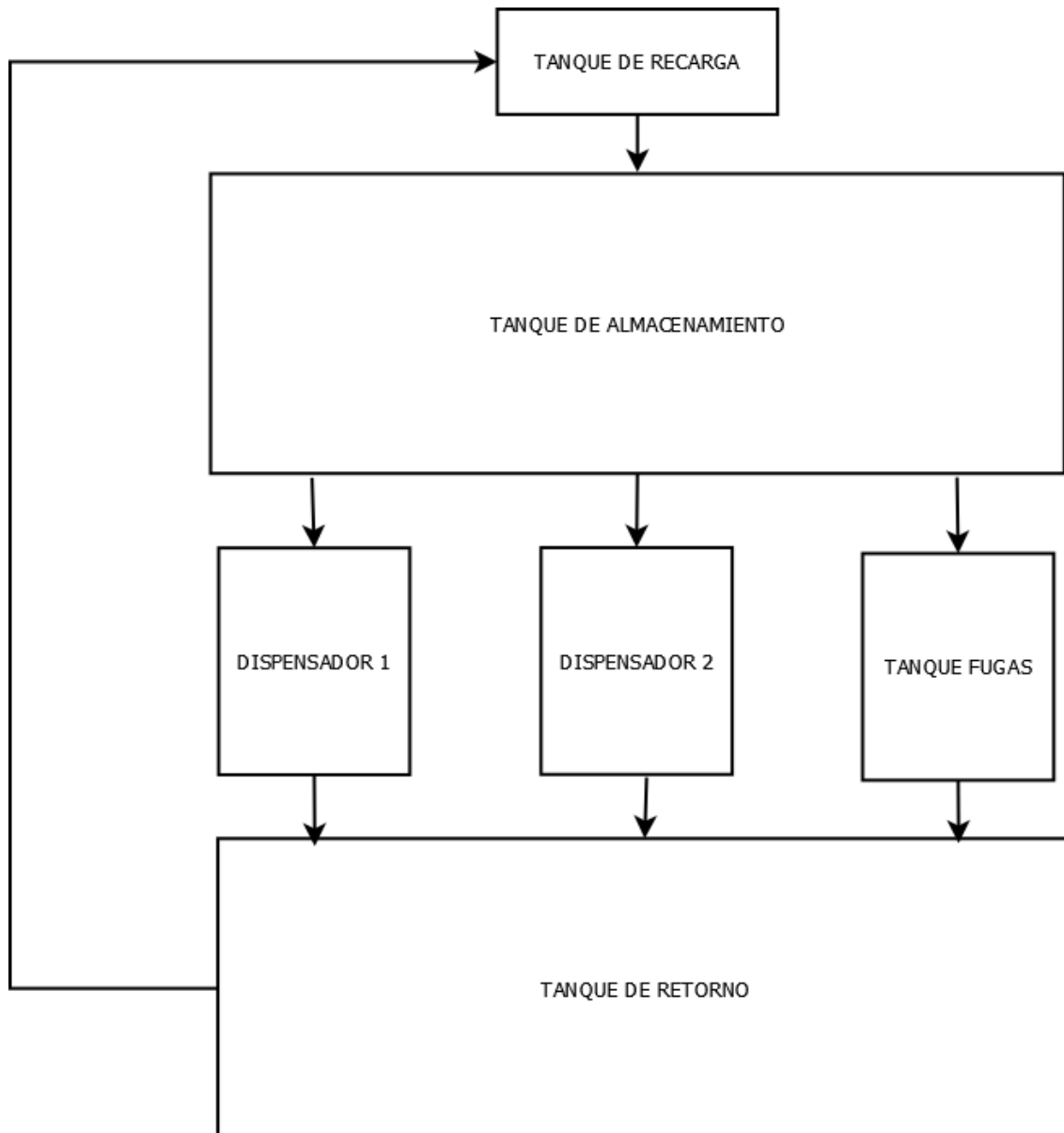


Figura 3.1: Esquema de la planta diseñada.

Debido a razones obvias, en este modelo no se usa combustible de verdad debido a los problemas de seguridad que podrían ocasionarse. Además, no poseemos la licencia para trabajar con este tipo de fluidos. En su lugar usamos agua como líquido de trabajo.

- **Tanque de recarga:** Este depósito es el equivalente al camión cisterna que recarga regularmente el tanque de almacenamiento de agua.

- **Tanque de almacenamiento:** Este depósito es, como su propio nombre indica, el que está soterrado y, por lo tanto, es donde se van a producir las fugas, cuyo caudal va a depender del volumen del tanque de almacenamiento.
- **Dispensadores:** Simulan las ventas al público. Son tanques de poco volumen que tienen un comportamiento muy activo en la simulación.
- **Tanque de fugas:** Es el recipiente donde se almacena toda el agua que se fuga del tanque de almacenamiento. Tiene la misma capacidad que los dispensadores.
- **Tanque de retorno:** Este depósito no tiene un equivalente real en una estación de servicio, simplemente se ha añadido al diseño para tener un tanque que reciba toda el agua que evacúan los dispensadores y el tanque de fugas. En caso de tener que reiniciar el sistema, se da la orden de apertura a todas las válvulas para que todos los niveles superiores se vacíen. Por tanto, la capacidad del tanque de retorno debe ser tal que pueda contener todo el líquido de los otros depósitos. Como se puede intuir en la Figura 3.1, las flechas representan el sentido que sigue el agua y el número de estas nos indica cuántas entradas o salidas tienen cada uno de los depósitos.

## 2. Materiales y componentes del prototipo físico

A pesar de que no se ha podido construir el prototipo físico, durante las primeras semanas del desarrollo del TFG se estuvo trabajando tanto en el diseño del mismo (ver Figura 3.1) como en la elección los materiales y componentes que debía incluir, y que se describen en este apartado.

El movimiento del agua entre depósitos se produce mediante mangueras que los conectan y una válvula solenoide que permite o impide el paso del fluido. Se asume que la presión ejercida por la gravedad es suficiente para que el agua pase de un tanque a otro, ya que la estructura es vertical. El tanque ubicado al nivel del suelo (tanque de retorno) recoge toda el agua que evacúan los otros tanques, y dispone de una bomba de corriente continua que la devuelve al nivel superior de la estructura.

En cuanto a los sensores, las variables que hay que medir son fundamentalmente los volúmenes de los depósitos. En primera instancia se pensó en usar sensores de nivel para determinar el volumen de agua en cada uno de los depósitos sin tener en cuenta el tanque de retorno, ya que no nos interesa monitorizar su volumen. Pero debido a su precio se descartó la opción. En su lugar, se decidió medir la masa de agua contenida en cada uno de los tanques. A partir de ese valor, y teniendo en cuenta que la densidad del agua es 1 Kg/L, su masa coincide aproximadamente con su volumen. Teniendo esto en cuenta, se decidió usar básculas para la medición de la masa de los depósitos y determinar así su volumen.

Una de las partes más complicadas en cuanto a la obtención de datos es la medición del caudal que se fuga del tanque de almacenamiento. Se barajaron diferentes opciones debido a que necesitamos un caudalímetro capaz de medir caudales del orden de ml/min. Los dispositivos disponibles en el mercado capaces de medir caudales de este orden de magnitud tienen un coste bastante elevado para un prototipo de estas características, ya que fácilmente podríamos superar los 1500 euros. Por eso se buscaron alternativas más baratas:

- **Sistema de riego por goteo:** se planteó la opción de usar un sistema de riego por goteo para obtener el caudal deseado para las fugas. Se encontró el problema de que es un sistema difícil de automatizar, ya que en su mayoría es manual. Además, el caudal mínimo que proporciona un dispositivo como este es de 133,33 ml/min, mayor que el que se precisa.



Figura 3.2: Boquilla de riego por goteo

- **Nebulizadores de agua:** otra idea fue usar nebulizadores para obtener el caudal de fugas que necesitamos. Estos sistemas son muy usados en el ámbito de la restauración para refrescar los establecimientos mediante una bruma de agua que no llega a mojar y enfría el aire. Una vez más, el caudal obtenido no era lo suficientemente pequeño para la aplicación que nos ocupa.



Figura 3.3: Nebulizador de agua

- **Perfusor de anestesia:** este aparato es utilizado en el ámbito médico para introducir la dosis exacta de anestesia en el paciente. Al ser una operación de gran precisión, necesita suministrar un caudal muy concreto. Finalmente, se optó por esta alternativa para obtener el caudal de fugas en nuestro modelo, ya que es capaz de proporcionarnos un caudal de 0.16 ml/min. Se tenía pensado conseguirlo de segunda mano para que el prototipo siguiera siendo de bajo coste.



Figura 3.4: Perfusor de anestesia

En cuanto a la capacidad de los distintos depósitos del prototipo físico, se decidió usar botellas de agua de litro y medio para los dispensadores y el tanque de fugas, una garrafa de 20 litros para el tanque de recarga y, finalmente, un depósito industrial de 150 litros para el tanque de almacenamiento. Este depósito tendría un peso aproximado de 6 kg debido a sus grandes dimensiones (1.2 m de largo, 0.45 m de ancho y 0.28 m de altura), pero se buscó la alternativa que, cumpliendo con las especificaciones que necesitábamos, fuera lo más barata posible (111 €). Los depósitos anteriores están conectados entre sí tal y como se muestra en la figura 3.1. Tal y como se comentó anteriormente, el tanque de retorno debe tener la capacidad suficiente como para albergar el agua contenida en el resto de los depósitos de la planta, razón por la que se decidió utilizar una cubeta capaz de cumplir este requerimiento. Además de estos depósitos, necesitamos también los siguientes materiales:

- **Válvulas solenoides:** son las encargadas de permitir el paso del agua entre los depósitos. Se escogieron unas válvulas ‘todo-nada’ que necesitan un impulso de 12 voltios para articular el cierre. Debido a esto necesitan una presión mínima de entrada de 0.8 MPa, lo cual justifica que nuestro sistema esté alimentado por gravedad (1.01 MPa).
- **Mangueras:** conectores por los que circula el agua entre depósitos. Estarán conectadas mediante las válvulas nombradas anteriormente.
- **Bomba de reposición:** con esta bomba se envía el agua desde el tanque de retorno al tanque de recarga. Es capaz de dar al fluido una altura de 3 metros, suficiente para llevarlo a lo alto de la estructura. También nos proporciona un caudal de 3000L/h y deberá alimentarse con 220 V de alterna, es decir, puede conectarse a la red eléctrica del laboratorio.
- **Bomba de fugas:** una bomba que nos permite establecer con precisión el caudal de fugas gracias a la presión que nos proporciona. Esta pequeña bomba requeriría de 12 voltios de continua como alimentación. La presión que nos proporciona a la salida es de 0.8 Bares, suficiente para ayudar al Perfusor a generar el caudal demandado.
- **Báscula del tanque de almacenamiento:** como el tanque de almacenamiento tiene una capacidad de 150 l, para poder obtener el volumen del agua que contiene a partir de la

medida de la masa es necesario disponer de una báscula capaz de llegar hasta los 150kg. Su precisión sería de 0.1 kg, suficiente teniendo en cuenta las cantidades que se están manejando. Sus dimensiones son de 40x35 cm.

- **Básculas para ventas y fugas:** usadas como sensores de masa para poder determinar el volumen de agua que hay en los depósitos. Estas básculas tienen una precisión de 0.01 g. Debido a que los depósitos en los cuales se utilizarán estas básculas llegarán a pesar como máximo 1.5 kg, esta precisión nos dará una certeza más que aceptable en la medida. Sus medidas son de 5x4.1 cm, ocupando un espacio mínimo en el modelo.

- **Módulo de relés:** son los encargados de regular el paso de corriente a las válvulas, siendo estos los que las activan o las desactivan. Se polarizan con una tensión continua de 5 voltios.

- **Arduino UNO:** es el microcontrolador que se usa para la controladora y para el simulador de la planta debido a sus características. Estos componentes presentan una serie de ventajas frente a otros microcontroladores que podemos encontrar en el mercado. En la tabla 3.1 se presenta una comparativa entre el Arduino UNO y otros dos microcontroladores muy demandados en el mercado, como son Arduino MEGA y Raspberry pi 3:

**Tabla 3.1: Comparativa entre Arduino UNO, Arduino MEGA y Raspberry Pi 3:**

Características	Memoria Flash	Entradas/Salidas Digitales	Entradas Analógicas	Precio
Arduino UNO	32 Kb	14	6	5€
Arduino MEGA	256 Kb	56	16	15€
Raspberry Pi 3	1Gb	27	13 (Audio y vídeo)	40€

Como se puede observar en la tabla anterior, el dispositivo que mejores prestaciones ofrece en términos generales es Raspberry Pi 3. A su vez, también es el que tiene un coste más elevado. El grueso de nuestro código ocupa solamente 6 Kb y necesitamos un total



de 13 pines digitales y 5 entradas analógicas. Como Arduino UNO cumple con dichas especificaciones, y además es el hardware más económico, ha sido la opción que se ha seleccionado para trabajar en este proyecto.

