

MACERINO

Controlador Arduino para Macerador de Cerveza Artesanal

Fernando Chacón Sánchez-Molina

Software Libre y Compromiso Social

Domingo Ortiz Boyer

Escuela Universitaria Politécnica de Córdoba

Abril 2017

Copyright (C) 2017 Fernando Chacón Sánchez-Molina

personal@fernandochacon.com

Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.3 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts.

A copy of the license is included in the section entitled "GNU Free Documentation License".

Contenido

1.	Introducción.....	2
2.	Objetivos del trabajo.....	2
2.1.	Adaptación de un Equipo cervecero de nivel 2	2
2.2.	Programación del controlador Arduino.	3
3.	Proceso de elaboración de la cerveza	4
4.	Equipo para la elaboración de cerveza.....	5
5.	Adaptación del equipo cervecero	7
5.1.	Adaptación del HLT.....	7
5.2.	Adaptación del MLT	8
5.3.	Componentes necesarios	9
5.3.1.	Placa Arduino	9
5.3.2.	Interface con el usuario.....	13
5.3.3.	Sondas de medida de temperatura.....	14
5.3.4.	Calentador.....	20
5.3.5.	Conmutador del calentador	23
6.	Integración de todos los elementos.....	24
6.1.	Adaptación de la caja	25
6.2.	Instalación de baja tensión	27
6.3.	Instalación de alta tensión	29
6.4.	Resultado final	30
7.	Programación del controlador Arduino	31
7.1.	Lectura de sondas.....	32
7.2.	Control de la temperatura del HLT.....	33
7.3.	Monitorización	33

7.4.	Configuración.....	33
7.5.	Estableciendo la temperatura HLT.....	34
8.	Licencia de uso.....	35
9.	GNU Free Documentation License.....	38

Referencias

- [1] «*brewmaster.com*,» [En línea]. Available: <http://brewmasters.com.mx/shop/planta-cervecera-profesional-brewmasters3j80>.
- [2] «<https://www.dfrobot.com>,» [En línea]. Available: [https://www.dfrobot.com/wiki/index.php/Arduino_LCD_KeyPad_Shield_\(SKU:_DFR0009\)](https://www.dfrobot.com/wiki/index.php/Arduino_LCD_KeyPad_Shield_(SKU:_DFR0009)).
- [3] L. Llamas, «Luis Llamas, Ingeniería, Informática y Diseño,» [En línea]. Available: <https://www.luisllamas.es/temperatura-liquidos-arduino-ds18b20/>.
- [4] H. V. Gallón, *Fundamentos de Electrónica Industrial*, Universidad Pontificia Bolivariana, 2013.
- [5] «Wikispaces,» [En línea]. Available: <http://control-pid.wikispaces.com>.
- [6] Ó. T. Artero, *Arduino. Curso práctico de formación*, RC Libros, 2013.

1. Introducción

Como contribución al desarrollo del software libre, en este trabajo vamos a desarrollar un controlador Arduino para gestionar la maceración de cerveza artesanal con un equipo casero de pequeñas dimensiones. Este trabajo consistirá en diseñar, documentar y desarrollar un sistema Arduino para el control de la temperatura de maceración en un *Equipo cervecero de nivel 2*.

El trabajo se puede dividir en dos bloques según la naturaleza de las actividades a realizar:

- Adaptación de un *Equipo cervecero de nivel 2* para su control mediante Arduino, lo que incluye la instalación de sensores de temperatura y calentadores de agua en diversos puntos del equipo.
- Programación del controlador Arduino para la monitorización de las sondas y la gestión de la temperatura de maceración.

Tanto la documentación de montaje del controlador como el programa Arduino se distribuirán bajo tipo de licencia de software libre.

2. Objetivos del trabajo

2.1. Adaptación de un Equipo cervecero de nivel 2

En primer lugar, se documentará la adaptación de un Equipo cervecero de nivel 2 para poder adaptar un controlador Arduino para gestionar la temperatura de maceración. Se incluye la instalación de sondas de temperatura y calentadores de agua en diferentes puntos del equipo y su conexión a la placa Arduino. Se generará toda la documentación necesaria para la puesta en funcionamiento del sistema de control Arduino. La intención es que cualquier persona con unos mínimos conocimientos de electrónica sea capaz de adaptar el equipo de elaboración de cerveza para su control con Arduino, sin que necesariamente tenga que ser un experto en esta tecnología.

El controlador de todo el proceso de maceración será una placa Arduino conectada a varios sensores de temperatura, de forma que sea posible conocer la evolución de la temperatura en los puntos de interés del circuito de recirculación del mosto. Además, la placa Arduino interactuara con un calentador sumergido en el HLT para adaptar la temperatura de este a la temperatura de maceración. La conexión de los sensores de temperatura y de la resistencia a la placa Arduino responde a reglas de electrónica sencillas, donde generalmente se hace necesario utilizar un divisor de tensión para la conexión. Para realizar este montaje son necesarios unos

conocimientos de electrónica básicos que no siempre están al alcance del constructor de equipos cerveceros. El objetivo de esta fase de diseño será realizar la documentación que guíe en la construcción del hardware de interconexión de la placa Arduino con los sensores de temperatura y calentadores, de forma que se facilite la implantación del sistema a personas con escasos conocimientos de electrónica.



Equipo cervecero de nivel 2 [1]

2.2. Programación del controlador Arduino.

Se diseñará y desarrollará el programa que ejecutará la placa Arduino y cuya función será mantener el mosto a la temperatura fijada para la maceración. Para ello, realizará la lectura de los sensores de temperatura localizados en el circuito de maceración e interactuará con el calentador ubicado en el HLT para mantener la temperatura deseada.

La placa Arduino se completa con un shield de pantalla y botonera que será utilizado para mostrar la lectura de la temperatura de las diferentes sondas, así como para permitir al usuario fijar la temperatura de trabajo en el proceso de maceración.

La aplicación no realiza un procedimiento especialmente complicado. Básicamente consiste en realizar la lectura de los diferentes sensores instalados en el equipo de maceración, realizar una estimación de la temperatura real y, en caso de que la temperatura de trabajo sea menor que la temperatura de maceración establecida, realizar el encendido del calentador situado en el HLT para elevar la temperatura hasta el punto deseado. Durante este proceso también habrá

que actualizar los datos presentados en la pantalla y comprobar si el usuario ha interactuando mediante la botonera para modificar la temperatura de maceración.

3. Proceso de elaboración de la cerveza

No es objeto de este trabajo explicar cómo se produce la cerveza, pero es necesario conocer a grandes rasgos en qué consiste el proceso para entender el producto que vamos a desarrollar. Para la elaboración de la cerveza se debe seguir una serie de pasos, que son prácticamente los mismos si el procedimiento es casero, artesanal o industrial. A grandes rasgos, los procesos a realizar son los siguientes:

- Maceración. El proceso parte de una mezcla de agua y malta. El objetivo de este proceso es someter la malta a una temperatura determinada, de manera que se den las condiciones para que se produzca una acción enzimática que rompa las cadenas largas de azúcares en cadenas más cortas que faciliten su extracción. El resultado de la acción enzimática depende de diversos factores, como las características del agua (PH, dureza del agua, contenidos en sales, ...), el tipo de malta utilizada y la temperatura a la que se desarrolle el proceso.
- Mashout. Cuando se estima que se ha completado el proceso de extracción de azúcares, se eleva la temperatura del mosto, de manera que se alteran las condiciones de maceración y se detiene el proceso enzimático. Además, al elevar la temperatura del mosto, este se hace menos denso, de forma que se facilita el lavado de la malta para recoger los azúcares extraídos.
- Hervido. El mosto obtenido en el proceso anterior se hierve con el doble propósito de eliminar proteínas que pueden dar mal sabor a la cerveza y realizar la esterilización para acabar con bacterias, mohos y microorganismos que tendrían una influencia negativa en el proceso de fermentación. En esta etapa se realiza también la aportación de lúpulo, que dará a la cerveza el toque amargo y aromático característico.
- Fermentación. En este paso se añaden levaduras al mosto. Las levaduras utilizan los azúcares extraídos en el proceso de maceración para obtener la energía necesaria para mantenerse vivas y reproducirse, y como resultado de esto, aportan alcohol y CO_2 a lo que, ahora sí, es cerveza.

Hemos visto la importancia que tiene la temperatura a lo largo de todo el proceso de fabricación de la cerveza. En el caso del proceso de maceración, la temperatura baja favorece la extracción de azúcares fermentables, que serán utilizados por las levaduras para alimentarse. Sin embargo, la maceración a alta temperatura tiene como resultado la extracción de azúcares no fermentables. Se trata de azúcares que no pueden ser asimilados por las levaduras, por lo que permanecerán en la cerveza dando un sabor dulce a esta. El equilibrio de ambos tipos de azúcares dará, entre otros muchos factores, el carácter y sabor de una cerveza. El punto de temperatura a partir del cual se realiza la extracción de azúcares no fermentables es muy sensible. Depende básicamente del tipo de malta utilizada, aunque un valor típico de referencia es de 66 grados. Solo un grado por encima o por debajo altera la naturaleza de azúcares extraídos. Las cervezas con más cuerpo, más dulces, más densas, requieren extraer mayor proporción de azúcares no fermentables, y por lo tanto se maceran por encima del punto de equilibrio. Las cervezas con menos cuerpo, más ligeras, se maceran a temperaturas inferiores. Por tanto, es muy importante poder controlar de forma muy precisa la temperatura de maceración, por una parte, para lograr obtener una cerveza con las características deseadas, y por otra, para establecer un procedimiento que sea posible reproducir fielmente en el futuro para poder obtener nuevos lotes de cerveza que mantengan las mismas características.

Aunque el control de la temperatura de maceración se ha realizado tradicionalmente con procedimientos manuales, la aparición de las nuevas tecnologías permite aplicar técnicas que facilitan notablemente obtener un control más preciso del proceso. Esto es especialmente útil en un entorno doméstico, donde no se cuenta con grandes medios para controlar el proceso de elaboración de cerveza.