- **Estantería:** soporte en el que reposa toda la estructura de la planta, en un principio, de 2 metros de altura.

- **Fuentes de alimentación independiente:** se encargan de la alimentación de las válvulas y de los microcontroladores. Estas fuentes variables serán capaces de darnos los 12 voltios que necesitan las válvulas y los 5 que necesitan los microcontroladores Arduino.

Además de estos materiales, también se ha de tener en cuenta la electrónica de acondicionamiento que se ha usado para tratar las señales (Capítulo IV, apartado 4), la cual ha sido considerada como material fungible. El presupuesto correspondiente al prototipo físico se incluye en el apartado de ANEXOS, concretamente el Anexo 1.



## CAPÍTULO IV: SIMULADOR DE LA PLANTA

### 1. Idea general

El simulador de la planta es el elemento que nos permite probar la controladora antes de conectarla al modelo real, y así comprobar su funcionamiento. Este código simula el comportamiento de los depósitos que conforman el modelo y simula los volúmenes de cada uno, teniendo en cuenta las pérdidas y ganancias de agua que se producen a lo largo de la simulación.

Una cosa a tener en cuenta es que el código implementado se ha desarrollado mediante una programación orientada a objetos, tomando como clase el estándar de depósito. Los objetos implementados han sido los propios tanques que forman la planta. Se tomó esta decisión debido a las numerosas ventajas que esta filosofía de programación nos trae a la hora de trabajar en un proyecto como este. En primer lugar, nos ofrece reusabilidad. Esto es bastante interesante por si en un futuro se quiere ampliar el sistema añadiéndole más depósitos; este tipo de programación nos permite realizar esta labor de forma sencilla. También nos ofrece una modificabilidad bastante amplia. A la hora de querer cambiar alguna funcionalidad, es tan fácil como ir al método en cuestión y realizar el cambio. Así se aplica a todos los objetos que hayamos desarrollado, evitando que tengamos que ir elemento a elemento realizando dicho cambio. Además, este tipo de programación nos ofrece fiabilidad ya que, al dividir los problemas en partes más pequeñas, es mucho más fácil evitar errores y fortalecer el código lo máximo posible. Finalmente, también nos aporta mantenibilidad. Debido a la sencillez para detectar el problema, los códigos basados en programación orientada a objetos son más sencillos de leer y comprender, pues nos permiten ocultar detalles de implementación dejando visibles solo los más relevantes.

Por todo lo expuesto anteriormente, se decidió implementar el código con este tipo de programación, haciendo el trabajo relativamente más sencillo.

## 2. Funcionamiento del simulador de la planta

Este simulador, como ya hemos explicado, va a reproducir el comportamiento del prototipo físico que se tenía pensado construir. Para ello, se ha realizado una simulación continua. Esto es debido a que el estado del sistema, en este caso el volumen de los depósitos cambia continuamente en el tiempo, pero dependiendo de unos eventos los cuales son la apertura y el cierre de las válvulas. Además, existen otro tipo de eventos que van a influir en la simulación, como puede ser que el depósito esté lleno o vacío, lo cual hace que el incremento o el decremento de volumen del depósito en ese instante sea 0.

Como ya se ha mencionado, el volumen de los depósitos va a ir incrementando o disminuyendo en cada ciclo de simulación en función de si las válvulas de llenado o vaciado están abiertas o cerradas. Si la válvula de llenado del depósito se encuentra abierta y la de vaciado cerrada, el depósito incrementa su volumen una determinada cantidad que depende del caudal de entrada que se ha preestablecido. A su vez, si la válvula de vaciado está abierta y la de llenado cerrada, el depósito disminuye su volumen una cantidad que depende también de otro caudal preestablecido. Cabe destacar que estos caudales se han supuesto constantes mientras el tanque no esté vacío. Otros eventos que se están teniendo en cuenta son cuando el tanque está vacío o lleno. Si está lleno, la cantidad incrementada será 0 ya que ha llegado a su límite, y si está vacío, la cantidad que decrementa es 0 debido a que no tiene agua que perder. Estas condiciones también se aplican a tanques interconectados. Si el tanque que se encuentra por encima está vacío, el tanque de debajo no se va a llenar.

En este caso, el paso de simulación establecido es variable, ya que depende de lo que tarde el Arduino en realizar un ciclo de código. Para hallar este valor, se ha usado la función `millis()`, que nos permite saber cuánto tiempo ha pasado desde el comienzo de la simulación. Este mismo paso de simulación es el que se ha usado para generar los incrementos y decrementos de volumen que se producen en los depósitos ya que, para hallar esa cantidad, hay que multiplicar el caudal de entrada o salida preestablecido por el paso de simulación. De esta manera, nos aseguramos de que estas cantidades sean lo más fiables posibles.

Cabe destacar que el valor de los sensores se actualiza con cada ciclo de simulación haciendo que el envío de datos esté acorde con la actualización que se produce en el simulador.

### **3. Electrónica para el acondicionamiento de señales**

Como ya hemos mencionado, el simulador manda a la controladora unas señales que representan el volumen de agua en los depósitos. Debido a la tecnología de los Arduinos seleccionados, no es posible mandar señales analógicas como salida. En su lugar, algunos de los pines digitales tienen la capacidad de enviar señales PWM (Pulse Width Modulation - Modulación por Ancho de Pulso) cuyos voltajes eficaces coinciden con la tensión analógica que queremos enviar. Estas señales tienen una frecuencia de 500 Hz y la conversión se realiza entre 255 niveles de tensión. Es decir, mandar un 0 corresponde a 0 voltios y mandar un 255 equivale a una tensión de 5 voltios. Los valores intermedios definen el ciclo eficaz de la PWM.

Esto en sí no es ningún problema, ya que la señal envía la información que nos interesa. Sin embargo, la controladora no es capaz de leer el voltaje eficaz de la señal, sino que la interpreta como intervalos de 0 y 5 voltios sin tener en cuenta el voltaje eficaz de la señal PWM.

Para solucionar este problema, se buscó la manera de convertir esa PWM en un voltaje que nuestra controladora fuera capaz de manejar. Para ello, se decidió implementar un filtro pasa-baja de primer orden que convirtiera la señal en un voltaje analógico, pese a que presentara luego un pequeño rizado. Este filtro se usa en todas las señales PWM que envía el simulador, y la salida es la tensión que recibe la controladora como señal que represente el volumen del depósito.

En un principio, se planteó el no implementar el filtro pasa-baja para evitar tener que introducir componentes analógicos en el proceso. Se pensó en implementar un filtro de la media en el código de la controladora para que la PWM realmente coincidiera con el valor analógico deseado. Esta opción se descartó porque el filtrado vía software requiere de

mucho tiempo de simulación, lo cual ralentizaría la simulación interfiriendo en los otros procesos. También está relacionado con el tiempo de muestreo de las entradas, que podía resultar insuficiente.

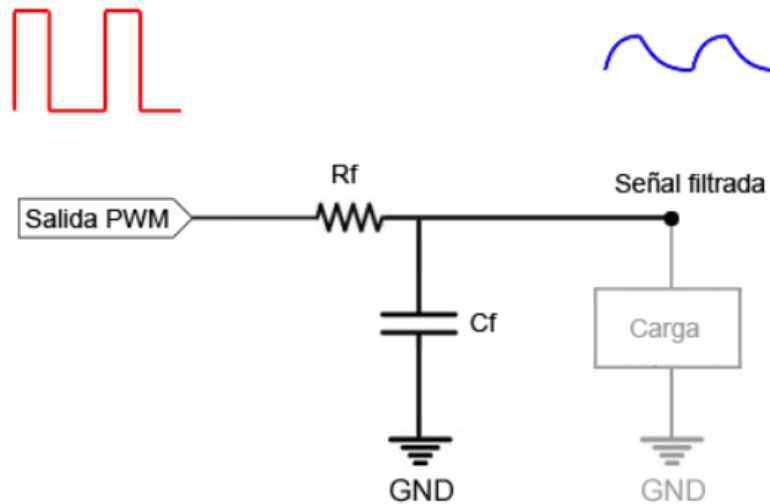


Figura 4.1: Esquema del filtro implementado

En la Figura 4.1 podemos observar la morfología del filtro que se ha implementado para la conversión de las señales PWM, así como el aspecto que tiene la señal al salir del mismo. Lo primero que debemos tener en cuenta es la frecuencia de nuestra señal PWM, que es de 500 Hz debido a las características del Arduino. Sabemos que la frecuencia de corte del filtro tiene que ser mucho menos que la de la señal. Por lo tanto, tenemos que seleccionar unos componentes que cumplan esta premisa, vigilando que el tiempo de respuesta no sea muy elevado, ya que podría entorpecer el funcionamiento de la controladora. Sabemos que el valor de la frecuencia de corte viene dado por la siguiente expresión:

$$f_c = \frac{1}{2\pi * C * R} \quad (4.1)$$

Hemos seleccionado un condensador de 1 microfaradio y una resistencia de 10k ohmios. Con estos componentes, obtenemos una frecuencia de corte de 15.91 Hz, cumpliendo la condición mencionada anteriormente, ya que la frecuencia de corte está muy por debajo de los 500Hz.

Sabemos que con estos valores se va a obtener una respuesta considerablemente rápida con un rizado mínimo, ya que por definición:

$$\tau = R * C \quad (4.2)$$

Dicho esto, obtenemos una constante de tiempo ( $\tau$ ) de 0.01 segundos, valor que nos permite alcanzar el valor de consigna, es decir, el valor deseado para el voltaje en el menor tiempo posible. En la Figura 4.2 podemos observar la comparación entre la señal PWM con un ciclo de trabajo del 50% (equivalente en nuestro caso a 2.5 voltios) con la señal de salida del filtro, observando que alcanza su valor de consigna en 0.03 segundos. La señal azul Ref(t) corresponde con la PWM y la roja U(t) con la señal filtrada.

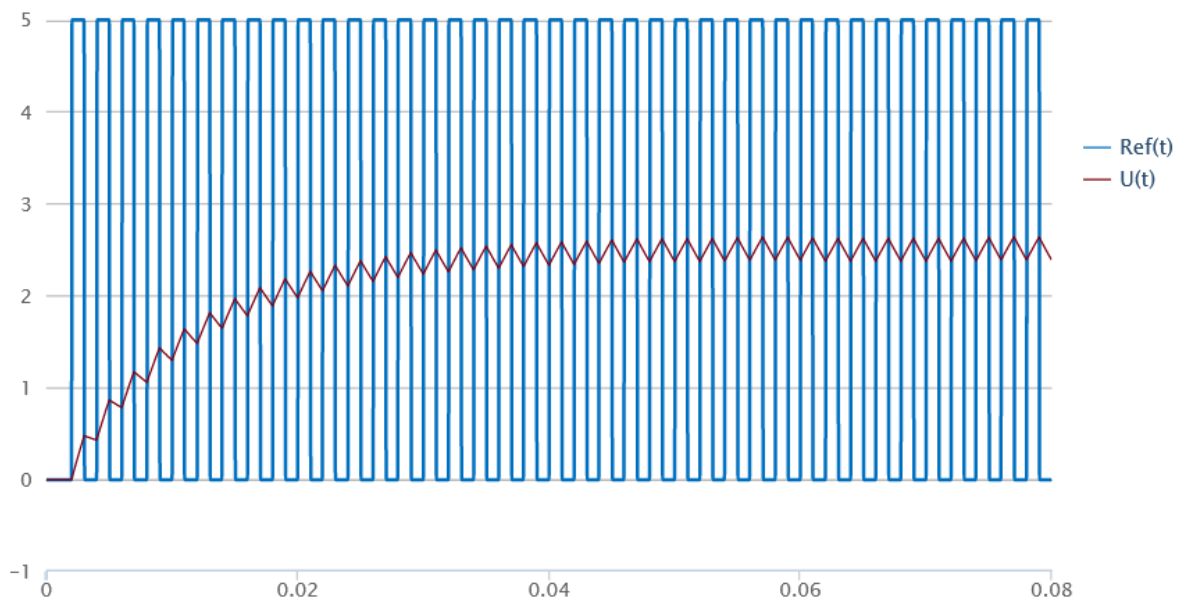


Figura 4.2: Comparación de la entrada y salida del filtro para una señal PWM equivalente a 2.5 voltios

Como podemos observar en la gráfica, gracias al filtro conseguimos generar un voltaje bastante similar a la señal que queremos obtener. No obstante, hay que tratar de disminuir el rizado de la señal de voltaje tanto como sea posible con el fin de que la controladora no genere situaciones que no se están dando. Este rizado, pese a haberlo minimizado, puede hacer que la controladora entre en un estado de alarma por desbordamiento si nos encontramos cerca del límite de volumen establecido. La controladora puede hacer saltar una alarma si el depósito en cuestión está a punto de superar su volumen máximo para evitar que el agua se rebose y caiga sobre las básculas. Debido a esto, se implementó por código un límite de seguridad para el nivel del agua en los depósitos, evitando que se produzca lo mencionado anteriormente. De ahí la importancia de eliminar este rizado. En el apartado 3 del capítulo V se explica cómo se ha conseguido.