4. Equipo para la elaboración de cerveza

El equipo necesario para realizar el proceso de elaboración de cerveza a nivel casero ha sido estudiado y desarrollado por la comunidad de homebrewers¹ hasta lograr un equipo optimizado en cuanto a componentes y funcionalidad. Existen varios diseños según la complejidad

¹ Cerveceros caseros, término inglés que no cuenta con traducción en español. No se trata de un grupo en particular de desarrolladores, sino que es el nombre que se le daría a los aficionados a la elaboración de cerveza casera.

y medios disponibles, pero en nuestro caso nos centraremos en el *Equipo cervecero de nivel 2*, ya que se trata de un diseño muy extendido en la comunidad que permite utilizar el equipo para completar casi todas las fases del proceso de maceración. Este equipo está formado por tres ollas para realizar el calentamiento del agua en cada proceso de elaboración, dos bombas para la recirculación y varias llaves para controlar el circuito de circulación del mosto. La parte que nos interesa en cuanto al procedimiento de maceración es el MLT² y el HLT³. El proceso de maceración se realiza depositando la malta y agua caliente en el MLT, pero el HLT se utiliza como un medio auxiliar para mantener agua caliente para utilizar posteriormente en el proceso de lavado, y, simultáneamente, es utilizado como medio para calentar el mosto que se encuentra en el MLT. Como vimos anteriormente, deseamos mantener la temperatura del mosto de maceración a una temperatura determinada. Para ello, contamos con que el HLT dispone de una resistencia y un serpentín. La temperatura del HLT se mantiene a la temperatura deseada de maceración, y mediante una bomba se extrae el mosto de la parte inferior de la olla de maceración (MLT) y se hace circular por el serpentín para que, mediante transferencia térmica, alcance la temperatura de maceración. Posteriormente se manda de vuelta a la olla de maceración, lo que produce una recirculación constante del mosto que, por una parte, al circular entre el grano favorece la extracción de los azúcares y por otra, al circular por el serpentín, mantiene a la temperatura de maceración.

La función del controlador Arduino que vamos a desarrollar para controlar el proceso de maceración será doble:

- Servirá para monitorizar la temperatura del mosto y de los tanques en diferentes puntos, de manera que sea posible auditar que la temperatura de maceración sea la correcta.
- Controlará la temperatura de maceración. Cuando disminuya la temperatura del mosto, actuará en el calentador situado el HLT para mantener la temperatura de maceración.

² Mash Lauter Tun, olla de maceración

³ Hot Liquor Tank, olla de agua caliente

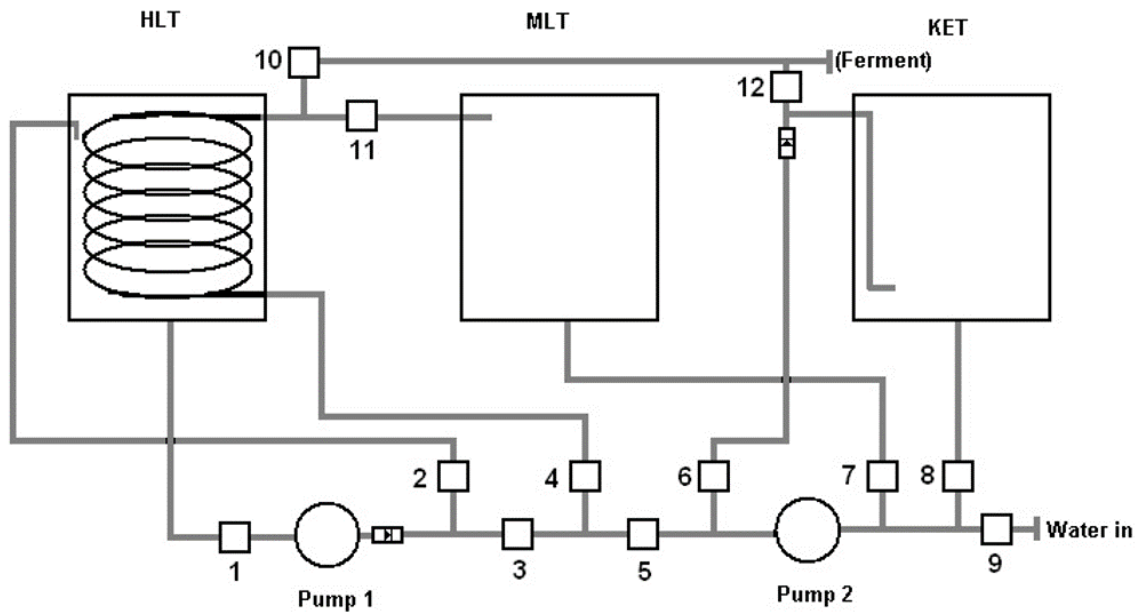


Figura 1 Esquema de Equipo cervecero de nivel 2

5. Adaptación del equipo cervecero

Vimos anteriormente que el Equipo cervecero de nivel II utiliza un proceso de recirculación del mosto procedente del MLT a través de HLT para compensar la pérdida de calor que sufre el sistema. El HLT dispone de un sensor de temperatura y un calentador que mantiene la temperatura del agua a la temperatura de maceración. El mosto procedente del MLT circula a través de un serpentín sumergido en el agua del HLT para que adquiera la temperatura de maceración, y posteriormente se manda de vuelta al MLT para recircular a través del grano.

El HLT dispone a su vez de una segunda recirculación de agua. Mediante una bomba se extrae agua de la parte inferior de la olla y se retorna por la parte superior, procurando que este flujo de agua produzca dentro del HLT un movimiento rotacional del agua. La finalidad de esta actividad es mejorar el intercambio térmico que se produce entre el agua del HLT y el serpentín sumergido. Si el agua del HLT se mantuviera estática, el intercambio de calor no sería tan efectivo.

Visto lo anterior, la función del controlador del macerador podría simplificarse en controlar la temperatura del HLT para que se mantenga a la temperatura de maceración. Además, vamos a completar el equipo añadiendo sensores de temperatura adicionales.

5.1. Adaptación del HLT

El controlador debe mantener la temperatura del agua contenida en el HLT a la temperatura de maceración. La temperatura típica de maceración es de 66 grados, aunque según

el tipo de cerveza se puede necesitar macerar a temperaturas que oscilan entre los 42 y los 76 grados. En todo caso, se trata de temperaturas superiores a la temperatura ambiente, por lo que el sistema siempre va a tener tendencia a perder energía y bajar la temperatura, por lo que debemos dotarlo de una resistencia que compense la pérdida de calor y mantenga la temperatura del agua. La resistencia irá encastrada en el HLT de manera que quede sumergida en el agua.

Para el control de la temperatura del HLT dispondremos de un termómetro que estará en contacto con el agua. El sistema Arduino utilizará el termómetro como termostato para activar la resistencia cuando la temperatura del agua baje de la deseada.

En la siguiente figura puede verse la disposición de la resistencia y el termóstato instalados en el HLT. La resistencia se coloca en la parte inferior de la olla, mientras que para instalar el termóstato se aprovecha la salida inferior del HLT por el que se realiza la recirculación de agua.

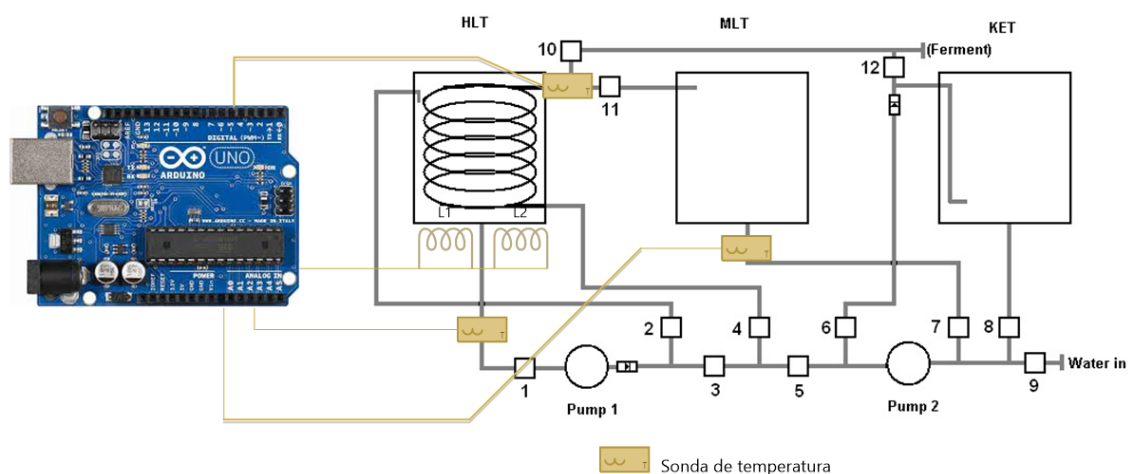


Figura 2 Esquema básico de conexión del equipo cervecero al Arduino

Para realizar la gestión de temperatura durante el proceso de maceración va a ser necesario contar con una serie de equipamiento que se instalará en el equipo cervecero. En los siguientes puntos veremos cuál será el equipamiento necesario, donde y como se adaptará al equipo.

5.2. Adaptación del MLT

Durante el proceso de maceración es muy importante monitorizar la temperatura a la que se está desarrollando la actividad enzimática, ya que se trata de un factor determinante para establecer el carácter de la cerveza que se está produciendo. La toma de temperatura durante la maceración podría realizarse instalando un termómetro dentro del MLT, en la zona donde se deposita el grano. De esta forma obtendríamos la temperatura exacta de maceración. Sin embargo,

ubicar un termómetro en este lugar tiene el inconveniente de que resulta molesto para realizar diversas actividades que se realizan durante el proceso de elaboración, como puede ser remover el grano para evitar apelmazamiento, extraer el falso fondo de filtrado o durante las tareas de limpieza del equipo.

Puesto que el equipo utiliza una bomba para recircular el mosto, extrayéndolo de la parte inferior y reintroduciéndolo por la parte superior del MLT, y ya que este proceso es aprovechado para recircularlo por el serpentín del HLT para corregir la temperatura de maceración, se ha optado por instalar un termómetro en la salida del MLT y otro en la entrada. De esta forma obtenemos la temperatura en dos puntos muy importantes del proceso de producción. Por una parte, al medir la temperatura del mosto que extraemos del HLT, obtenemos la temperatura a la que se está realizando la maceración, y, por otra parte, al obtener la temperatura después de recircular por el serpentín del HLT, podemos comparar la temperatura de entrada y salida del serpentín y obtener la eficiencia del uso del HLT como calentador.

En la siguiente imagen podemos la instalación de una sonda de temperatura en la salida del MLT, lo que corresponde también con la entrada en el serpentín, y otra en la entrada del MLT, que corresponde con la salida del serpentín.



Figura 3 Sondas de temperatura en la salida y entrada del MLT

5.3. Componentes necesarios

5.3.1. Placa Arduino

El cerebro del sistema será una placa Arduino. Este tipo de placa dispone de un controlador, una memoria y unos terminales de entrada/salida, de manera podemos cargar un programa que se ejecuta de forma cíclica y cuyo funcionamiento básico será realizar la lectura de

los sensores conectados a la placa, evaluar los datos de entrada y mandar órdenes de actuación a los dispositivos conectados.

Estamos desarrollando un proyecto de software libre, por lo que es importante utilizar productos que utilicen licencias compatibles. Arduino se distribuye bajo licencia Creative Commons Attribution Share-Alike. Arduino utiliza un entorno de desarrollo IDE con licencia GNU, una colección de librerías con licencia GNU LGPL. Arduino se puede programar en C++ o utilizando un lenguaje de programación propio de código abierto basado en Processing.

Existen varios modelos de placas Arduino en función del procesador, la memoria que incorpore y el número de conexiones de entrada y salida. En nuestro caso, el proceso a realizar va a ser bastante sencillo, por lo que elegiremos un modelo básico, concretamente un Arduino UNO. Esta placa se caracteriza por integrar un procesador ATmega328P a 16MHz. Dispone de una memoria Flash 32KB para almacenar el programa a ejecutar por la placa, 2KB de memoria SRAM para almacenar las variables del programa y 1KB de memoria EEPROM para guardar información de forma permanente incluso sin alimentación. Para la conexión de dispositivos cuenta con 6 entradas analógicas y 14 entradas/salidas digitales, de las cuales 6 disponen de PWM. Estas características son suficientes para el proyecto que se desea realizar.



Figura 4 Arduino UNO

5.3.1.1. Pines de conexión

La placa Arduino UNO, que será la que utilizaremos en este proyecto, dispone de varios pines para la conexión de componentes externos. En la siguiente figura se puede ver un esquema de la placa con los diferentes pines:

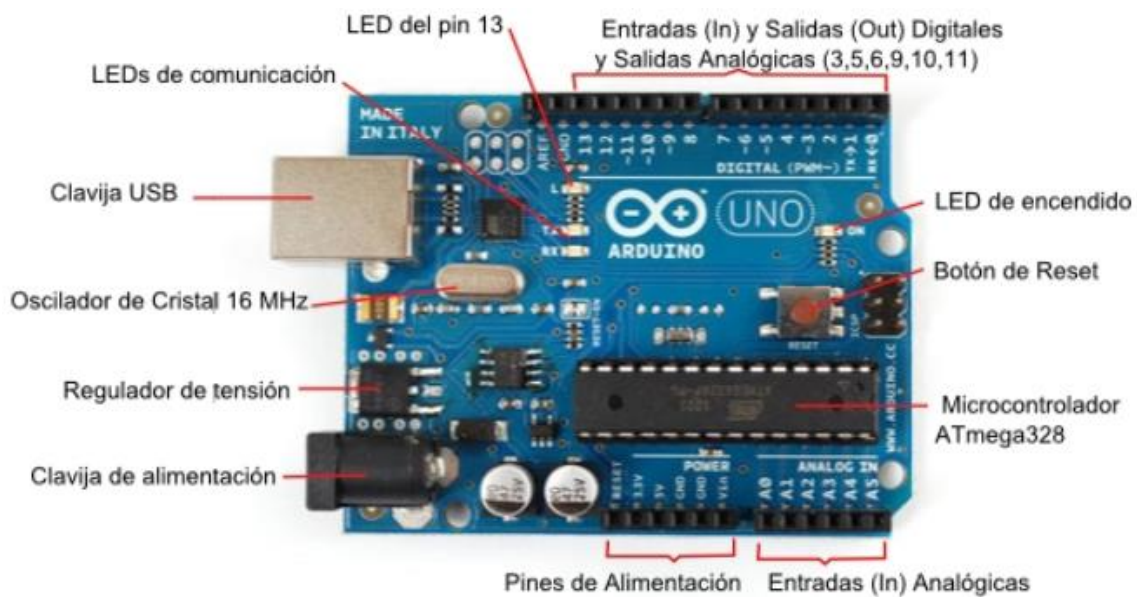


Figura 5 Conexiones de una placa Arduino UNO

Según el modo de funcionamiento, disponemos de los siguientes pines de conexión:

- Entradas analógicas. Dispone de 6 entradas analógicas etiquetadas como *ANALOG IN A0, A1, A2, A3, A4, A5* y *A6*. Utilizando estas entradas es posible medir diferencia de potencial comprendido entre 0 y 5v, con una resolución de 8 bits. En ellas podríamos conectar dispositivos que ofrecieran señales analógicas. En nuestro proyecto, las termoresistencias las podríamos conectar en estos pines, aunque ya veremos posteriormente que no haremos uso de ellas.
- Entradas y salidas digitales. Dispone de 14 entradas digitales que identifican valores LOW para entradas entre 0 y 2 v o valores HIGH entre 2 y 5v. Estos pines están etiquetados como *DIGITAL (PWM)*. Aunque se trata de entradas/salidas digitales, algunos puertos se pueden utilizar en modo *PWM* (*pulse-width modulation*), lo que quiere decir que podrían utilizarse para simulación de entradas/salidas analógicas mediante la modulación de la señal por ancho de pulso. En nuestro caso, no haremos uso de *PWM*, ya que los dispositivos que conectaremos a la placa serán varias sondas *DS18B20* con comunicación digital 1-wire, y el interruptor de estado sólido, que también funciona en modo digital.

En la siguiente tabla se muestran la asignación de pines que utilizaremos para la conexión de los dispositivos conectados a la placa Arduino:

Pin	Dispositivo	Función
Analog 1	Pulsador	Subir temperatura maceración
Analog 2	Pulsador	Bajar temperatura maceración
Analog 0	LCD	Botones control
Digital 4		DB4
Digital 5		DB5
Digital 6		DB6
Digital 7		DB7
Digital 8		RS (Data or Signal Display Selection)
Digital 9		Enable
Digital 10		Backlit Control
Digital 3	SSR	Control
Digital 11	Sonda HLT.	Datos 1-wire
Digital 12	Sonda MLT out.	Datos 1-wire
Digital 13	Sonda MLT in.	Datos 1-wire

Figura 6 Uso de los pines

5.3.1.2. Alimentación de la placa

La placa Arduino necesita una alimentación de 5v para su correcto funcionamiento. Disponemos de dos formas diferentes para proporcionar tensión a la placa:

- Conexión de la placa a una fuente de alimentación externa mediante una clavija de 2,1 mm con una tensión entre 7 y 12 voltios. Puesto que la placa funciona internamente con 5v, la tensión suministrada es rectificada a este voltaje.
- Conexión de la placa mediante alimentación USB. La placa Arduino dispone de un conector USB tipo B para la conexión a un PC, lo que permite transferir el programa a la placa o la comunicación con una consola. Esta conexión se puede utilizar también para proporcionar la alimentación requerida por la placa. En estas circunstancias, la placa se alimentará con 5v y dispondrá de un máximo de 500 mA para el funcionamiento propio y alimentar los dispositivos conectados, como pueden ser las sondas de temperatura, interruptores SSR o leds.

En nuestro caso vamos a conectar pocos elementos a la placa Arduino y estos son de bajo consumo, por lo que la conexión USB es suficiente para alimentar todos los dispositivos. Además,

este tipo de conexión es más cómoda que la fuente de alimentación externa, ya que solo necesitamos un simple cargador USB y un cable USB tipo B (el típico de impresora o discos externos) para hacer funcionar la placa.

5.3.1.3. Conexión al PC

El desarrollo de aplicaciones para Arduino se realiza en un PC mediante el IDE Arduino. Mediante esta aplicación se realiza la codificación y compilación del programa, y posteriormente este es transferido a la placa Arduino. La transferencia del programa se realiza conectando la placa al ordenador mediante un cable USB tipo B. Una vez finalizada la transferencia, es posible desconectar la placa del PC y el programa cargado en la placa se ejecutará sin necesidad de utilizar el PC. Solo será necesario proporcionar alimentación eléctrica a la placa según vimos en el punto anterior.

5.3.2. Interface con el usuario

Con el fin de que el sistema sea autónomo y que no sea necesario utilizar un ordenador conectado al Arduino, la placa se completará con una pantalla y una botonera que hará las veces de interface de entrada/salida con el usuario. Para simplificar el proceso de montaje se ha elegido un shield que incorpora en una sola placa la pantalla y la botonera. Por su popularidad y precio, el modelo elegido es el LCD KeyPad Shield SKU DFR0009.



Figura 7 LCD KeyPad Shield SKU DFR0009

5.3.2.1. Instalación y conexión

La pantalla LCD se distribuye en formato shield, lo que permite su conexión con la placa Arduino simplemente *pinchando* las placas mediante los pines de conexión.



Figura 8 LCD Keypad Shield conectada a una placa Arduino

Al instalar la pantalla en la placa se ocupan todos los pines del Arduino, aunque en realidad no utiliza todos ellos. En la siguiente tabla se muestra el esquema de los pines utilizados por la pantalla:

Pin	Function
Analog 0	Button (select, up, right, down and left)
Digital 4	DB4
Digital 5	DB5
Digital 6	DB6
Digital 7	DB7
Digital 8	RS (Data or Signal Display Selection)
Digital 9	Enable
Digital 10	Backlit Control

Figura 9 Pines de conexión de la pantalla LCD Keypad Shield [2]

Esta información habrá de tenerse en cuenta a la hora de seleccionar los pines para la conexión del resto de los dispositivos, de forma que no se utilicen pines que entren en conflicto entre varios componentes.

El inconveniente de utilizar la pantalla en formato *shield* es que este formato ocupa todos los pines de conexión de entrada/salida de la placa Arduino, donde deberemos conectar las entradas de los sensores de temperatura y el controlador del calentador. Puesto que los pines se encontrarán ocupados, aunque sin uso, el resto de las conexiones las tendremos que hacer mediante soldadura a la placa del LCD.

5.3.3. Sondas de medida de temperatura

Como sensor de temperatura es posible elegir entre sondas *Tipo k* y *DS18B20*. Las primeras pertenecen al grupo de termo resistencias, es decir, se trata de una resistencia cuyo valor varía en función de la temperatura de trabajo. Son sondas muy sencillas de utilizar, ya que solo hace falta establecer la relación existente entre la caída de tensión en la resistencia y la temperatura leída. Tienen el inconveniente de que no son muy precisas y que, debido a las

limitaciones de lecturas analógicas de la placa Arduino, su resolución es demasiado baja (la cantidad de valores que la sonda es capaz de captar está limitada, lo que hace que la distancia entre dos valores próximos que la sonda es capaz de leer sea demasiado grande, del orden de medio grado).

Las sondas *DS18B20* podrían considerarse como sondas digitales. Una vez realizada la lectura de la temperatura, la sonda se comunica con la placa Arduino estableciendo una conexión digital mediante el protocolo *1-wire*. La ventaja de este tipo de sondas es que presenta mayor precisión que las de *tipo k*, aunque el procedimiento de conexión y comunicación con la placa Arduino es algo más complejo. Otra ventaja de este tipo de sondas es que permite la conexión en serie de numerosas sondas, lo que permite utilizar gran cantidad de ellas utilizando un solo pin de conexión de la placa Arduino.

Este tipo de sensores se puede encontrar en diferentes formatos de encapsulado. Los más populares son la configuración como integrado TO-92 y como sonda impermeable en vaina. En nuestro caso utilizaremos la segunda opción, ya que la sonda se instalará de manera que quede en contacto con los líquidos, por lo que necesitamos que el encapsulado sea aislante para permitir sumergir el componente en el mosto.