## CAPÍTULO V: CONTROLADORA

### 1. Funcionamiento e idea general

La controladora es otro elemento clave del proyecto. Es la encargada de gestionar el funcionamiento de toda la planta, ya sea la apertura o el cierre de las válvulas o la generación del caudal de fugas. Además, también genera los valores de la distribución aleatoria que determina la cantidad de agua que se descargará en el siguiente movimiento de cada tanque o el tiempo en el que se producirá esa descarga. Por otra parte, también se encarga de gestionar el panel de control de la planta, del que se hablará más adelante.

En un supuesto caso donde tuviéramos el prototipo físico de la planta, esta controladora estaría conectada a ella en todo momento gestionando su funcionamiento. En el caso que nos ocupa, en el que se ha desarrollado un modelo simulado de la planta física, la controladora se conecta directamente con el simulador. Como se comentó en el apartado 1 del capítulo IV, también se ha usado la programación orientada a objetos para desarrollar el código de la controladora.

La controladora a su vez va a generar una nueva simulación, que es la encargada de crear los eventos que definen sus propias acciones, es decir, las ventas y las recargas del depósito de almacenamiento. Para ello, se ha realizado una simulación de eventos discretos.

Estos eventos van a ir definiendo las cantidades de agua que se van a descargar y los instantes de tiempo en los que se van a producir estas descargas. Esto se consigue generando, mediante una distribución aleatoria, los valores que corresponden con los parámetros ya mencionados. Los volúmenes de descarga son usados por la controladora como condiciones de evolución en la máquina de estados y, en el caso de los tiempos, se comprueba cuánto ha pasado desde el inicio de la simulación y se compara con el valor que se ha generado en la simulación. De este modo, también se pueden producir cambios de estado. Cabe mencionar que el valor del tiempo en el cual se va a producir la siguiente descarga no se sobrescribe, sino que se irá sumando al valor anterior. De este modo,

podemos generar un efecto cola que simula varias ventas sucesivas que están a la espera de que acabe la que se está produciendo en ese momento. En la Figura 5.1 podemos observar cuál sería el comportamiento ideal de los depósitos en función de las cantidades de agua que gestionan:

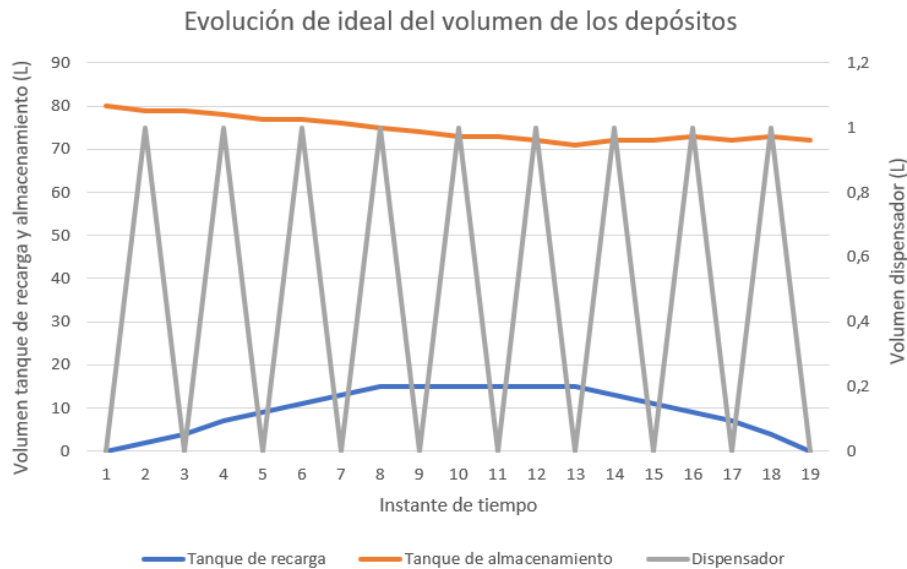


Figura 5.1: Evolución ideal del volumen de los depósitos.

Para poder explicar mejor esta generación de eventos, se han dividido los tanques en dos tipos, los tanques activos y los pasivos. Los tanques activos son aquellos que generan eventos de descarga y actúan sobre las válvulas. Los pasivos van a recibir las descargas de los depósitos anteriores y no participan activamente en la simulación. Como tanques activos tenemos el tanque de recarga, los dispensadores y el tanque de fugas. Como pasivos tenemos el tanque de almacenamiento y el de retorno. En estos depósitos pasivos no se tienen en cuenta los valores de la distribución aleatoria, haciendo que se mantenga en un estado fijo durante la simulación. Otra razón para esto es la coincidencia de válvulas. La válvula de vaciado del tanque de recarga corresponde con la de llenado del tanque de almacenamiento, y si en algún punto de la simulación llegan a estar en el mismo estado, se podría producir una irregularidad en el sistema ya que estarían dando órdenes distintas a un mismo actuador.

A su vez, estos sucesos generados en la controladora son los detonantes que generan los eventos del simulador de la planta, que son la apertura y cierre de las válvulas.

## 2. Estados que contempla la controladora

Para este proyecto se ha decidido crear una máquina de estados tipo Moore, en la cual el propio estado define las acciones que se van a llevar a cabo. Esta máquina tiene anidadas a su vez una máquina de estado por cada depósito activo, las cuales también son máquinas tipo Moore. En función del estado en el que se encuentra la máquina principal, las demás actúan en consecuencia y se comportan de una manera o de otra. Empezaremos explicando la máquina de estados de la controladora y, una vez entendido su funcionamiento, veremos cómo actúa la máquina de estados de los depósitos.

La máquina de estados de la controladora está formada por 4 estados que representan el funcionamiento de la planta. Todos ellos están basados en los estados que puede tener una planta industrial, con el fin de ser lo más fiel posible a este ámbito. Los estados contemplados, así como las posibles evoluciones entre ellos y las condiciones que los provocan se muestran en la figura 5.2:

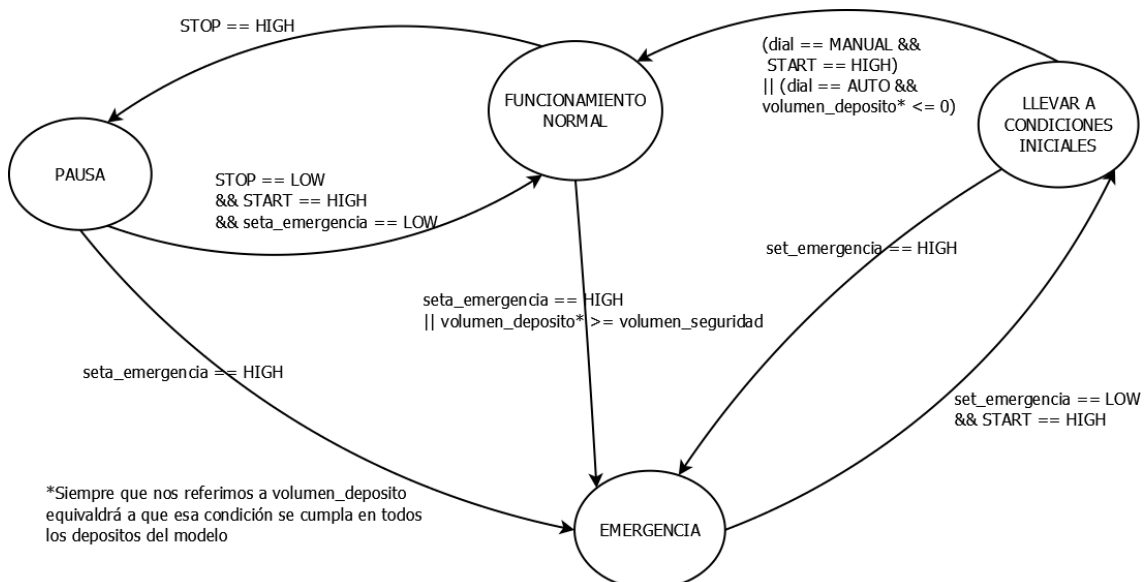


Figura 5.2: Máquina de estados de la controladora.

A continuación, se va a explicar en qué consisten los estados de la controladora:

- **FUNCIONAMIENTO NORMAL:** estado en el cual todos los depósitos actualizan su estado para ver en qué parte del proceso se encuentran. A partir de este estado se puede entrar en el estado de PAUSA si se acciona el pulsador de STOP.
- **PAUSA:** estado en el cual todas las válvulas se cierran y los depósitos se mantienen esperando a que se vuelva a FUNCIONAMIENTO NORMAL accionando el pulsador de START.
- **EMERGENCIA:** cuando se activa la seta de emergencia o un depósito sobrepasa el volumen de seguridad, el sistema entra en este estado forzando en todos los depósitos el estado de emergencia. Se mantienen en ese estado hasta que el operario decida volver a condiciones iniciales, desactivando la seta de emergencia y pulsando START. Hay que añadir que se puede transitar a este estado desde cualquier otro existente.
- **LLEVAR A CONDICIONES INICIALES:** una vez el operario saca el sistema del estado de emergencia, debe reiniciarlo haciendo que se vacíen todos los depósitos para empezar la simulación desde cero. Cuando todos estén vacíos, o el operario dé la orden, se volverá a FUNCIONAMIENTO NORMAL. Este proceso sigue en bucle hasta que el usuario decida que debe terminar o, en su defecto, cuando las variables de simulación lleguen a su límite en un periodo de 50 días debido a que la variable de simulación se desbordará pasado este periodo (uso de la función millis() de Arduino). Podemos observar el diagrama de estados en la Figura 5.1.

El panel de control que se ha diseñado para la controladora dispone de un dial, tres pulsadores y un indicador LED que nos avisa si el sistema ha entrado en estado de alarma. A continuación, se explica la función que realiza cada uno de los elementos del panel:

- **Dial manual/automático:** este dial es el que define el modo de funcionamiento de la planta. Normalmente está en la posición de automático, pero en determinados momentos nos interesa poder pasar al modo manual, por ejemplo, cuando entramos en la etapa de reinicio y no queremos que los depósitos se vacíen completamente. En función de la posición del dial, entramos en un modo u otro.

- **Pulsador de START:** Como su propio nombre indica, este pulsador nos permite reanudar la simulación si se ha producido una pausa o, en caso de que estemos en modo manual, reanudar la simulación después de un reinicio.
- **Pulsador de STOP:** Como su propio nombre indica, es el pulsador encargado de parar el sistema si el operario lo precisa. Para reanudar el funcionamiento hay que accionar el pulsador de START.
- **Seta de emergencia:** Es el elemento encargado de poner el sistema en un estado de emergencia forzado por el operario. Una vez activada la seta, se enciende un LED que nos indica que estamos en estado de emergencia. Para salir de este estado hay que devolver la seta a su posición inicial y accionar START.

En la Figura 5.3 podemos observar un esquema de cómo es el panel de control descrito anteriormente:

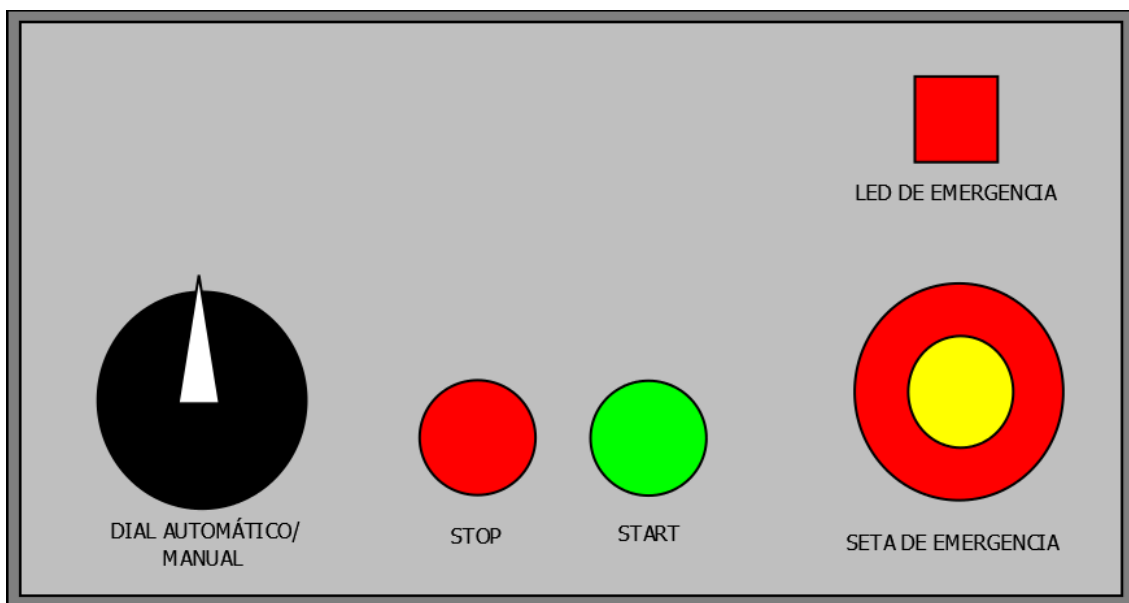


Figura 5.3: Panel de control de la controladora.