Figura 10 Sonda de temperatura DS18B20 en formato integrado TO-92 y en vaina

5.3.3.1. Instalación

La sonda la instalaremos en tres puntos diferentes del circuito de recirculación:

- Salida de MLT. Temperatura del mosto en el MLT, temperatura de maceración.
- Entrada del MLT. Medirá la temperatura de retorno del mosto después de pasar por el intercambiador de calor en el HLT (serpentin).
- Salida del HLT. Temperatura del agua del HLT.

Todas las sondas se instalarán en tomas de salida o entrada de las ollas. Estas tomas ya existen en el equipo y son utilizadas para realizar la recirculación del mosto, lo único que habrá que hacer es adaptar las tomas para instalar la sonda.

Las sondas a utilizar vienen configuradas en una vaina de acero de 6mm de diámetro y 50mm de longitud. No disponen de ningún tipo de sistema de fijación mediante rosca o presión, por lo que necesitaremos adaptarlas a las tomas de salida del equipo. Para ello utilizaremos un acoplamiento de compresión de acero inoxidable de 6mm con rosca macho BTSP de 1/8". Mediante este mecanismo conseguimos incrustar la sonda en el acoplamiento y queda sellada por la compresión que ejerce una pequeña arandela de aluminio que se adapta a la vaina mediante la presión de la rosca. Este sistema garantiza la estanqueidad del montaje y dota a la sonda de un sistema de instalación por rosca.



Figura 11 Acoplamiento de compresión y Sonda en la montura

Para instalar el acople de la sonda a las salidas de las ollas, utilizaremos una T con rosca BTSP hembra de 1/2". Esta pieza la intercalaremos entre la toma de las ollas y las llaves de circulación, y en la otra toma enroscaremos el acople de la sonda. Para garantizar el sellado de las roscas utilizaremos cinta de teflón.



Figura 12 Montaje del acoplamiento de la sonda y la T

5.3.3.2. Guía de conexión

La sonda DS18B20 utiliza el protocolo de comunicación 1-wire, que es un protocolo propietario de Dallas Semiconductor. A pesar de ser propietario, se puede utilizar sin necesidad

de realizar ningún pago adicional por su uso, ya que la licencia de uso se paga en el momento de realizar la compra de la sonda.

La sonda dispone de tres terminales:

- Vq. Línea de datos. Cable amarillo.
- Vdd. Línea de alimentación. Cable rojo.
- GND. Toma de tierra. Cable negro.

La sonda funciona con tensión de entre 3 y 5,5v, por lo que es posible conectar la sonda directamente a una de las todas de entrada/salida del Arduino. En la siguiente imagen se pueden distinguir las líneas correspondientes a los diferentes encapsulados:



Figura 13 Patillas de conexión de los integrados DS18B20

Los dispositivos 1-wire pueden funcionar en dos configuraciones diferentes:

- Modo normal. Esta configuración utiliza las tres conexiones de la sonda para su funcionamiento. La alimentación del dispositivo se realiza mediante las tomas Vdd y GND, mientras que la comunicación se realiza sobre la línea Vq.
- Modo parásito. La principal ventaja que presentan los dispositivos que utilizan el protocolo 1-wire es que, a pesar de disponer de tres conexiones diferentes, es posible realizar la alimentación y comunicación utilizando únicamente dos cables. Para ello, la sonda se alimenta utilizando la línea de transferencia de datos y la toma de tierra. La sonda dispone de un condensador que es utilizado para mantener la alimentación mientras se realiza la transferencia de datos, volviendo a activar esta línea como toma de alimentación una vez que se concluye la comunicación.

En ambos casos es necesario instalar una resistencia pull-up de 4k7 entre Vcc y Vq.

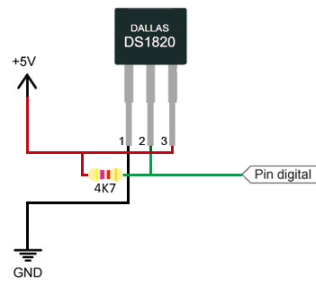


Figura 14 Esquema de conexión del DS18B20 [3]

En el protocolo 1-wire la comunicación de datos se realiza de forma digital, por lo que este tipo de sondas presenta respecto a las sondas de tipo-k (analógicas) la ventaja de que permiten utilizar cables de mayor longitud sin que ello altere el valor transmitido, además de ser inmune a interferencias electromagnéticas externas.

En la siguiente figura se muestra el esquema de conexión en serie de varias sondas DS18B20 a una placa Arduino en configuración normal, utilizando los tres cables de las patillas del componente, y en la siguiente el esquema equivalente para la configuración en modo parásito.

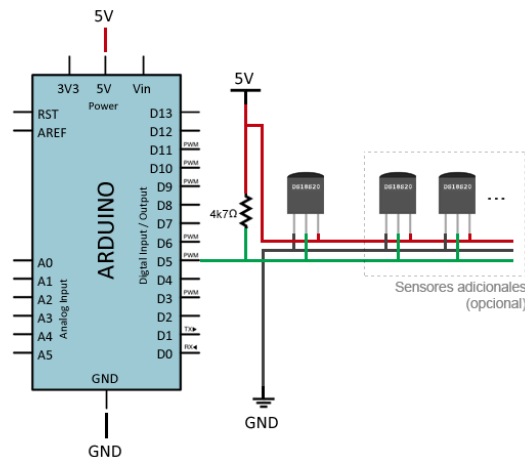


Figura 15 Conexión de DS18B20 y Arduino mediante configuración normal [3]

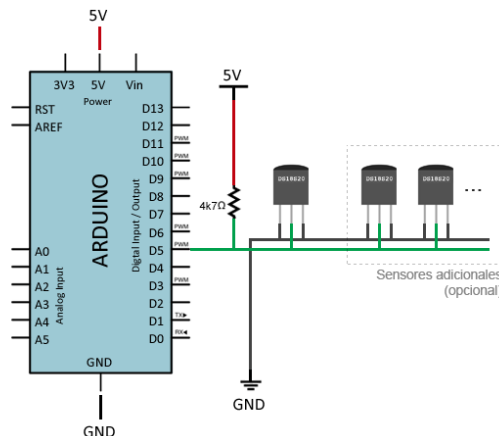


Figura 16 Conexión de DS18B20 y Arduino mediante configuración en modo parásito [3]

Cada dispositivo 1-wire dispone de una memoria ROM en la que se graba en fabrica un identificador único de 64 bits. Los 8 primeros bits corresponden a la familia (0x28 para el DS18B20). Los siguiente 48 bits son el número de serie único. Los últimos 8 bits son un código CRC.

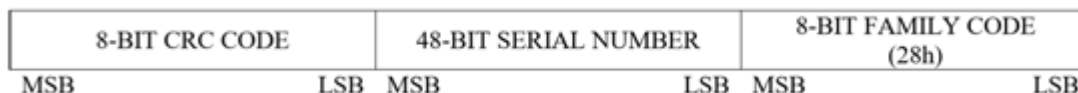


Figura 17 Direccionado de un componente 1-wire [3]

Esta cadena se utiliza a la hora de lanzar la orden de lectura de la temperatura para identificar la sonda que debe transmitir la lectura de la temperatura. Mediante este procedimiento es posible conectar varias sondas DS18B20 en serie utilizando una sola línea de datos. La sonda identificada realizará la lectura de la temperatura y devolverá el valor en formato digital por la línea de comunicación. Este proceso se trata de un procedimiento complicado y lento, que requiere un tiempo de lectura aproximado de 750 milisegundos.

El sistema de direccionamiento 1-wire permite instalar 2^{48} (más de 218 billones) de sondas en serie para un solo pin del Arduino. Sin embargo, esta forma de instalar las sondas tiene el inconveniente de que es necesario conocer el identificador de la sonda que queremos que realice lectura e indicarlo en el momento de lanzar la orden. Por lo tanto, necesitamos conocer el identificador de la sonda en el momento de compilar el programa, además de que no es posible sustituir la sonda a posteriori. Como solución a este inconveniente, se puede establecer un procedimiento de configuración y lectura del identificador de la sonda como parte del programa, pero esto complica notablemente la programación. En nuestro caso, como solo vamos a trabajar con tres sondas y disponemos de pines libres en el Arduino, vamos a utilizar una sola sonda por cada pin. De esta forma se puede lanzar la orden de lectura sin necesidad de conocer el identificador de la sonda.

Disponemos de tres sondas diferentes, que instalaremos, una en el HLT y dos en la entrada y salida del MLT. Cada sonda irá conectada a un pin digital diferente de la placa Arduino. En la siguiente tabla se muestra el esquema de conexión de las sondas:

Dispositivo	Conector	Pin Arduino	Función
Sonda HLT.	Vd (amarillo)	Digital 11	Datos
	Vdd (rojo)	5v	Alimentación
	GND (negro)	GND	tierra
Sonda MLT out.	Vd (amarillo)	Digital 12	Datos
	Vdd (rojo)	5v	Alimentación
	GND (negro)	GND	tierra
Sonda MLT in.	Vd (amarillo)	Digital 13	Datos
	Vdd (rojo)	5v	Alimentación
	GND (negro)	GND	tierra

Figura 18 Tabla de pines de conexión de las sondas al Arduino

Hay que recordar que tenemos que instalar una resistencia de 4k7 entre Vcc y Vq de cada sonda.

Las sondas quedarán montadas en el MLT y HLT de forma permanente, e irán conectadas a la placa Arduino. Para hacer portable el equipo cervecero, la conexión de las sondas se hará mediante clavijas que permitan desmontar el equipo y guardarlo por separado. Para ello utilizaremos conexiones *Jack* de 3,5 mm. Se trata de conectores muy populares y económicos, por lo que no es complicado encontrarlos en tiendas de electrónica. En la siguiente imagen podemos ver el esquema de conexión de la sonda al conector Jack.



Figura 19 Conexión de la sonda de temperatura al Jack

5.3.4. Calentador

El calentador será el elemento encargado de elevar y mantener la temperatura de los líquidos del sistema. Este se encontrará sumergido en el tanque HLT y estará en contacto con el agua, por lo que necesitamos que sea un elemento diseñado específicamente para trabajar en estas condiciones. Además, el agua que tiene que calentar será añadida finalmente al mosto

mientras que se realiza el proceso de lavado del grano, por lo tanto, es de vital importancia que cumpla con la normativa de procesamiento de productos alimentarios. En el mercado existen gran variedad de calentadores que pueden ser acoplados a una olla. Al tratarse de un equipamiento casero, se busca que se trate de un producto económico que cumpla con los requisitos exigidos. Aunque existen productos diseñados específicamente para fines industriales, estos tienen un coste demasiado elevado para el uso personal. En estos casos, la comunidad cervecera suele utilizar calentadores para calderas eléctricas de agua caliente sanitaria. En este caso, ya que no se trata de un producto específico para la industria alimentaria, es necesario comprobar que cumple con los requisitos que la normativa exige para estos casos. Gran parte de estos calentadores están fabricados con una vaina sumergible de bronce. Este tipo de calentadores no son recomendables, aunque el bronce por sí mismo no es tóxico, sí que se corre el riesgo de que se forme óxido en su superficie y este puede ser perjudicial para la salud. Otros calentadores presentan una vaina de bronce recubierta con una capa de otro material con aspecto de acero. Sin embargo, este puede llegar a ser aún más perjudicial que la vaina sin recubrimiento, ya que, el calentador puede llegar a alcanzar altas temperaturas que ocasionen que el recubrimiento se desprenda y pueda llegar a tener contacto con la cerveza.

Preferentemente se utilizará un calentador formado por la vaina de acero. Las propiedades del acero son ideales para su utilización en la industria alimentaria. Se trata de un material resistente a las altas temperaturas de trabajo del calentador, no presenta riesgo de oxidación y es de fácil limpieza.

A la hora de dimensionar el calentador, habrá de tenerse en cuenta que inicialmente deberá calentar 30 litros de agua desde temperatura ambiente, estimada en 18 grados, hasta la temperatura típica de maceración de 66 grados. Con la práctica se ha determinado que este proceso lleva 40 minutos si se utiliza una resistencia de 230v y 3000w. Aunque podría utilizarse una resistencia de mayor potencia para acortar los tiempos, hay que tener en cuenta que estamos diseñando un equipo casero, por lo que aumentar la potencia de la resistencia podría ocasionar que tuviéramos problemas en la instalación del hogar, ya que el consumo máximo suele estar limitado a un valor cercano a los 3000w por circuito. Una vez alcanzada la temperatura de maceración, la resistencia funcionará para compensar la pérdida térmica del equipo, para lo que tendrá que encenderse esporádicamente hasta volver a alcanzar la temperatura deseada. Este proceso requiere de mucha menos energía que el calentamiento inicial. Se estima que, para

mantener la temperatura de servicio, la resistencia tendrá que mantenerse encendida durante un 5% del tiempo de maceración. Se trata de un tiempo de referencia, ya el control de la temperatura mediante el encendido y apagado de la resistencia será controlado por el sistema Arduino.



Figura 20 Resistencia sumergible Instalación

5.3.4.1. Instalación

El calentador irá instalado en el fondo del HLT. El calentador dispone de un sistema de acople roscado, cuyo diámetro y rosca varía de uno a otro, siendo el más frecuente el de 1". Habrá que practicar un orificio en la parte inferior de la olla del tamaño de la rosca del calentador, asegurándonos de que quede lo más justo posible. El calentador se fijará a la pared de la olla mediante una tuerca de la rosca que requiera el calentador, y se utilizará una junta de teflón o silicona que impida la fuga de agua.

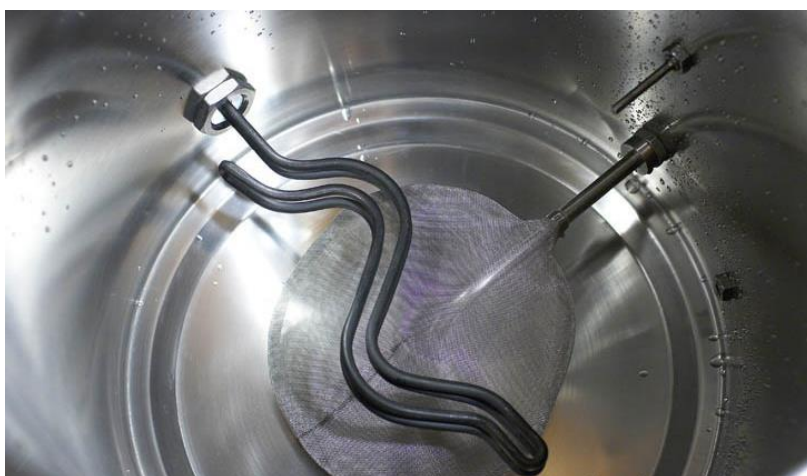


Figura 21 Calentador en el HLT

5.3.4.2. Guía de conexión

El calentador utilizado funciona a 220v y 3000w, mientras que el controlador Arduino admite el control de cargas de 3,5 y 5v, por lo que su conexión directa al controlador no es posible. Para adaptar la conexión del calentador será necesario utilizar un conmutador gestionado por el Arduino que realice las funciones de interruptor. Este componente lo veremos en el siguiente punto.

5.3.5. Conmutador del calentador

Dadas las características de alimentación del calentador visto en el punto anterior, no es posible realizar una conexión directa a la placa Arduino. Por lo tanto, es necesario utilizar un dispositivo que esté controlado por el Arduino y que actúe como interruptor para el calentador. Para realizar estas funciones se ha utilizado tradicionalmente un *relé*, pero la potencia necesaria por el calentador impide que se pueda utilizar en este caso. En su lugar utilizaremos un interruptor de estado sólido (SSR).



Figura 22 Interruptor SSR

5.3.5.1. Guía de conexión

Un SSR dispone de una entrada controlada por una tensión entre 3 y 32 voltios, lo que hace que sea compatible con Arduino. Por otra parte, dispone de la parte conmutada que soporta una carga de hasta 380 voltios y 40A, donde es posible conectar el calentador de agua del HLT. Cuando la entrada de control recibe tensión por la parte del Arduino, activa la conmutación de la carga, lo que hace que se encienda el calentador. Cuando deja de recibir tensión, interrumpe la alimentación de la resistencia.

En la siguiente figura se puede ver el esquema de conexión del calentador mediante un SSR.

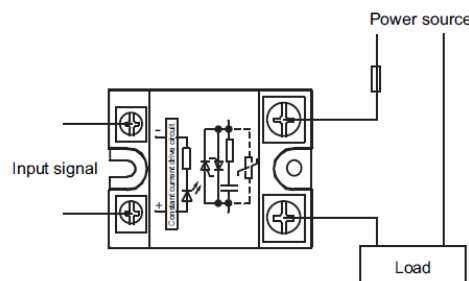


Figura 23 Esquema de conexión de un Calentador mediante SSR [4]

En la figura anterior, *input signal* representa la entrada de control procedente del Arduino. En la configuración de conexiones de pines de la placa establecimos que el SSR estaría controlado por la salida digital 11. *Load* representa la carga a controlar por el SSR, que en nuestro caso será el calentador localizado en el HLT, y *Power source* se refiere a la alimentación de 220 v de la carga, que irá conectada directamente a un enchufe.

6. Integración de todos los elementos

En los puntos anteriores hemos visto los elementos necesarios para adaptar el equipo cervecero para que pueda ser controlado por un Arduino. En esta sección vamos a ver cómo podemos integrar todos los componentes para que interactúen entre sí.

A continuación, se muestra la lista de componentes que necesitamos para montar el equipo. Se trata de una referencia del equipamiento necesario, pero que fácilmente es adaptable a las necesidades o posibilidades del montaje que cada uno pueda realizar:

- Arduino UNO
- Pantalla shield SKU DFR0009
- Caja estanca de conexión eléctrica de 216x116x90.
- Enchufe hembra estuco.
- 2 mini pulsadores
- 3 conectores Jack 3,5mm hembra para colocar en caja
- 3 sondas de temperatura DS18B20
- 3 conectores Jack 3.5 mm macho
- Interruptor 220v 6^a
- 2m de cable de 2.5mm² de sección
- Pasamuros de 8mm
- Enchufe estuco macho
- Transformador de corriente de 6v
- Fichas de empalme

En la siguiente figura podemos ver los principales componentes que formarán nuestro montaje y la disposición que tendrán en la caja.



Figura 24 Distribución de los componentes en la caja

6.1. Adaptación de la caja

El primer paso será marca los cortes a realizar en la caja que permitan realizar el montaje de componente. Habrá que medir el tamaño de cada componente para ajustar la dimensión de los agujeros a realizar.

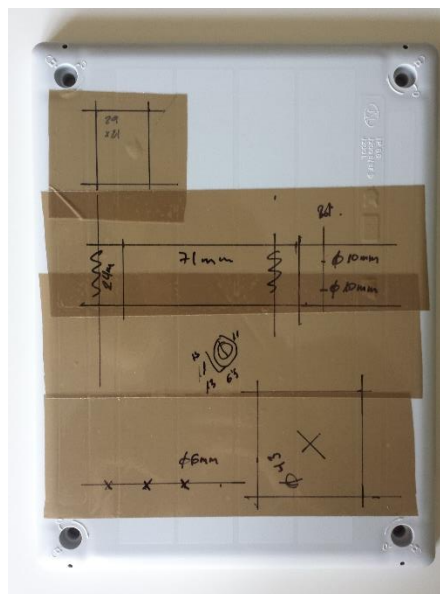


Figura 25 Marcas de agujereado de la caja

Para realizar los diferentes orificios utilizaremos las herramientas adecuadas. En mi caso he utilizado un taladro con brocas de diferente tamaño para realizar los orificios donde insertar los pulsadores y los conectores Jack para las sondas. Para los agujeros de mayor tamaño se puede

utilizar una herramienta tipo dremel, aunque en mi caso he preferido utilizar una sierra de marquetería. El repaso final y los pequeños ajustes se realizan mediante una lima para metales.

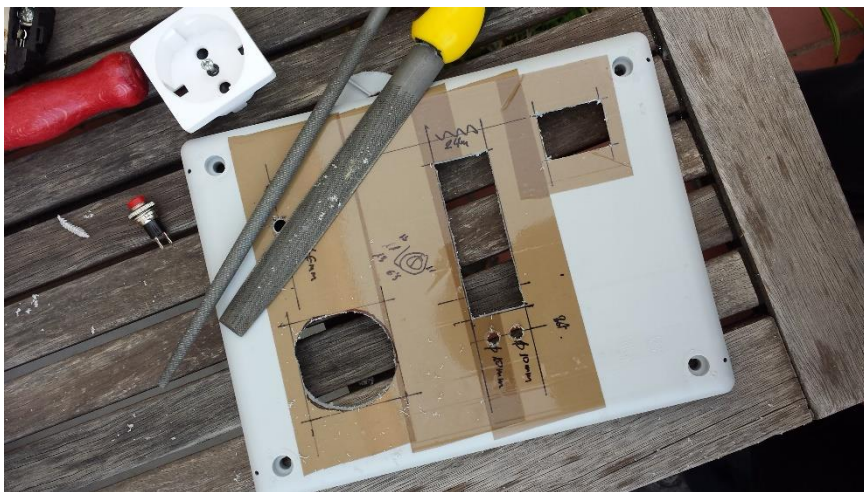


Figura 26 Trabajos de preparación de la caja

Una vez realizados los agujeros, ya es posible realizar la instalación de los componentes en la tapa de la caja. A la hora de colocar la pantalla LDC me he encontrado con el problema de que la placa tiene un potenciómetro que sobresale sobre el nivel de la superficie de la propia placa, lo que impide instalar la pantalla a ras de la tapa. La solución a este problema ha pasado por quitar el potenciómetro de su instalación original y realizar una nueva en horizontal, como se muestra en la foto. Para ello tendremos que hacer uso de un soldador de estaño de electrónica.