Una vez explicada la controladora globalmente, empezaremos ahora con la explicación de la máquina de estados de los depósitos, la cual se puede ver en la Figura 5.4:

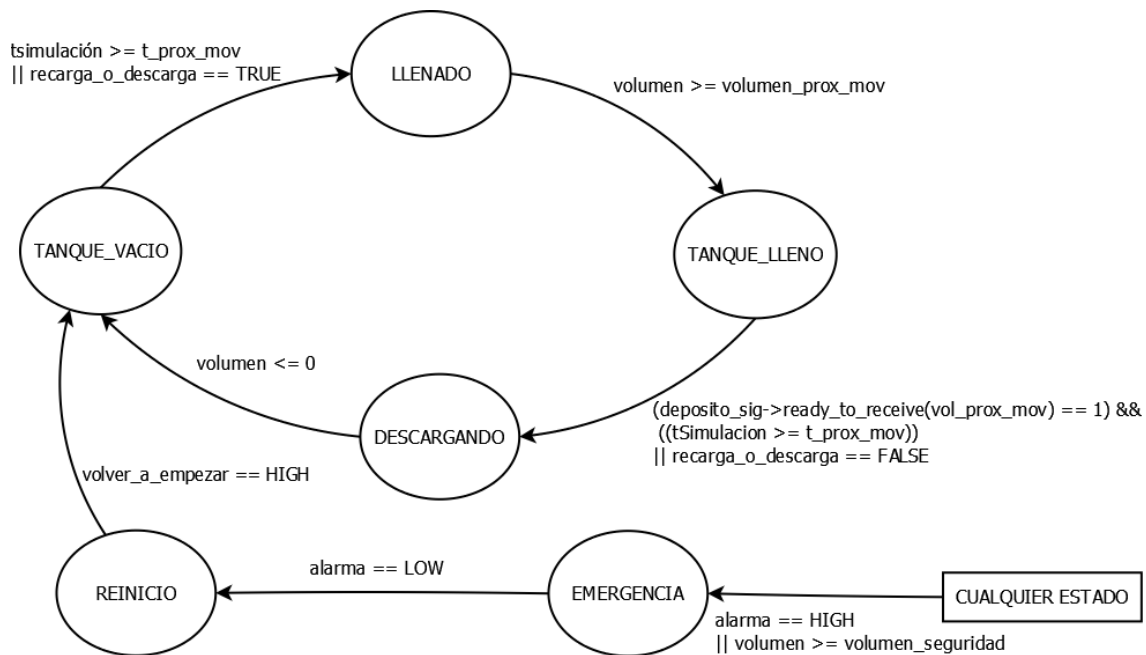


Figura 5.4: Máquina de estados de los depósitos.

Podemos observar que estos estados hacen referencia, en su mayoría, a la cantidad de agua que contienen los depósitos, sin tener en cuenta el estado de emergencia y el de reinicio:

- **TANQUE\_VACIO:** Estado inicial de todos los depósitos activos. Mantiene las válvulas de llenado y vaciado cerradas hasta que sea el momento de empezar el llenado.
- **LLENADO:** En este estado se abre la válvula de llenado y se mantiene cerrada la de vaciado hasta que el depósito se llene con la cantidad deseada de agua.
- **TANQUE\_LLENO:** El depósito mantiene ambas válvulas cerradas hasta que la simulación diga que es el momento de descargar.
- **VACIADO:** En este estado se abre la válvula de vaciado y se mantiene cerrada la de llenado hasta que el depósito quede vacío y vuelva al estado inicial.
- **EMERGENCIA:** Los depósitos entran en este estado de emergencia si se ha activado la seta o si han sobrepasado su volumen de seguridad. Mantiene todas las válvulas cerradas.

- **REINICIO:** Una vez se ha salido del estado de emergencia, se abren todas las válvulas del sistema para vaciar todos los tanques. Una vez finalizado el vaciado, se puede volver a empezar la simulación.

Como ya hemos remarcado, cada depósito tiene su propia máquina de estado, cada uno con sus propios atributos. Funcionan independientemente, pero si es necesario pueden coger datos de los otros, como el volumen de los depósitos que se encuentran debajo de ellos. La funcionalidad de esto reside en que así los depósitos pueden saber si el tanque siguiente está preparado para recibir una descarga o si esta produciría un desbordamiento.

### 3. Acondicionamiento de las señales de entrada

En el Anexo 2 se habla de la conversión que realiza el Arduino para convertir una señal analógica en valores que pueda entender el microcontrolador. Antes de realizar la conversión que se menciona en ese apartado, debemos realizar un paso previo. En el capítulo IV, apartado 3 se explicó que se ha tenido que implementar un filtro de primer orden con el fin de convertir una señal PWM en una tensión analógica reconocible por el microcontrolador. Pero esa tensión de salida del filtro presenta un rizado que puede alterar el funcionamiento de la controladora. Por eso, se ha implementado un método para poder eliminar ese ruido presente en la señal.

El rizado que presenta la señal filtrada sigue una función parabólica que depende de la tensión que queremos enviar. Esto es debido a que las variaciones de tensión son mayores cuando una PWM tiene un ciclo de trabajo del 50%, es decir, en la mitad de su periodo se encuentra en el nivel bajo y en la otra mitad en el nivel alto. A su vez, cuando estos ciclos de trabajo son del 0% o del 100%, no presenta ningún rizado, ya que corresponden con tensiones de continua. Como podemos observar en la Figura 5.5, el rizado se hace 0 cuando enviamos una señal de 0 o de 5 voltios, llegando a su máximo valor cuando enviamos una señal de 2.5 voltios (ciclo de trabajo del 50%).



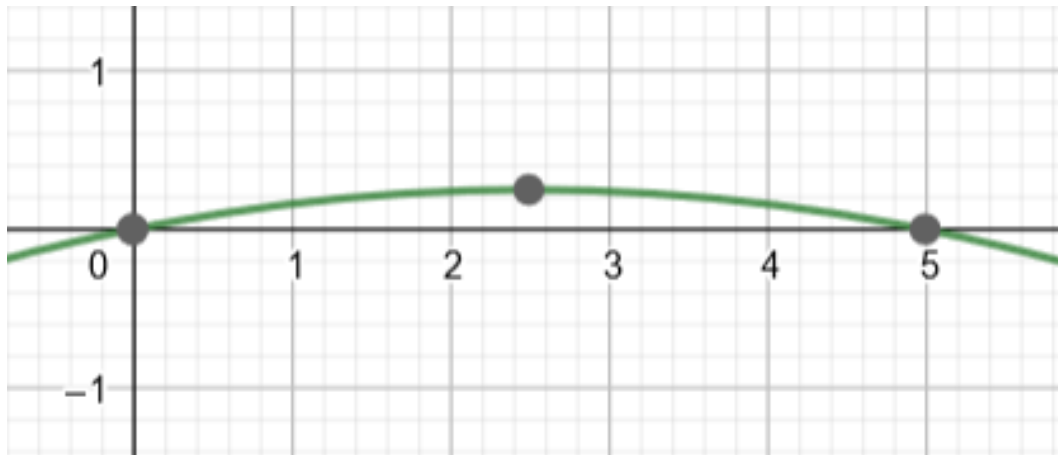


Figura 5.5: Función del rizado en la salida del filtro.

Esta gráfica ha sido obtenida mediante la calculadora online Geogebra, la cual nos permitió generar y obtener la función que sigue el rizado de la señal en función de su ciclo de trabajo.

A la vista del comportamiento de la señal, para eliminar el rizado que presenta se coge la señal analógica que entra en el Arduino y se le resta la función de rizado para así eliminar los valores que se encuentran por encima de su valor real. Esta función del rizado es la siguiente:

$$Vr = 0.25 - \left(\frac{2.5 - Ventrada}{5}\right)^2 \quad (5.1)$$

Como ya se dijo anteriormente, el rizado máximo que se va a presentar es de 0.25 voltios y aparece con una tensión de 2.5 voltios. Hay que recordar que “Ventrada” es la tensión que nos da el filtro y “Vr” la tensión de rizado que presenta. Teniendo en cuenta estos límites, a partir de la ecuación 5.1 se obtienen todos los valores de rizado que pueden darse en el intervalo 0V-5V que nos da la señal del Arduino. Una vez restado este valor a la tensión que recibimos, ya podemos hacer la conversión a volumen de la señal recibida, la cual se describe en el Anexo 2.



# CAPÍTULO VI: SIMULACIONES Y CONCLUSIONES OBTENIDAS

## 1. Simulador utilizado y situaciones simuladas

Para poder validar nuestro modelo, se han realizado una serie de simulaciones para comprobar si los códigos implementados llevan a cabo su función de manera satisfactoria.

Dada la situación actual, no fue posible llegar a implementar físicamente ambos microcontroladores, ya que debido al confinamiento no se disponía del material necesario para crear el modelo. En su lugar, se decidió hacer una simulación vía software usando alguno de los simuladores más usados en la red. Por otra parte, hay que añadir que el hecho de simular vía software es un proceso que elimina cualquier error que pudiera producirse debido al hardware, ya sea que alguno de los condensadores no funcionara bien o que alguno de los cables estuviera en mal estado. Así conseguimos aislar los problemas derivados del hardware.

Se barajaron varias opciones en cuanto a qué simulador utilizar. Inicialmente se pensó en utilizar el programa *Fritzing*, debido a que es una herramienta bastante utilizada, con una interfaz intuitiva y que, según las críticas, funcionaba de manera satisfactoria. Desgraciadamente, al trabajarlo en profundidad se vio que simplemente era un programa de modelado de circuitos y no permitía simular los archivos. La segunda opción que se contempló fue el simulador *Virtual Breadboard*, esta vez asegurando que sí se trataba de un simulador. Tenía buenas críticas por Internet y su interfaz era bastante cómoda. Pero para poder simular microcontroladores Arduino, había que obtener la versión premium y, por tanto, de pago. Debido a esto se descartó esta opción. Finalmente se decidió usar el simulador online *Tinkercad* de *Autodesk*, siendo esta la opción donde se han llevado a cabo todas las simulaciones. Pese a tener una interfaz más simple y menos opciones que el anterior simulador, cumplía los mínimos que se necesitaban para poder trabajar y simular lo que este proyecto requiere.

Para poder validar el proyecto, se realizaron varias simulaciones con el fin de depurar el código todo lo posible y ver contempladas todas las situaciones que se pueden dar en la práctica. A continuación, se van a describir todas estas situaciones:

- **Simulación de un solo depósito:** con el fin de comprobar si la clase depósito estaba bien definida, se creó un archivo de simulación donde se implementaba un solo depósito para comprobar si la máquina de estados funcionaba correctamente. Como podemos ver en la Figura 6.1, para simular el sensor de volumen del tanque se usó un potenciómetro alimentado con 5 voltios para que, variando la posición del dial, nos da una tensión que simula el comportamiento de un sensor. También podemos observar que se añadieron unos LEDs para comprobar que se activaba la salida correspondiente. Cabe mencionar que el depósito implementado se comporta como un dispensador, con el fin de que realizaran acciones con mayor periodicidad debido a que presentan una capacidad menor a la de otros depósitos y se apreciara más rápidamente el funcionamiento del código.

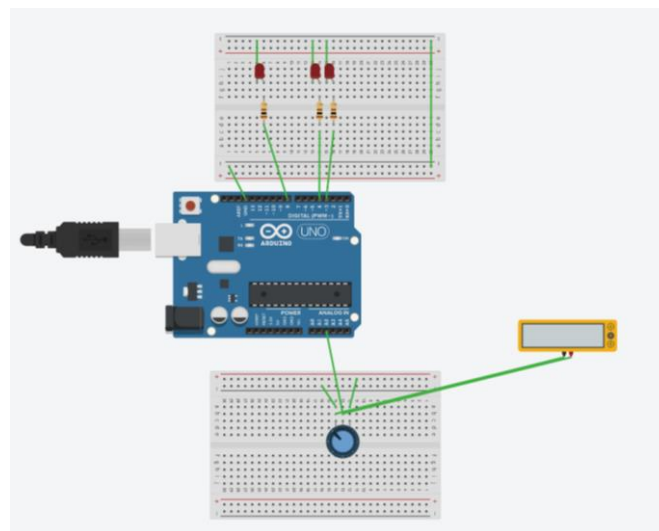


Figura 6.1: Simulación de un depósito.

- **Simulación de la planta sin el depósito de fugas:** una vez comprobado el funcionamiento de los depósitos, se decidió validar el simulador de la planta con el fin de observar si realizaba satisfactoriamente su función. En esta simulación no se implementó el depósito de fugas porque tiene un comportamiento diferente al resto y se decidió

comprobar su funcionalidad por separado. Se puede observar el diseño de la simulación en la Figura 6.2.