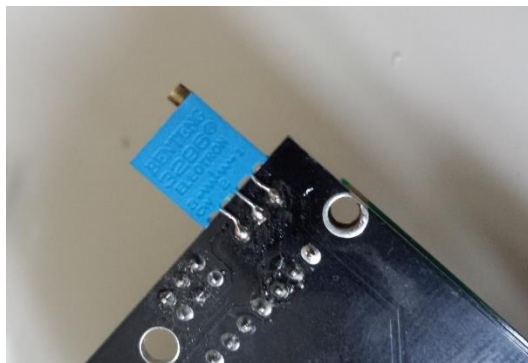


Figura 27 Cambio en la instalación del potenciómetro del LCD

En la siguiente figura se puede ver como quedan los componentes instalados en la caja a falta de realizar el cableado.

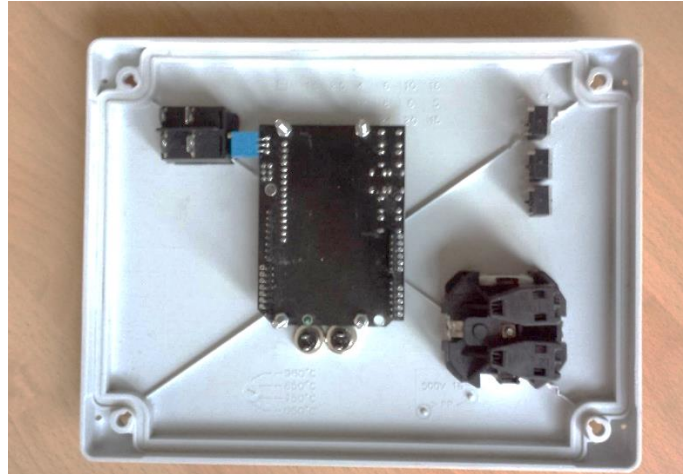


Figura 28 Componentes instalados en la placa

6.2. Instalación de baja tensión

Al acoplar la pantalla LCD en formato shield sobre la placa Arduino se ocupan los pines de conexión que necesitamos para realizar las conexiones de las sondas, pulsadores y el SSR. Como primera opción pensé en soldar los cables directamente a la placa del Arduino, pero finalmente opté por utilizar unos cables finos colocados en los pines antes de insertar la pantalla, de forma que al colocar la pantalla los cables quedarán fijados por la presión de los pines de la propia pantalla. En la siguiente figura se puede ver como he realizado el proceso.

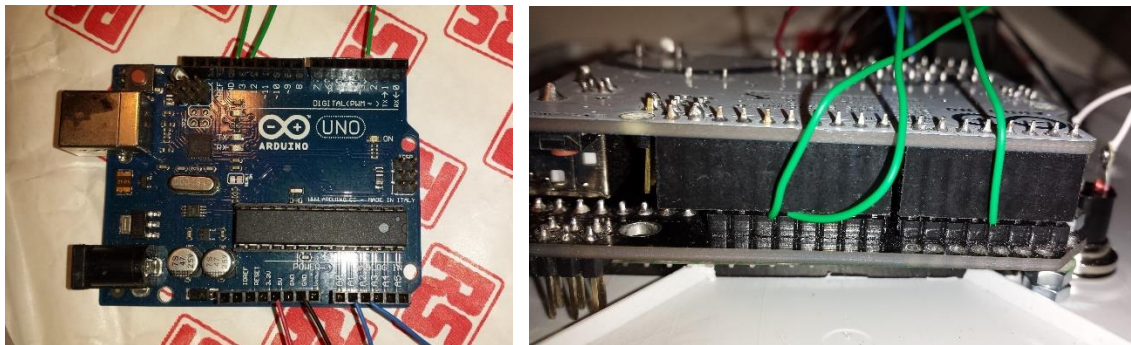


Figura 29 Conexión de los cables a la placa Arduino

En las siguientes fotos se puede apreciar la conexión de los cables a los jacks de conexión de las sondas y los pulsadores que permiten el control de la temperatura.

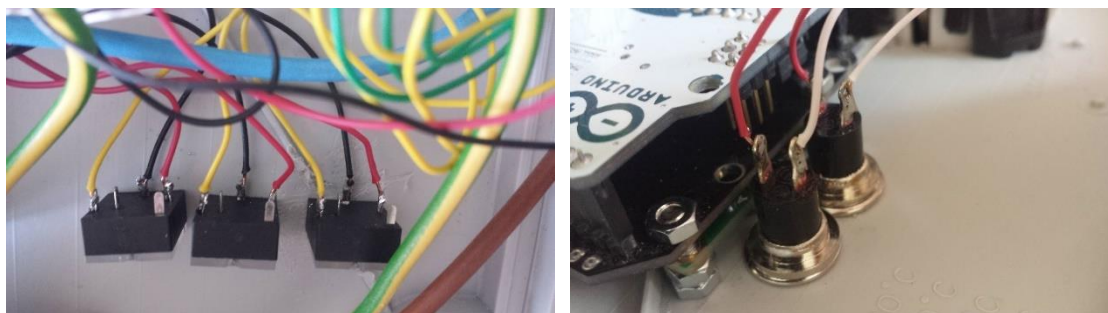


Figura 30 Conexión del cableado a los Jacks y a los pulsadores

Una vez que se ha completado la conexión del cableado de todos los componentes, queda por hacer la conexión entre los componentes con el Arduino. Las sondas de temperatura y los pulsadores requieren del uso de unas resistencias para adaptar la corriente e intensidad de funcionamiento. A continuación se muestra un esquema de conexión de los dispositivos, resistencias y Arduino:

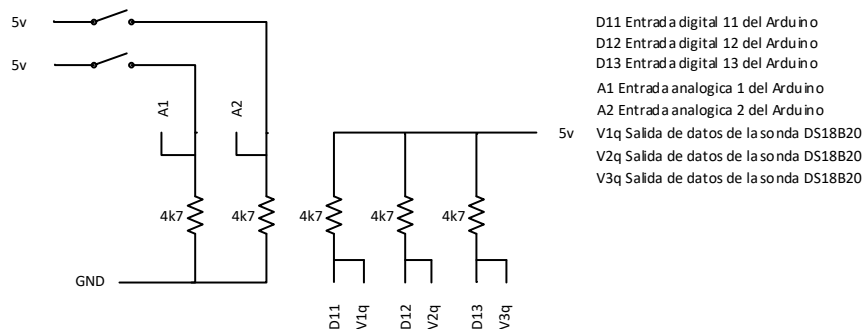


Figura 31 Esquema de conexión de baja tensión

Para implementar el esquema de interconexión vamos a utilizar una placa de prototipado que permita realizar la soldadura entre las resistencias y el cableado. El diseño de la placa sigue el esquema visto anteriormente. El resultado es la siguiente placa, donde pueden verse las resistencias que hemos tenido que utilizar y las conexiones del cableado:

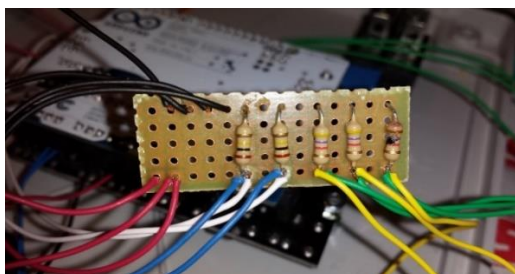


Figura 32 Instalación de las resistencias

6.3. *Instalación de alta tensión*

En este apartado vamos a ver cómo realizar la instalación de 220v. Esta tensión es necesaria para alimentar la resistencia de calentamiento del HLT, y aprovecharemos para alimentar también la placa Arduino.

El esquema que seguiremos será el que se muestra en la siguiente figura:

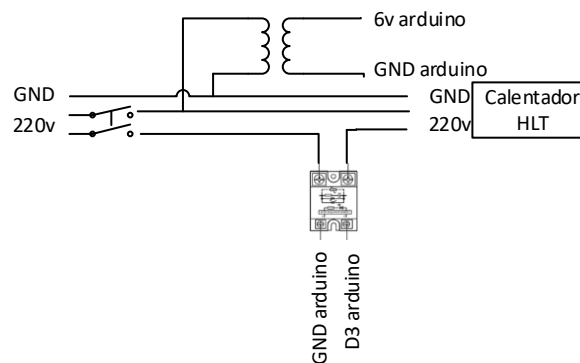


Figura 33 Esquema de conexión de alta tensión



Figura 34 Detalle de la instalación del interruptor

Una vez realizado el montaje completo, el resultado final será el siguiente:

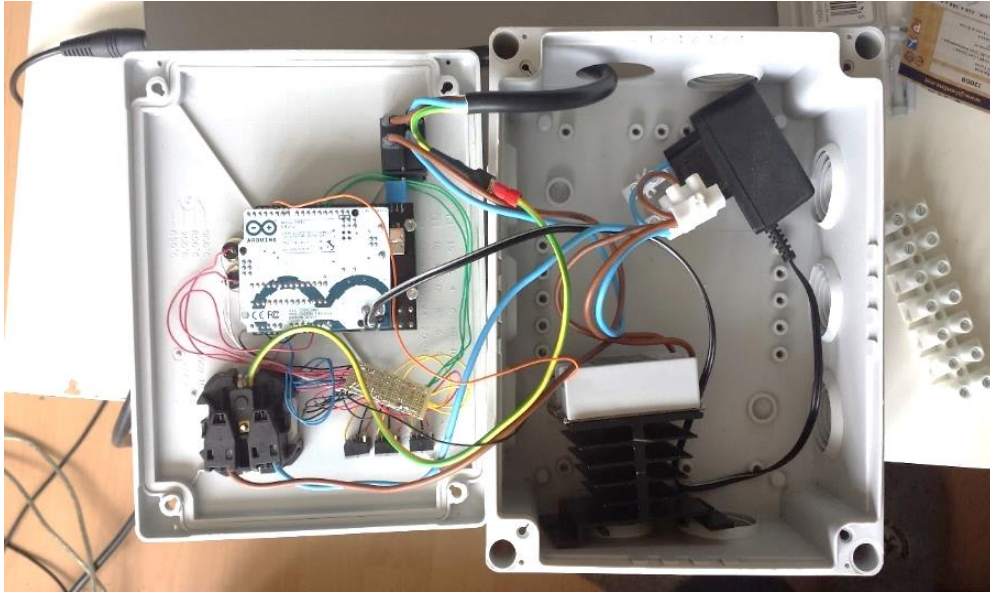


Figura 35 Instalación final de los componentes

6.4. Resultado final

Y este es el aspecto del controlador Arduino una vez finalizada la instalación de los componentes:



Figura 36 Controlador de maceración terminado

El controlador dispone de un interruptor general que realiza el corte de alimentación de todos los circuitos. En la parte central se encuentra la pantalla de monitorización, donde podemos ver la temperatura de funcionamiento del equipo cervecero, además de la temperatura de

maceración deseada. Junto a la pantalla hay dos pulsadores que se utilizan para fijar la temperatura de maceración.

En la parte inferior tenemos tres conectores donde se insertan las clavijas de las sondas de temperatura, y, junto a ellas, un enchufe donde conectar la resistencia del HLT, que será controlada por el Arduino para gestionar la temperatura de maceración.

7. Programación del controlador Arduino

La función del controlador Arduino será mantener el mosto a la temperatura de maceración. Para ello, hemos dotado al equipo de varios sensores de temperatura repartidos entre los puntos más significativos de los circuitos de recirculación de líquidos, de manera que sea posible determinar la temperatura a la que se está desarrollando la maceración de la malta. Además, hemos instalado un calentador localizado en el HLT, que puede ser activado por el Arduino para mantener la temperatura de maceración.

El proceso de configuración de la temperatura de funcionamiento no es especialmente complejo. El mosto que se encuentra en el MLT recircula por el serpentín colocado en el HLT, de forma que el mosto adquiere la temperatura del agua que se encuentra en el HLT antes de volver al MLT. La función del Arduino será controlar que el agua que se encuentra en el HLT esté a la temperatura necesaria para que la maceración se desarrolle a temperatura establecida. Para ello, el HLT dispone de una sonda de temperatura que se encuentra conectada al Arduino. Esta comprueba si la temperatura del mosto que retorna al MLT es menor que la temperatura de maceración, y de ser así, se activa el calentador del HLT hasta que se logra que el mosto alcance la temperatura de maceración.

Por otra parte, el sistema sirve de monitorización de la temperatura en varios puntos del sistema, donde se encuentran instaladas las sondas de temperatura. El controlador realiza constantemente la lectura de los valores de temperatura, y muestra esta información en la pantalla LCD para verificar si las temperaturas del proceso de maceración se encuentran dentro de los valores normales.

El siguiente diagrama de flujo de datos muestra el proceso que debe realizar el controlador Arduino.

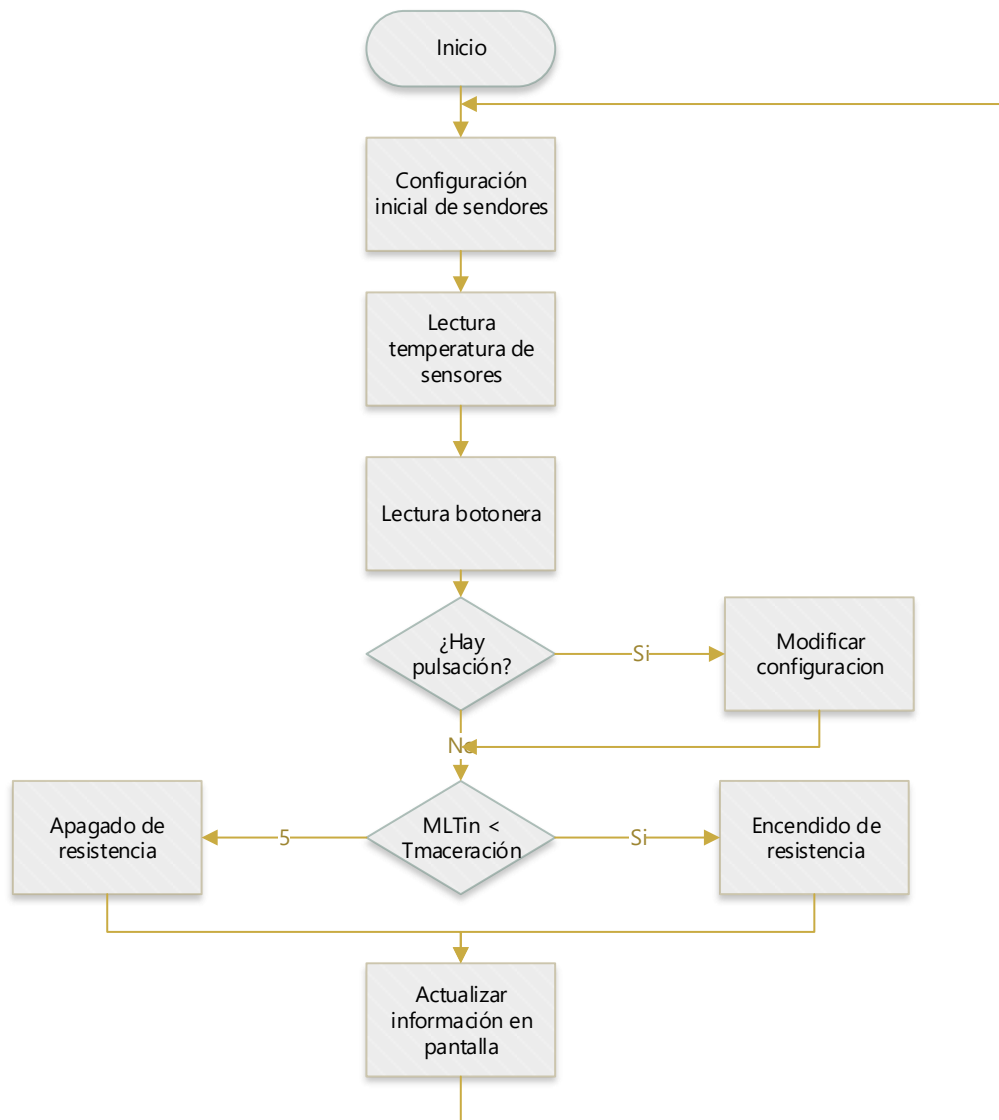


Figura 37 Diagrama de flujo del programa Arduino

Funcionalmente, podemos dividir el sistema en los siguientes subsistemas:

- Lectura de sondas.
- Control de la temperatura del HLT.
- Monitorización de la temperatura.
- Configuración del sistema.

7.1. Lectura de sondas

Realiza la lectura de las sondas instaladas en el sistema. El principal inconveniente que nos encontramos en este proceso es que la lectura de cada sonda lleva 750 milisegundos en completarse, por lo que no es recomendable realizar la lectura continua de todas las sondas en cada ciclo de ejecución del Arduino, ya que bloquearía el controlador realizando lecturas continuas

e impediría completar el resto de las operaciones que debe completar el sistema. Un caso especialmente sensible es el de la pulsación del teclado por parte del usuario. En este caso, la respuesta del sistema al usuario debe ser inmediata, pues tener un retraso cercano a un segundo daría la impresión de que ha fallado la operación.

La solución que se ha tomado para mejorar este proceso ha sido establecer temporizadores para la lectura de las sondas. Por ejemplo, se establece un temporizador de dos segundos, de forma que cada dos segundos se realiza la lectura de las sondas. El resto del tiempo comprendido en los dos segundos que no se utilizan para realizar la función de lectura de la sonda se emplea en atender otras operaciones del sistema, como interactuar con el usuario, actualizar la pantalla y controlar el calentador.

7.2. Control de la temperatura del HLT

En este proceso se gestiona el control de la temperatura del agua del HLT. El sistema debe de mantener el HLT a una temperatura tal que permita que el mosto de retorno al MLT, después de pasar por el serpentín, sea la temperatura establecida por el usuario como temperatura de maceración. El proceso consiste en comparar la temperatura actual del HLT con la temperatura fijada, y en caso de ser necesario, activar el calentador hasta que se alcance la temperatura deseada.

7.3. Monitorización

El usuario cuenta con una pantalla LCD como sistema de monitorización y auditoría, donde se muestra la temperatura de cada una de las sondas, la temperatura establecida como temperatura de maceración, la tendencia de las temperaturas (a mantenerse, subir o a bajar) y el estado del calentador del HLT.

7.4. Configuración.

Permite al usuario establecer la temperatura de maceración, para lo cual utilizará el teclado para establecer el valor numérico que indique los grados a los que se desea realizar la maceración.

7.5. Estableciendo la temperatura HLT

El proceso de funcionamiento del controlador de maceración que hemos visto hasta ahora es bastante sencillo; leer temperaturas, compararlas con un valor de referencia y, en caso de ser necesario, activar el calentador para elevar la temperatura. La complicación del proceso consiste en establecer el valor de la temperatura del HLT, que denominaremos T.

Entenderemos por T como la temperatura a la que se debe encontrar el agua del HLT de tal forma que permita que el mosto que circula a través del serpentín localizado en el HLT alcance como temperatura de salida la temperatura fijada para la maceración.

En un principio podríamos fijar T como la temperatura de maceración, pero, aunque en principio podría ser una buena aproximación, no se trata del valor ideal. Si suponemos que estamos con el equipo en funcionamiento con las temperaturas estabilizadas, la temperatura de salida del MLT será ligeramente inferior a la temperatura de maceración, digamos que por experiencia propia podríamos establecer que esta temperatura se encuentra entre uno y dos grados por debajo de la temperatura de maceración. Al circular por el HLT alcanzará prácticamente la misma temperatura a la que se encuentra el HLT, es decir, la temperatura de maceración. Sin embargo, en la práctica esto no es así, ya que por lo general se requiere que la temperatura del HLT sea del orden de medio grado o un grado superior a la temperatura de maceración para que la temperatura de retorno del mosto sea la establecida como temperatura de maceración.

Otro escenario donde se aprecia más claramente la dificultad de establecer el punto T sería en el momento de realizar la puesta en funcionamiento del sistema. Si partimos de mosto a una temperatura ambiente de 18 grados y lo hacemos circular por el serpentín del HLT a 66 grados, este no tiene capacidad para elevar la temperatura 48 grados. En estas circunstancias, la temperatura de salida del serpentín sería en torno a 35 grados. Necesitaríamos que el HLT se encontrase a una temperatura mucho mayor para que sea posible alcanzar la temperatura de maceración. Si fijamos la temperatura del HLT a la temperatura de maceración, a la larga el mosto alcanzara la temperatura deseada, pero esto podría requerir mucho tiempo.