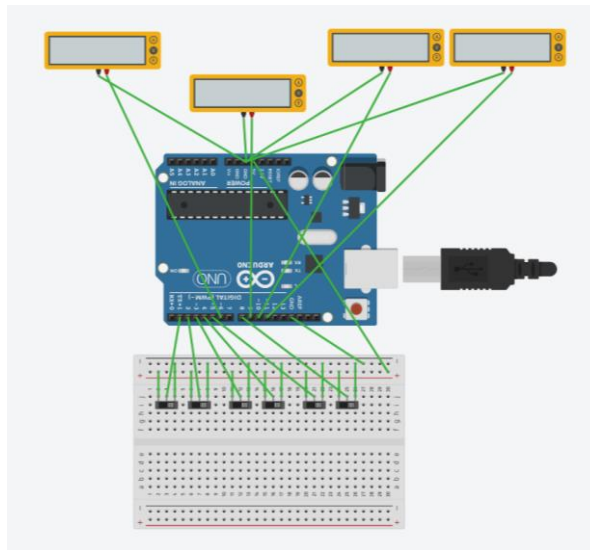


Figura 6.2: Simulación de la planta sin el depósito de fugas.

En este caso, se implementaron unos interruptores todo-nada con los que se simula la apertura y cierre de las válvulas, permitiendo o impidiendo el paso del agua entre los depósitos. Los multímetros sirven para poder observar la tensión de salida que nos da el simulador, lo cual equivale al volumen de agua en los depósitos. En función de la combinación que se realice con los interruptores, los depósitos se comportan de una manera u otra. Por ejemplo, aunque se abra la válvula de llenado de un depósito, si el que está por encima está vacío, este no se va a llenar.

- **Simulación de la controladora sin el depósito de fugas:** tomando como punto de partida la primera simulación de un único depósito, el siguiente paso fue simular la controladora en su totalidad, sin añadir el tanque de fugas por las mismas razones que en la simulación anterior. Siguiendo la línea de trabajo usada en las anteriores simulaciones, se han usado LEDs para comprobar la apertura y el cierre de las válvulas, potenciómetros para simular las señales de los sensores de volumen y multímetros para monitorear este valor. Podemos observar en la Figura 6.3 que, en este caso, hay unos interruptores que emulan el panel de control que está implementado en la controladora y, de esta manera, poder ver si la máquina de estados funciona correctamente.

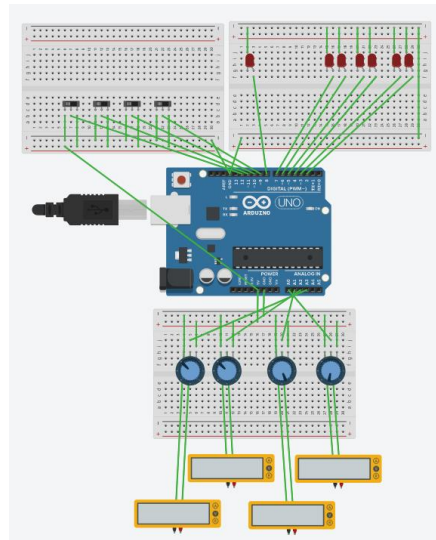


Figura 6.3: Simulación de la controladora sin el depósito de fugas.

- **Simulación completa de la planta:** una vez comprobado que tanto el código de la controladora como el del simulador de la planta funcionan bien por separado, el siguiente paso consistió en incluir el tanque de fugas. El esquema es prácticamente el mismo, aunque con pequeñas diferencias como se puede observar en la Figura 6.4. Se añadió un potenciómetro para simular el caudal de fugas, ya que es la única entrada analógica que recibe el simulador. En función de este caudal, el depósito se llenará a mayor o menor velocidad. Debemos recordar que el llenado del depósito de fugas es un proceso bastante lento ya que hablamos de caudales muy pequeños en comparación con la capacidad del tanque de almacenamiento.

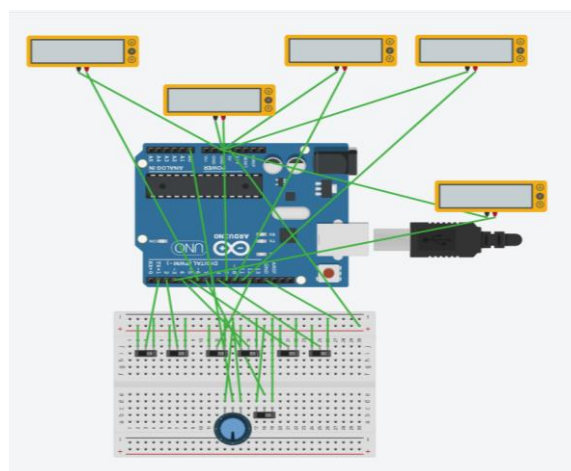


Figura 6.4: Simulación completa de la planta.

- **Simulación completa de la controladora:** el siguiente paso es la simulación de la controladora añadiendo esta vez el depósito de las fugas. Los pequeños cambios que se han hecho con respecto a la simulación sin el tanque de fugas son la adición de un multímetro que nos dice el caudal que genera la controladora, el cual va dirigido al simulador de la planta para generar estas fugas. También se añadió la válvula de vaciado de dicho tanque y el potenciómetro que simula la señal del sensor de volumen del mismo. Por lo demás la simulación es la misma que en el caso sin fugas. Se puede ver el esquema en la Figura 6.5.

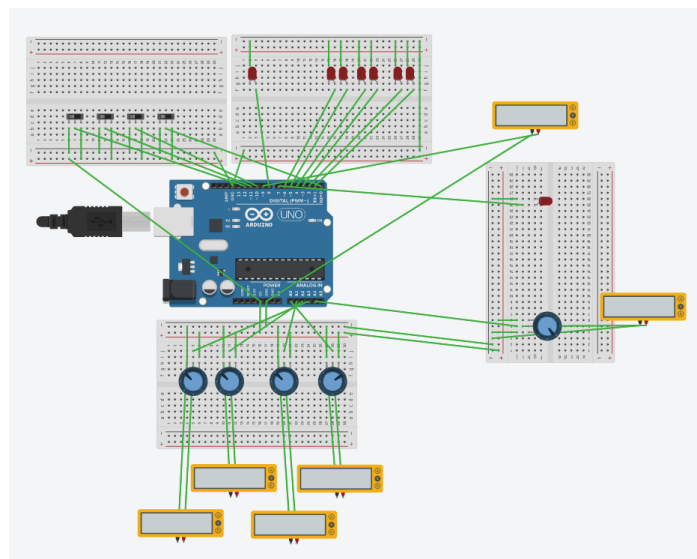


Figura 6.5: Simulación completa de la controladora.

- **Simulación conjunta sin fugas:** ya comprobado el funcionamiento del código de la controladora y el del simulador de la planta, la siguiente simulación consistió en unir ambos microcontroladores y añadir la electrónica ya comentada en el apartado 3 del Capítulo IV para observar si el sistema funciona como debe. En la figura 6.6 se pueden apreciar los filtros implementados para las señales analógicas y los multímetros que nos dan esos valores de tensión antes y después de su paso por el filtro.



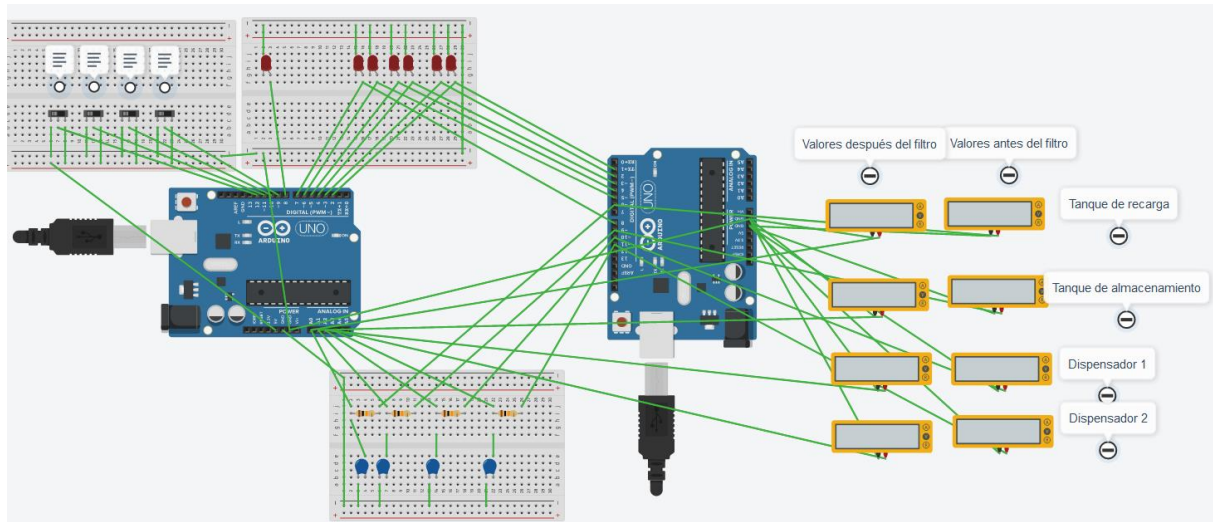


Figura 6.6: Simulación conjunta sin fugas.

• **Simulación completa:** después de haber fragmentado la simulación para asegurarnos de que todo funciona correctamente, el paso final es probar todos los códigos en su totalidad, sin eliminar ninguna de las partes. El aspecto que tendría el sistema si lo llegásemos a implementar físicamente sería el de la Figura 6.7. A partir de esta última simulación se han obtenido los datos que se presentan en el siguiente apartado. En el repositorio del código se ha añadido un vídeo comentado con esta última simulación, [11].

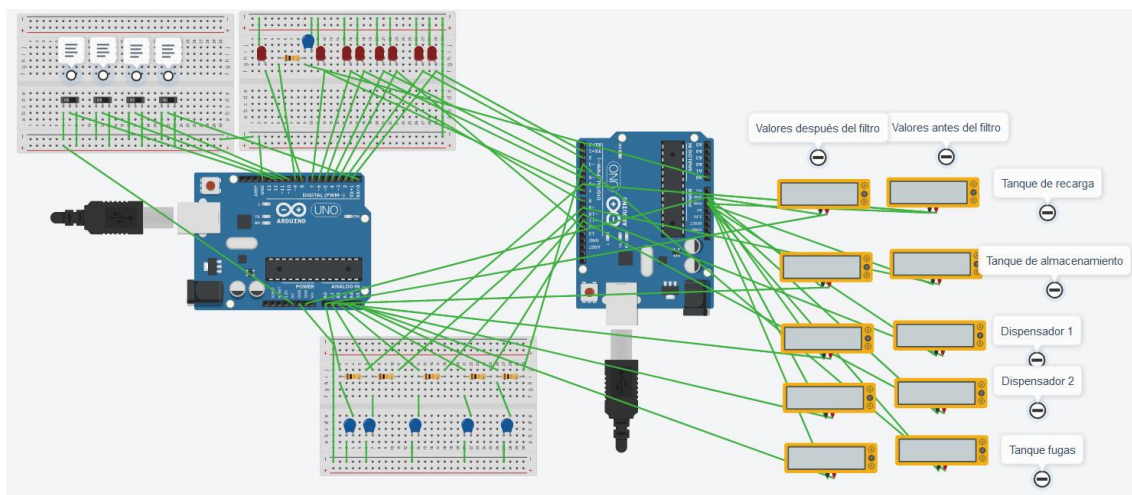


Figura 6.7: Simulación completa.



## 2. Validación del simulador de la planta y la controladora

A partir de las simulaciones nombradas anteriormente, se pudo comprobar que el código desarrollado para la controladora y el simulador de la planta cumplen con su función. Pero pese a haber completado ya esta tarea, este simulador no permitía extraer los datos de la simulación debido a la falta de implementación de algunas librerías. Así que, pese a que estas simulaciones son las que han permitido validar nuestro sistema, se consideró oportuno implementar los microcontroladores físicamente para poder obtener los datos mediante el monitor serial del entorno de Arduino. Esto hizo posible validar el modelo también con una implementación física que en primera instancia no se había podía realizar debido a la falta de material.

Los datos obtenidos que validan nuestro sistema corresponden con el valor de volumen que presentan los depósitos en un determinado instante del tiempo de simulación. Se ha analizado el comportamiento del tanque de recarga, del tanque de almacenamiento, de un dispensador y del tanque de fugas, para así poder comprobar los datos de todos los depósitos que participan activamente en la simulación. Por esto mismo, no se ha considerado analizar el tanque de retorno. Se puede observar en las gráficas presentadas en las Figuras 6.8, 6.9, 6.10 y 6.11 la evolución del volumen del tanque correspondiente en un tiempo de simulación de 10 minutos, excepto para el tanque de fugas debido a que su dinámica es mucho más lenta que la de los demás depósitos y se le ha dado un tiempo de simulación de 1 hora. Todos estos datos se han presentado en una gráfica lineal donde se representa el volumen del depósito en un instante determinado del tiempo de simulación. Se da junto al gráfico una pequeña explicación de lo que representa para poder luego analizar los datos que muestran y sacar las conclusiones pertinentes.

La primera gráfica que se presenta en la Figura 6.8 corresponde con el tanque de recarga, aquel que periódicamente rellena el tanque de almacenamiento simulando un camión cisterna. En la gráfica se observa el volumen del depósito (litros) frente al tiempo de la simulación (minutos):

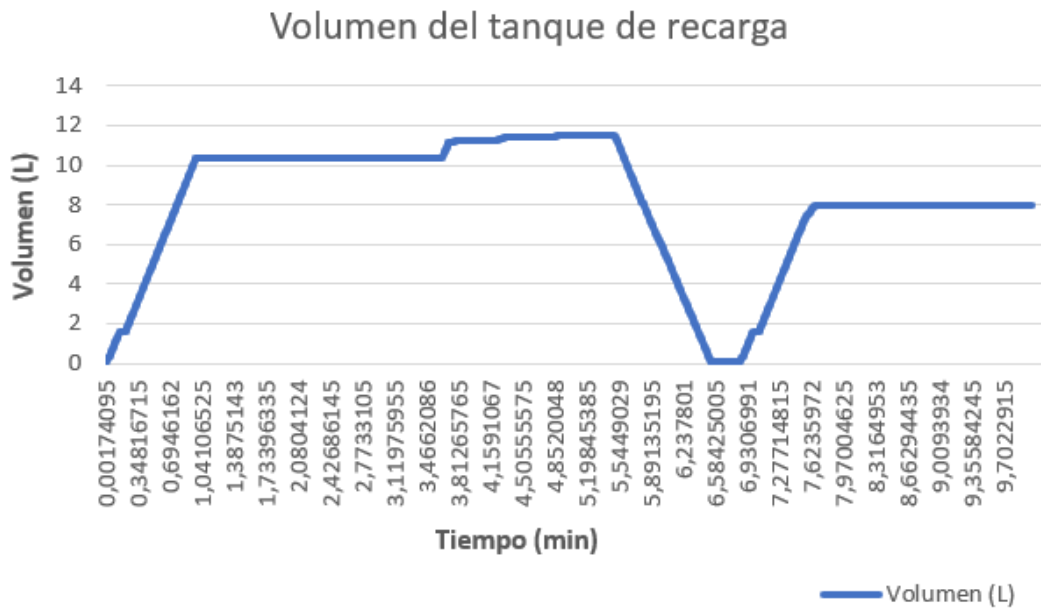


Figura 6.8: Datos referentes al volumen del tanque de recarga.

En la gráfica se puede observar cómo desde el inicio de la simulación su volumen se incrementa hasta un límite. Este valor límite es el que ha preestablecido la distribución aleatoria y mantiene ese volumen hasta que la controladora indique que es el momento de realizar la descarga en el tanque de almacenamiento. Se observa un pequeño salto sobre los 3,46 minutos de simulación provocado por fluctuaciones producidas por la electrónica usada al construir el modelo de simulación. Sobre los 5 minutos y medio de simulación se observa cómo ha empezado la descarga y el volumen del depósito empieza a disminuir. En el momento en el que la variable alcanza el cero, empieza a llenarse otra vez para así poder esperar a que la controladora le ordene la siguiente descarga. La simulación termina mientras el tanque espera a que se produzca la siguiente descarga. En 10 minutos solo se ha llegado a producir una descarga; esto es debido a los límites introducidos en la distribución, los cuales son los que determinan los tiempos en los que se producen la descarga. Estos valores están dentro de un rango entre 1 y 8 minutos, para que así el tanque de almacenamiento esté preparado para recibir la descarga después de haber perdido agua con las ventas.

La siguiente gráfica que se presenta en la Figura 6.9 corresponde con el tanque de almacenamiento. Podemos observar que este depósito parte de un volumen de 80 litros, ya que se supuso en la simulación que no estaba vacío para así poder trabajar con los dispensadores desde el principio de la simulación:

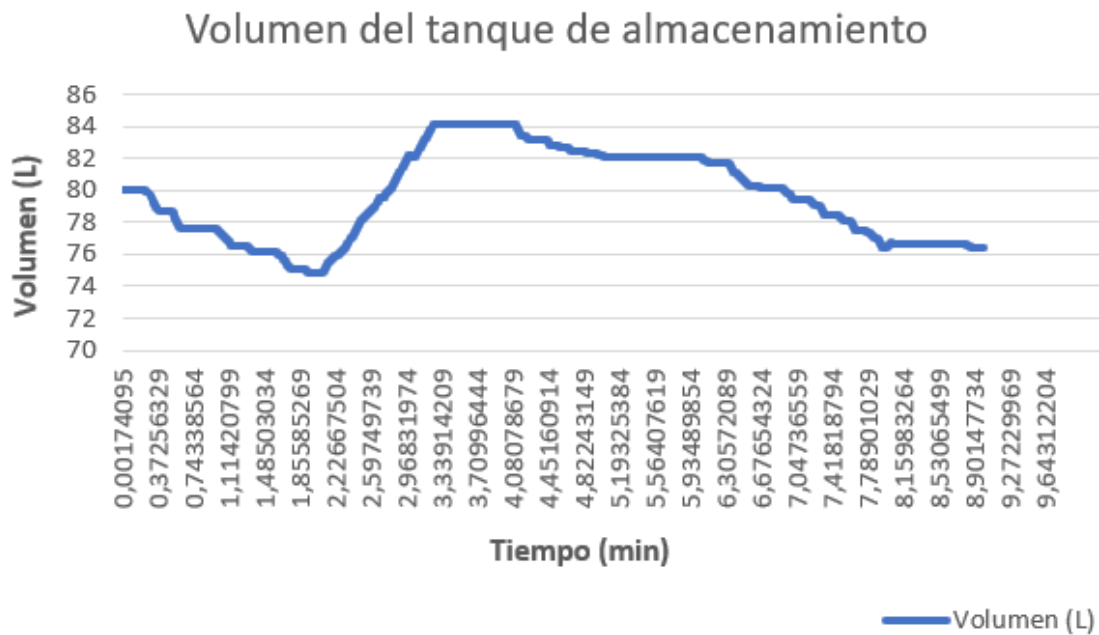


Figura 6.9: Datos referentes al volumen del tanque de almacenamiento.

En esta gráfica se puede observar cómo progresivamente va disminuyendo su volumen debido a las ventas que se están produciendo en los dispensadores. Cuando estos requieren agua, el tanque de almacenamiento va perdiendo una cantidad equivalente a la venta que se está produciendo. A partir de los dos minutos de simulación, se aprecia como su volumen aumenta debido a que el tanque de recarga está realizando una descarga. Pero este aumento de volumen es irregular debido a que mientras el tanque de recarga realiza su labor, los dispensadores siguen demandando agua al tanque de almacenamiento. Durante el resto de la simulación se observan más ventas que restan volumen al tanque.

A continuación, se presenta el gráfico de la Figura 6.10, correspondiente a uno de los dispensadores. Pese a que nuestro sistema tiene dos tanques de este tipo, se han tomado datos solo de uno, concretamente del dispensador 1, ya que su comportamiento es idéntico. Se puede ver que la actividad de este depósito es más frecuente que el de los anteriores:

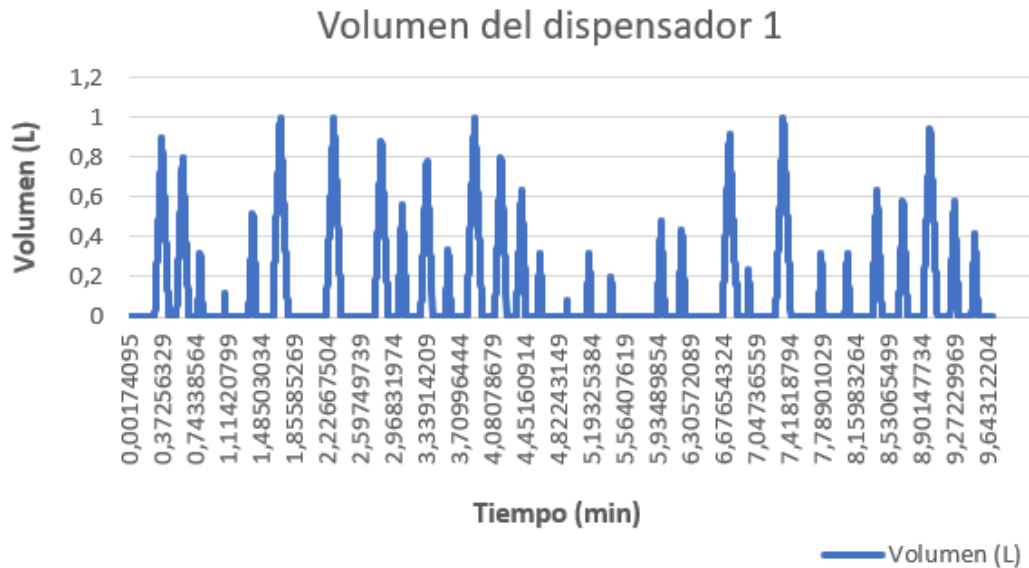


Figura 6.10: Datos referentes al volumen del dispensador 1.

En esta gráfica se puede ver perfectamente el comportamiento de este depósito: cuando la controladora da la orden, este se recarga con el agua del tanque de almacenamiento y cuando llega al valor deseado, se vacía para esperar a la siguiente venta; espera un pequeño periodo de tiempo y luego reinicia el proceso. Esta acción es mucho más frecuente que la de recargar el tanque de almacenamiento ya que, en una estación de servicio real, las descargas en el tanque principal no se realizan todos los días, mientras que la periodicidad de las ventas es mucho mayor en comparación. Para esto se ha definido un intervalo de actuación de entre 0 y 1 minuto y medio. De esta manera se puede apreciar la continuidad de las ventas.

Finalmente, se presenta en la Figura 6.11 el gráfico correspondiente al tanque de fugas, aquel que va a almacenar todas las fugas simuladas. Como ya se ha comentado, el tiempo de simulación para la obtención de datos ha sido de 1 hora, para poder así observar las pequeñas variaciones de volumen que se presentan debido a las fugas del tanque de almacenamiento. Hay que destacar que debido a cómo está configurado el monitor serial de Arduino, solo ha sido posible representar datos con dos decimales, razón por la cual la gráfica presenta la forma de escalón que se aprecia en la figura. Aun así, se puede observar cómo progresiva y lentamente va aumentando el volumen de este depósito:

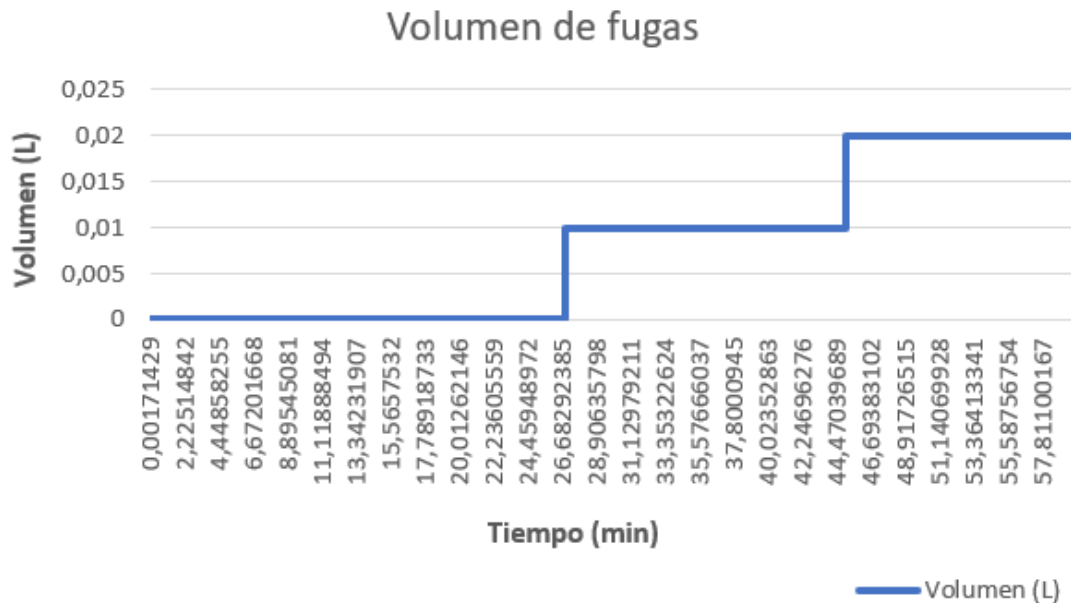


Figura 6.11: Datos referentes al volumen del tanque de fugas.

Debido a la lenta dinámica del depósito, en este caso la duración de la simulación es de una hora. Pese a ello, se puede observar que el aumento de volumen es mínimo, llegando solamente a 0,02 litros de agua. Esto está hecho intencionadamente, ya que esta simulación está pensada para aguantar funcionando varios días. De manera que, si dejáramos la simulación funcionando un día entero, el volumen de fugas que se genera es menor a 0,5 litros. En la gráfica podemos observar que antes de que empiece a aumentar el volumen, hay un periodo de 15 minutos donde no se generan fugas. Esto es debido a que la normativa nos exige que en la simulación exista un periodo inicial en el cual no se produzcan fugas, y así poder constatar que un futuro sistema de detección de fugas es capaz de discernir cuándo se está produciendo una fuga y cuándo no. Este periodo es de varios días, pero para poder apreciarlo, se ha disminuido el tiempo al valor mencionado. A partir de este intervalo de tiempo, ya empieza a aumentar el volumen del depósito. El sistema está configurado de manera que las fugas dependen del volumen del tanque de almacenamiento y son proporcionales a este valor. De esta manera podemos simular una fuga variable.