Como hemos visto, la elección del valor de T viene determinado por la diferencia de temperatura que existe entre la temperatura del mosto y la temperatura de maceración. La solución a este problema pasa por diseñar un sistema que, en función de la temperatura de entrada en el serpentín y de la temperatura de salida del mismo, y en función de la distancia que

existe entre la salida y la temperatura deseada, establezca el valor de la temperatura del HLT. Sería un sistema que basaría su funcionamiento en el cálculo del error mediante realimentación del error de la salida.

Para solucionar este tipo de problemas existen los controladores PID. Podemos obtener la definición de un controlador PID de [5]: *“Un controlador PID (Proporcional Integrativo Derivativo) es un mecanismo de control genérico sobre una realimentación de bucle cerrado, ampliamente usado en la industria para el control de sistemas. El PID es un sistema al que le entra un error calculado a partir de la salida deseada menos la salida obtenida y su salida es utilizada como entrada en el sistema que queremos controlar. El controlador intenta minimizar el error ajustando la entrada del sistema.”*

Aplicando el funcionamiento a nuestro sistema, utilizaremos la temperatura de salida del serpentín como valor a ajustar por el PID, estableciendo como *setpoint* la temperatura de maceración. El valor que devolverá el PID será la temperatura de funcionamiento del HLT, que en nuestro caso hemos definido como T.

El PID para Arduino lo tenemos implementado con la librería *Arduino PID Library* by de *Brett Beauregard*, que será el que utilicemos en este sistema.

8. Licencia de uso

En este proyecto hemos trabajado en dos líneas diferentes, aunque relacionadas. Por una parte, hemos documentado el diseño y la adaptación de un equipo cervecero para su control por un Arduino, y por otra, hemos implementado la programación del Arduino para el control del equipo anteriormente diseñado. La distribución de estos productos se hará atendiendo a los criterios de software libre y copyleft.

En lo referente a esta documentación, pretendo que sea libre y que mantenga tal concepto una vez distribuida, por lo que he elegido hacerla pública bajo licencia la Licencia de Documentación Libre de GNU (GFDL)

Respecto al programa controlador del Arduino, este será también software libre. Siguiendo las recomendaciones de la Free Software Foundation, utilizaremos la licencia copyleft más fuerte que se adapte a las necesidades del proyecto.

Para determinar la licencia GNU a utilizar, antes tendremos que ver la compatibilidad de las librerías utilizadas por el sistema en términos de licencia. Para establecer un tipo de licencia particular, es necesario que todo el código que se utilice en el programa sea compatible con dicha licencia, por lo que previamente vamos a estudiar las librerías utilizadas.

Librerías propias de Arduino

Arduino distribuye sus librerías bajo licencia GNU LGPL. Respecto a GPL, la licencia LGPL, se trata de una licencia más permisiva, ya que permite enlazar con módulos privativos. Sin embargo, es compatible con GPLv2 y GPLv3.

Librería Wire

Se trata de una librería desarrollada por Nicholas Zambetti para comunicaciones TWI/I2C. Se distribuye bajo licencia GPL v2.

Librería LiquidCrystal.h

Propia de Arduino bajo licencia GNU LGPL.

Librería DallasTemperature

Librería distribuida bajo licencia GPL v2.1

Librería PID_v1

Librería creada por Brett Beauregard para implementar un controlador PID. Se distribuye bajo licencia GPL v3.

Librería OneWire.h

Librería OneWire que implementa la comunicación con dispositivos 1-wire de Maxim/Dallas. Esta librería se distribuye como software propietario bajo la licencia que se muestra a continuación.

Copyright (C) 1999-2004 Dallas Semiconductor Corporation, All Rights Reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

Como se puede ver en la licencia, a pesar de ser software propietario se puede utilizar sin ningún tipo de limitación, por lo que también es compatible con GPL v3.

Analizadas las licencias con los que se distribuyen las librerías utilizadas por el sistema, podemos comprobar que todas ellas son compatibles con GNU v3, por lo que finalmente adoptaremos esta licencia para distribuir el programa Macerino.

9. GNU Free Documentation License

Version 1.3, 3 November 2008

Copyright © 2000, 2001, 2002, 2007, 2008 Free Software Foundation, Inc.
<<http://fsf.org/>>

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

0. PREAMBLE

The purpose of this License is to make a manual, textbook, or other functional and useful document "free" in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or noncommercially. Secondly, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

This License is a kind of "copyleft", which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

We have designed this License in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

1. APPLICABILITY AND DEFINITIONS

This License applies to any manual or other work, in any medium, that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. Such a notice grants a world-wide, royalty-free license, unlimited in duration, to use that work under the conditions stated herein. The "Document", below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as "you". You accept the license if you copy, modify or distribute the work in a way requiring permission under copyright law.

A "Modified Version" of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A "Secondary Section" is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document's overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (Thus, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

The "Invariant Sections" are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License. If a section does not fit the above definition of Secondary then it is not allowed to be designated as Invariant. The Document may contain zero Invariant Sections. If the Document does not identify any Invariant Sections then there are none.

The "Cover Texts" are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License. A Front-Cover Text may be at most 5 words, and a Back-Cover Text may be at most 25 words.

A "Transparent" copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, that is suitable for revising the document straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety of formats suitable for input to text formatters. A copy made in an otherwise Transparent file format whose markup, or absence of markup, has been arranged to thwart or discourage subsequent modification by readers is not Transparent. An image format is not Transparent if used for any substantial amount of text. A copy that is not "Transparent" is called "Opaque".

Examples of suitable formats for Transparent copies include plain ASCII without markup, Texinfo input format, LaTeX input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML, PostScript or PDF designed for human modification. Examples of transparent image formats include PNG, XCF and JPG. Opaque formats include proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or processing tools are not generally available, and the machine-generated HTML, PostScript or PDF produced by some word processors for output purposes only.

The "Title Page" means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, "Title Page" means the text near the most prominent appearance of the work's title, preceding the beginning of the body of the text.

The "publisher" means any person or entity that distributes copies of the Document to the public.

A section "Entitled XYZ" means a named subunit of the Document whose title either is precisely XYZ or contains XYZ in parentheses following text that translates XYZ in another language. (Here XYZ stands for a specific section name mentioned below, such as "Acknowledgements", "Dedications", "Endorsements", or "History".) To "Preserve the Title" of such a section when you modify the Document means that it remains a section "Entitled XYZ" according to this definition.

The Document may include Warranty Disclaimers next to the notice which states that this License applies to the Document. These Warranty Disclaimers are considered to be included by reference in this License, but only as regards disclaiming warranties: any other implication that these Warranty Disclaimers may have is void and has no effect on the meaning of this License.

2. VERBATIM COPYING

You may copy and distribute the Document in any medium, either commercially or noncommercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

3. COPYING IN QUANTITY

If you publish printed copies (or copies in media that commonly have printed covers) of the Document, numbering more than 100, and the Document's license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also

clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a computer-network location from which the general network-using public has access to download using public-standard network protocols a complete Transparent copy of the Document, free of added material. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

4. MODIFICATIONS

You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

- A. Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.

- B. List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has fewer than five), unless they release you from this requirement.
- C. State on the Title page the name of the publisher of the Modified Version, as the publisher.
- D. Preserve all the copyright notices of the Document.
- E. Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.
- F. Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below.
- G. Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice.
- H. Include an unaltered copy of this License.
- I. Preserve the section Entitled "History", Preserve its Title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section Entitled "History" in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence.
- J. Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the "History" section. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission.
- K. For any section Entitled "Acknowledgements" or "Dedications", Preserve the Title of the section, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein.

- L. Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles.
- M. Delete any section Entitled "Endorsements". Such a section may not be included in the Modified Version.
- N. Do not retitle any existing section to be Entitled "Endorsements" or to conflict in title with any Invariant Section.
- O. Preserve any Warranty Disclaimers.

If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their titles to the list of Invariant Sections in the Modified Version's license notice. These titles must be distinct from any other section titles.

You may add a section Entitled "Endorsements", provided it contains nothing but endorsements of your Modified Version by various parties—for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

5. COMBINING DOCUMENTS

You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them

all as Invariant Sections of your combined work in its license notice, and that you preserve all their Warranty Disclaimers.

The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections Entitled "History" in the various original documents, forming one section Entitled "History"; likewise combine any sections Entitled "Acknowledgements", and any sections Entitled "Dedications". You must delete all sections Entitled "Endorsements".

6. COLLECTIONS OF DOCUMENTS

You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

7. AGGREGATION WITH INDEPENDENT WORKS

A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, is called an "aggregate" if the copyright resulting from the compilation is not used to limit the legal rights of the compilation's users beyond what the individual works permit. When the Document is included in an aggregate, this License does not apply to the other works in the aggregate which are not themselves derivative works of the Document.

If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one half of the entire aggregate, the Document's Cover Texts may be placed on covers that bracket the Document within the aggregate, or the electronic

equivalent of covers if the Document is in electronic form. Otherwise they must appear on printed covers that bracket the whole aggregate.

8. TRANSLATION

Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant Sections in addition to the original versions of these Invariant Sections. You may include a translation of this License, and all the license notices in the Document, and any Warranty Disclaimers, provided that you also include the original English version of this License and the original versions of those notices and disclaimers. In case of a disagreement between the translation and the original version of this License or a notice or disclaimer, the original version will prevail.

If a section in the Document is Entitled "Acknowledgements", "Dedications", or "History", the requirement (section 4) to Preserve its Title (section 1) will typically require changing the actual title.

9. TERMINATION

You may not copy, modify, sublicense, or distribute the Document except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, or distribute it is void, and will automatically terminate your rights under this License.

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated

and not permanently reinstated, receipt of a copy of some or all of the same material does not give you any rights to use it.

10. FUTURE REVISIONS OF THIS LICENSE

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See <http://www.gnu.org/copyleft/>.

Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License "or any later version" applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation. If the Document specifies that a proxy can decide which future versions of this License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Document.

11. RELICENSING

"Massive Multiauthor Collaboration Site" (or "MMC Site") means any World Wide Web server that publishes copyrightable works and also provides prominent facilities for anybody to edit those works. A public wiki that anybody can edit is an example of such a server. A "Massive Multiauthor Collaboration" (or "MMC") contained in the site means any set of copyrightable works thus published on the MMC site.

"CC-BY-SA" means the Creative Commons Attribution-Share Alike 3.0 license published by Creative Commons Corporation, a not-for-profit corporation with a principal place of business in San Francisco, California, as well as future copyleft versions of that license published by that same organization.

"Incorporate" means to publish or republish a Document, in whole or in part, as part of another Document.

An MMC is "eligible for relicensing" if it is licensed under this License, and if all works that were first published under this License somewhere other than this MMC, and subsequently

incorporated in whole or in part into the MMC, (1) had no cover texts or invariant sections, and (2) were thus incorporated prior to November 1, 2008.

The operator of an MMC Site may republish an MMC contained in the site under CC-BY-SA on the same site at any time before August 1, 2009, provided the MMC is eligible for relicensing.