Una vez analizados todos los componentes de los cuales se han obtenido datos, podemos observar que todos los depósitos funcionan de la manera en que se programaron y cumplen su función predefinida en este sistema, conclusión a la cual también se llegó con el simulador antes de generar los datos con el modelo físico.

### 3. Conclusiones

Este proyecto se llevó a cabo con unas expectativas iniciales que, debido a la situación actual, no han llegado a hacerse realidad en su totalidad. Pese a ello, se ha realizado un trabajo con unos resultados bastante satisfactorios que demuestran que, a pesar de las dificultades, hemos sido capaces de adaptarnos a la situación que nos ha tocado vivir. Se ha conseguido crear un simulador de una pequeña estación de servicio que funciona correctamente, una controladora que gestiona el funcionamiento del elemento anterior y, además, se han sentado las bases para una futura ampliación del proyecto. Teniendo en cuenta los objetivos que se propusieron al comienzo del documento, podemos afirmar que se han cumplido la mayoría de ellos. Se ha conseguido realizar satisfactoriamente el diseño de la planta, como se proponía en el primer objetivo. Como ya se ha comentado a lo largo del documento, no se ha podido construir el prototipo debido al problema con los materiales, pero si se llegó a desarrollar la controladora y el simulador de la planta, validando su funcionamiento mediante una simulación software y una implementación física de los microcontroladores. Además, se ha conseguido que el presupuesto sea lo más bajo posible. Por lo tanto, la mayoría de los objetivos de este trabajo se han cumplido, a excepción de la construcción del prototipo.

Después de haber concluido el trabajo del proyecto, es destacable que uno de los problemas que se presentaron en el desarrollo fue a la hora de programar, ya que hubo una transición de mentalidad de la programación estructurada, filosofía de programación a la que estaba acostumbrado, a la orientada a objetos. No obstante, una vez entendido el método de trabajo, los avances fueron mucho más rápido. Así mismo, también se ha entendido la importancia de mantener una metodología de trabajo constante, concepto que no se contemplaba debido a que este ha sido el primer gran proyecto que he desarrollado. Pero esto es una lección importante para una futura entrada al mundo laboral, ya que mantener esa constancia me ayudará a evitar retrasos en entregas que hipotéticamente llegue a realizar.

Personalmente, la idea inicial de este proyecto me atraía bastante. El hecho de construir un modelo que funcionara de la manera que se ha demostrado en este documento me parecía desafiante y, al mismo tiempo, entretenido. En este sentido, la desafortunada situación que se está viviendo actualmente perjudicó mucho al proyecto. Pero el hecho

de haberse adaptado a estas adversidades de la manera en que se ha hecho y haber sacado adelante el trabajo, me parece más enriquecedor para mí como futuro ingeniero que el hecho de haber llegado a finalizarlo con la idea inicial. Esto me servirá como referencia en un futuro para tener una idea de cómo adaptarme a situaciones imprevistas que requieran dar una vuelta al modo de realizar una labor. Y eso es algo que me acompañará a lo largo de toda mi carrera.

### **Conclusions:**

This project was carried out with initial expectations which could not come true due to the current situation, Despite these facts, the work has been realized with quite satisfactory results that demonstrate a remarkable flexibility when dealing with adversities. It has been possible to create a simulator of a small service station that works correctly, a controller that manages the functioning of the previous element and, in addition, the foundations for a future expansion of the project have been laid. Seeing the objectives that were proposed at the beginning of the document, we can affirm that most of them have been achieved. The plant design has been successfully done, as proposed in the first objective. As already mentioned throughout the document, the prototype could not be built due to the problem with the materials, but the controller and the plant simulator were developed, validating their operation through a software simulation and a physical implementation of microcontrollers. Furthermore, the budget has been kept as low as possible. Therefore, most of the objectives of this work have been achieved, except for the construction of the prototype.

After completing the project, it is remarkable that one of the problems that appeared in the development was programming, because there was a mentality transition from structured to object-oriented programming. This change took a couple of hours of work due to the little habit to program with this discipline. But once the working method was understood, progress was much faster. Likewise, the importance of maintaining a constant work methodology has also been understood, a concept that was not contemplated because this has been the first great project that I have developed. But this is an important lesson for a future entry into the laboral world, since maintaining that constancy will help to avoid delays in future projects.

Personally, the initial idea of this project appealed to me a lot. Building a model that would work the way this document has shown seemed challenging and entertaining. Due to this, the unfortunate situation that is currently being experienced greatly damaged the project. But the fact of having adapted to these adversities in the way it has been done and have carried out the project, seems more enriching for me as a future engineer than finishing the project with the initial idea. This will serve as a reference in the future to get an idea of how to adapt to unforeseen situations that require a turn in the way of doing a job. This is something that will accompany me throughout my career.

#### 4. Líneas de trabajo futuras

Tomando como punto de partida este TFG, se plantean distintas líneas de trabajo futuras que contribuirían a ampliar y mejorar el funcionamiento del prototipo para la simulación de fugas de combustible en estaciones de servicio. A continuación, se presentan las más destacadas:

- **Construcción del modelo de la planta:** como ya hemos comentado en varias ocasiones, la idea inicial era construir una planta que simulase el funcionamiento de una estación de servicio que, debido a la situación actual, no fue posible llevarla a cabo. Por eso, se propone como línea futura más inmediata la construcción del prototipo físico. En este documento se dispone de la información presupuestal y técnica para la elaboración de esta, aunque cualquier mejora que se proponga será bienvenida. Se debe tener en cuenta que el modelo requeriría de un espacio de trabajo bastante amplio, y que sea capaz de albergar la estructura, la cual se estima que tendría unos 2 metros de altura y 1,5 metros de ancho. Se puede consultar el Anexo 1, que dispone del presupuesto que se estimó para los materiales necesarios.

- **Ampliación de la planta:** recordemos que el modelado de la planta diseñada es bastante básico: se ha usado agua en lugar de combustible, se han implementado solamente dos depósitos de ventas y uno de almacenamiento, y no se han tenido en cuenta parámetros como la temperatura o la presión del fluido. Proponemos entonces que se añadan ampliaciones al proyecto teniendo en cuenta cómo se podría acercar más a una estación de servicio real:



-Ampliación física de la planta: un punto bastante relevante sería la ampliación física del modelo, añadiendo más dispensadores y otros tanques de almacenamiento. Esto nos ayudaría a representar más fielmente el funcionamiento de las estaciones de servicio que operan hoy en día. Esto conllevaría un aumento del material necesario para la construcción y la adición al código de los depósitos correspondientes, pero debido al método que se ha usado para la definición de los depósitos, solo se debería crear un nuevo objeto y añadir sus acciones a la máquina de estados de la controladora.

-Adición de varios fluidos de venta: una mejora notable sería la adición de otros tipos de combustibles (o fluidos, ya que no podemos usar combustible) al proceso, y que se gestione cada uno de manera independiente. Para ello deberíamos tener en cuenta sus datos en cuanto a densidad, ya que no fluirían de la misma manera por las mangueras, y también su ratio de demanda para tenerlo en cuenta a la hora de generar los instantes de tiempo en los que se producen las ventas de cada uno de ellos.



Figura 6.12: Dispensador de combustible con varios productos.

-Incorporación de nuevos tipos de fugas: en este TFG se ha supuesto que las fugas de combustible dependen del volumen del tanque de almacenamiento. En un futuro se podrían añadir otros tipos de fugas que dependan de la densidad del combustible, que se produzca en distintas zonas del depósito o que vengan dadas en función de la demanda

de combustible, es decir, cuanto más movimiento haya en la estación de servicio, más fugas se produzcan.

-Incorporación de nuevas distribuciones para sustituir a la aleatoria: otra pequeña modificación que podría hacerse es que los instantes de tiempo en los que se producen las ventas y las descargas vengan dados por otro tipo de distribución que no sea aleatoria. Podrían implementarse distintos métodos, como una distribución normal, binomial o, incluso, basada en datos reales recogidos por las estaciones de servicio.

- **Modificaciones en el simulador de la planta:** otra línea de trabajo podría ser la modificación del simulador como elemento docente para poder fomentar el uso de Arduino en las asignaturas de control. Podría simplificarse su funcionamiento y que los alumnos realicen un código de acuerdo con las indicaciones dadas por el profesor. Además, puede modificarse para que no corresponda a una estación de servicio, sino a cualquier otro ámbito que use estructuras basadas en depósitos, como puede ser una depuradora de agua o una planta industrial de embotellado, por ejemplo.

- **Cambio de microcontrolador:** en caso de que el proyecto experimente una ampliación significativa, es posible que un microcontrolador Arduino UNO no sea lo suficientemente potente como para poder soportar el código de la controladora. En este caso se consideraría la posibilidad de cambiar el Arduino UNO por otras opciones, como Raspberry Pi, Arduino MEGA u otros también bastante económicos y que presentan unas buenas prestaciones.



Figura 6.13: Microcontrolador Raspberry Pi.

- **Conversión de la planta a un laboratorio remoto:** otra línea de desarrollo que se podría estudiar es la reconversión de la planta a un laboratorio remoto en el cuál fuera posible iniciar su funcionamiento desde cualquier lugar. Podrían incluirse webcams para monitorear lo que estuviera sucediendo en el modelo y crear una página web desde la cual se pudiera controlar la planta y accionar el panel de control.
- **Sistema de detección de fugas:** dado que hemos simulado las fugas de un depósito, una línea inmediata de trabajo sería la implementación de un sistema que pudiera detectarlas. Como tenemos control sobre los parámetros de generación de fugas, sería muy fácil validar este sistema, ya que sus datos sobre las fugas deberían ser idénticos a la configuración de fugas que hayamos implementado

En definitiva, este proyecto es capaz de ofrecer un amplio rango de posibilidades para un estudiante de cualquier ingeniería industrial o incluso ingeniería informática, para así poder ampliarlo desde el punto de vista de cualquier carrera técnica y completarlo aún más. Las posibilidades son muchas, abriendo un gran abanico de ofertas de TFG que pueden proporcionarse al alumnado.



## ANEXOS

### Anexo 1: Presupuesto

En el siguiente anexo se expone el presupuesto que se desarrolló para la construcción del prototipo de la planta. Cabe destacar que es importante incluirlo porque, pese a que no se llegó a construir el prototipo, puede servir como información de referencia para aquellos que deseen retomar este proyecto para ampliarlo o completar las partes que no pudieron realizarse debido a la situación actual. Todos los componentes que se incluyen en este presupuesto han sido descritos en el Apartado 2 del Capítulo III:

**Tabla A.1: Presupuesto correspondiente a los materiales para la construcción del prototipo de la planta.**

Componente	Cantidad	Precio/un(€)	Precio (€)
Deposito 150 l	1	111	111
Válvulas solenoides	10	5	50
Mangueras	10	10	10
Bomba reposicionamiento de agua	1	20	20
Bomba de fugas	1	30	30
Perfusor de anestesia	1	125	125
Báscula deposito principal	1	25	25
Básculas para ventas y fugas	3	11	33
Módulo relés	1	1	5
Arduino Uno	2	5	10
Fuentes de alimentación	2	70	140
Estantería	1	22	22
		<b>Precio total</b>	<b>581</b>

Además de los materiales para implementar el prototipo, se ha incluido también el presupuesto por mano de obra en función de las horas empleadas para desarrollar este proyecto.

**Tabla A.2: Presupuesto correspondiente al trabajo de desarrollo del proyecto.**

Nº horas	Precio/hora	Precio total
300	25 €	7500 €

## Anexo 2: Guía del desarrollador. Controladora

En este anexo se detalla la estructura del código implementado en la controladora, hablando de que archivos lo componen, las características de la clase desarrollada y las partes que la forman.

Uno de los microcontroladores Arduino UNO es el que desempeña el papel de la controladora. El código está formado por 4 archivos:

- **Controladora.ino:** archivo principal que alberga la máquina de estados de la controladora, así como la definición de todos los elementos del panel de control.
- **Controladora.h:** archivo auxiliar de variables el cual contiene todas las constantes, definiciones de pines y variables locales del código.
- **Deposito.cpp:** archivo que alberga los métodos de la clase ‘Deposito’, su máquina de estados y el constructor.
- **Deposito.h:** archivo .h asignado a la clase ‘Deposito’, el cual nos indica cuál es la parte pública y privada del código.

En el archivo correspondiente a la controladora se empieza definiendo cuáles son los pines de entrada y salida del panel de control, ya sean los accionadores nombrados anteriormente o la alarma de rebosamiento. Seguidamente define cada uno de los tanques como objetos de la clase ‘Deposito’, la cual explicaremos más adelante y finalmente, empieza el bucle principal.

Con el fin de asegurar un código modular y fácilmente ampliable, se decidió generar una clase ‘Deposito’, cuyos objetos fueran todos los depósitos que conforman la planta, cada uno con sus propias características y particularidades. Cada uno tiene unos atributos que los definen y que son usados en la propia máquina de estados del depósito. Cabe recalcar que los atributos de la clase se han puesto en la zona privada y los métodos en la pública, para poder trabajar con estos fuera del propio archivo. Estos atributos son:

- **recarga\_o\_descarga:** en nuestra simulación, hay tanques en los que se controla el tiempo en el que se llenan, y otros en los que se vacían. Este atributo diferencia entre los dos tipos de tanques que puede haber. Si nos interesa su tiempo de llenado, se le asigna un 0, y si nos interesa su vaciado, será un 1.
- **es\_fugas:** simplemente nos indica que el tanque es o no el tanque de fugas. Necesitamos este parámetro ya que algunos procesos son distintos a los de los demás depósitos.
- **PIN\_llenado\_tanque:** atributo correspondiente al PIN de la válvula de llenado de ese tanque.
- **PIN\_vaciado\_tanque:** atributo correspondiente al PIN de la válvula de vaciado de ese tanque.
- **puerto\_sensor\_vol:** referencia al PIN correspondiente al sensor de volumen de ese depósito.
- **deposito\_sig:** es un puntero que señala a la posición de memoria del depósito siguiente, todo con el propósito de poder obtener datos de él y poder actuar en consecuencia. Por ejemplo, no podemos autorizar una descarga si el tanque siguiente no es capaz de contenerla.
- **capacidad\_máxima:** como su nombre indica, es la cantidad máxima de agua que puede contener el depósito en litros.
- **vol\_max\_seguridad:** cantidad de agua en litros inferior a la anterior, pero usada en los límites de seguridad para asegurarnos que nunca haya un desbordamiento.
- **volumen\_descarga\_max:** parámetro de la distribución aleatoria que marca el límite superior del rango de valores de volumen en litros.
- **volumen\_descarga\_min:** parámetro de la distribución aleatoria que marca el límite inferior del rango de valores de volumen en litros.



- **t\_prox\_descarga\_max:** parámetro de la distribución aleatoria que marca el límite superior del rango de valores de tiempo en segundos.
- **t\_prox\_descarga\_min:** parámetro de la distribución aleatoria que marca el límite inferior del rango de valores de tiempo en segundos.
- **estado\_actual:** parámetro que devuelve en todo momento el estado en el que se encuentra dicho depósito.
- **volumen:** parámetro que nos dice en todo momento la cantidad de agua en litros que tiene el depósito en su interior.
- **caudal:** este atributo es usado solo por el tanque de fugas, y nos indica que caudal se fuga del tanque de almacenamiento en ese ciclo.
- **vol\_prox\_mov:** este atributo contiene la cantidad de agua en litros que se descargará en el depósito en el próximo movimiento de la simulación.
- **t\_prox\_mov:** este atributo indica el instante en segundos en el cual se producirá el movimiento de la cantidad de agua anterior, también obtenido mediante la distribución aleatoria.

A parte de los atributos, tenemos los métodos de la clase, aquellos que nos permiten que los depósitos se comporten como nosotros queremos y realicen las operaciones necesarias para el funcionamiento de la controladora. Estos son los métodos definidos en la clase sin tener en cuenta el constructor:

- **actualizar\_estado:** método que implementa las transiciones y acciones de la máquina de estados de los depósitos que va definiendo las acciones que realizan las válvulas. Recibe como parámetros el tiempo de simulación, el estado de la seta de emergencia y un parámetro de reinicio de la simulación. Devuelve un booleano que es 'true' cuando el depósito entra en estado de emergencia y 'false' en cualquier otro caso.

- **abrir\_valvula\_llenado:** como su nombre indica, abre la válvula de llenado. En el caso del tanque de fugas, sobrescribe un valor de caudal en el PIN.
- **cerrar\_valvula\_llenado:** método que cierra la válvula de llenado.
- **abrir\_valvula\_vaciado:** como su nombre indica, abre la válvula de vaciado.
- **cerrar\_valvula\_vaciado:** método que cierra la válvula de vaciado.
- **nuevo\_vol\_descarga:** método que contiene el generador de valores para la distribución aleatoria en cuanto a volumen se refiere y nos da un nuevo valor cada vez que es llamado.
- **nuevo\_t\_prox\_mov:** método que contiene el generador de valores para la distribución aleatoria en cuanto a tiempo se refiere y nos da un nuevo valor cada vez que es llamado.
- **ready\_to\_recieve:** este método comprueba si el depósito puede recibir una cantidad determinada de agua. Mandará una respuesta en función de si es posible o no.
- **get\_volumen:** método que lee el sensor de volumen y nos da su valor para poder trabajar con él.
- **genera\_caudal:** este método solo lo usa el tanque de fugas. En función del volumen del tanque de almacenamiento genera un caudal que sale como salida analógica.
- **conversión\_volumen:** debido a que del sensor recibimos tensión, tenemos que convertir ese valor a volumen. En Arduino, los pines de entradas analógicas contienen un CAD (conversor analógico-digital) de 1023 niveles de tensión. Por lo tanto, para obtener el equivalente en volumen de la tensión recibida, debemos realizar la siguiente operación:

$$volumen = \frac{tension * capacidad\_máxima}{1023} \quad (A.1)$$

Antes de realizar este paso, tenemos que recordar que la tensión que nos llega después de pasar por el filtro presenta un rizado que hace que los valores recibidos disten de los

reales. Por ello, antes de realizar la conversión de tensión a volumen, debemos eliminar este rizado para poder trabajar con valores correctos (Capítulo V, Apartado 3).

### Anexo 3: Guía del desarrollador. Simulador de la Planta

En este anexo se explica la estructura del código implementado en el simulador de la planta, hablando de que archivos lo componen, las características de la clase desarrollada y las partes que la forman.

Uno de los microcontroladores Arduino UNO es el que desempeña el papel del simulador de la planta. El código está formado por 4 archivos:

- **Planta.ino:** archivo principal que gestiona las actualizaciones de volumen en cada ciclo del código y sobrescribe los sensores de los depósitos.
- **Planta.h:** archivo auxiliar de variables el cual contiene todas las constantes, definiciones de pines y variables locales del código.
- **Deposito\_sim.cpp:** archivo que alberga los métodos de la clase ‘Deposito\_sim’, sus atributos y el constructor.
- **Deposito\_sim.h:** archivo .h asignado a la clase ‘Deposito\_sim’, el cual nos indica cuál es la parte pública y privada del código.

El código contenido en el archivo correspondiente al simulador de la planta es bastante intuitivo. Al igual que en la controladora, se definen todos los depósitos como objetos de la clase definida como ‘Deposito\_sim’. A continuación, empieza el bucle principal, cuya función es únicamente actualizar el volumen de los depósitos activos e incrementar o disminuir el volumen de los pasivos en función de los movimientos que hagan los anteriores. En este código se definen como depósitos activos aquellos que generan descargas a partir de la distribución aleatoria. Estos son los dispensadores, el tanque de retorno y el tanque de fugas. Los pasivos son los que simplemente soportan las acciones de los anteriores. Estos son el tanque de retorno y el de almacenamiento. Finalmente se sobrescribe el volumen de los tanques en ese instante en el sensor para poder enviarlo a la controladora.

Esta clase 'Deposito\_sim', generara objetos que corresponderán con todos los depósitos que conforman la planta, cada uno con sus propias características y atributos. Estos atributos son:

- **valvula\_llenado\_tanque:** atributo correspondiente al PIN de la válvula de llenado de ese tanque.
- **valvula\_vaciado\_tanque:** atributo correspondiente al PIN de la válvula de vaciado de ese tanque.
- **sensor\_vol:** referencia al PIN correspondiente al sensor de volumen de ese depósito y por donde se envía el valor de volumen del tanque.
- **activo\_o\_pasivo:** este parámetro nos indica el tipo del tanque, 1 si es activo y 0 si es pasivo.
- **es\_fugas:** nos indica que el tanque es o no el tanque de fugas. Necesitamos este parámetro ya que algunos procesos son distintos a los de los demás depósitos.
- **deposito\_ant:** es un puntero que apunta a la posición de memoria del depósito anterior, todo con el propósito de poder obtener datos de él y poder actuar en consecuencia. Por ejemplo, no podemos aumentar el volumen del tanque si el anterior está vacío, así que tenemos que controlar ese dato.
- **limite\_tanque:** es la cantidad máxima de agua en litros que puede contener el depósito.
- **caudal\_entrada\_max:** caudal que entra al depósito para su llenado en l/min.
- **caudal\_salida\_max:** caudal que sale del depósito para su vaciado en l/min.
- **caudal:** parámetro exclusivo del tanque de fugas. Contiene el valor de caudal que nos ha enviado la controladora sobre cuanto se va a fugar y se presenta en ml/min.

- **incremento:** este atributo contiene cuánta agua va a entrar en el depósito en ese ciclo de código. Su valor depende de si la válvula de llenado está abierta o no y se presenta en litros.
- **decremento:** este atributo contiene cuánta agua va a salir del depósito en ese ciclo de código. Su valor depende de si la válvula de vaciado está abierta o no y se presenta en litros.
- **volumen:** parámetro que nos dice en todo momento la cantidad de agua en litros que tiene el depósito en su interior. Su valor se sobrescribe en el sensor.

A parte de los atributos, tenemos los métodos de la clase, aquellos que nos permiten que los depósitos se comporten como nosotros queremos y varíen el valor de volumen de estos para poder simular fielmente la planta. Estos son los métodos definidos en la clase sin tener en cuenta el constructor:

- **actualizar\_volumen:** método que actualiza el valor de la variable volumen de cada depósito, todo ello en función de si las válvulas de llenado y vaciado están abiertas o cerradas. Recibe como parámetro la duración del ciclo.
- **incrementar\_volumen:** aumenta el volumen de un depósito una cantidad determinada que nosotros le indiquemos.
- **disminuir\_volumen:** disminuye el volumen de un depósito una cantidad determinada que nosotros le indiquemos.
- **get\_incremento:** nos devuelve el valor del atributo incremento.
- **get\_decremento:** nos devuelve el valor del atributo decremento.
- **sobrescribir\_sensor:** método que plasma la variable volumen en el PIN de salida que corresponde con el sensor de volumen.

• **conversión\_volumen:** debido a que debemos mandar el valor de volumen como una señal analógica, debemos realizar una conversión. Los dispositivos Arduino UNO no poseen salidas analógicas. En su lugar, posee salidas digitales que pueden enviar señales PWM, que pueden actuar como señales analógicas si tenemos en cuenta su voltaje eficaz. Estas señales poseen una frecuencia de 500 Hz de frecuencia y 255 niveles de tensión; por lo tanto, la conversión debe hacerse de esta manera:

$$tensión = \frac{volumen * 255}{limite\_tanque} \quad (A.2)$$

Cabe recalcar que los atributos de la clase se han puesto en la zona privada y los métodos en la pública, para poder trabajar con estos fuera del propio archivo.

## REFERENCIAS

- [4] Alayón, S., Sigut, M., Arnay, R., Toledo, P., 2020. Time windows: the key to improving the early detection of fuel leaks in petrol stations. Safety Science, pp
- [5] El País, Barcelona, 29/04/2012, “Dos gasolineras en la fiscalía”,  
[https://elpais.com/ccaa/2012/04/28/catalunya/1335640290\\_886423.html](https://elpais.com/ccaa/2012/04/28/catalunya/1335640290_886423.html).
- [6] Europapress, Madrid, 1/12/2016, “Piden 4 años para el administrador de una gasolinera acusado de contaminar el Jarama”,  
<<https://www.europapress.es/nacional/noticia-piden-anos-administrador-gasolinera-acusado-contaminar-jarama-20161201174949.html>>.
- [7] Global Estaciones de Servicio SL, Murcia, 18/03/2020, “Surtidores para estaciones de servicio”, <<https://globalestacionesdeservicio.com/surtidores-para-estaciones-de-servicio-2/>>.
- [8] Moya Turbica, A., Escola Tècnica Superior d’Enginyeria Industrial de Barcelona, Octubre/2003, “Ampliación de las instalaciones de un supermercado: Construcción de una gasolinera”, pp 6-7, 18-19.
- [9] Sánchez Mejía E.J, Luis Marín H.A., Bogotá, Universidad Católica de Colombia, 2017, “Análisis de la amenaza presente en la infiltración de combustibles líquidos en las estaciones de servicio de la ciudad de Barranquilla”, pp 24-26.
- [10] Sigut, M. Alayón, S., Hernández, E., 2014. Applying pattern classification techniques to the early detection of fuel leaks in petrol stations. J. Clean. Prod. 80, 262-270.
- [11] Códigos Utilizados y vídeo explicativo. TFG-Luis-Arriaga. Github.  
<<https://github.com/LuisArriaga13/TFG-Luis-Arriaga-Campos>>