

Universidad de Costa Rica

Facultad de Ingeniería

Escuela de Ingeniería Eléctrica

IE-0624 Laboratorio de Microcontroladores

II ciclo 2022

Laboratorio 3

Arduino: PID,GPIO,ADC y comunicaciones

Luis Javier Herrera B93840

Juan Ignacio Montealegre Salazar B95001

Profesor: Marco Villalta Fallas

Martes 11 de octubre del 2022

# Índice

Índice de figuras	III
Índice de tablas	IV
<b>1. Resumen</b>	<b>1</b>
<b>2. Nota teórica</b>	<b>1</b>
2.1. Microcontrolador ATmega328P . . . . .	1
2.2. Arduino Uno . . . . .	2
2.3. Termistor TTC05 . . . . .	3
2.4. Sensor de humedad DHT-22 . . . . .	4
2.5. Pantalla LCD PCD8544 . . . . .	4
2.6. Circuito generador de calor . . . . .	5
2.7. Componentes electrónicos adicionales . . . . .	6
2.7.1. LEDs de colores . . . . .	6
2.7.2. Capacitores . . . . .	7
2.8. Diseño de Circuito . . . . .	7
2.8.1. Lista de componentes . . . . .	9
<b>3. Desarrollo y Análisis de Resultados</b>	<b>10</b>
3.1. Desarrollo del circuito . . . . .	10
3.2. Desarrollo del programa . . . . .	11
3.3. Repositorio de Git . . . . .	13
3.4. Funcionalidad del circuito y del programa . . . . .	13
<b>4. Conclusiones y Recomendaciones</b>	<b>16</b>
<b>5. Referencias</b>	<b>17</b>
<b>6. Anexos</b>	<b>18</b>

## Índice de figuras

1.	Distribución de pines del ATmega328P [1]	2
2.	Diagrama de bloques del ATmega328P [1]	2
3.	Pines de salida del Arduino UNO [2]	3
4.	Topología del Arduino UNO [2]	3
5.	Sensor DHT-22 [3].	4
6.	Pantalla LCD PCD8544 [4].	5
7.	Circuito calentador propuesto con BJT	5
8.	Circuito calentador propuesto con BJT y capacitor	6
9.	Diagrama de pines de LED básico [5]	7
10.	Diseño de circuito final en SimulIDE	7
11.	Sensores y actuador en circuito propuesto como solución	10
12.	Visualización de resultados, señales de alerta y comunicación serial	11
13.	Diagrama de bloques de sistema realimentado	11
14.	Diagrama de flujo del proyecto	12
15.	Sistema por debajo de temperatura de operación	13
16.	Sistema por encima de temperatura de operación	14
17.	Sistema funcionando con control de 20°C	14
18.	Sistema funcionando con control de 42°C	15
19.	Extracto de archivo .csv generafo	15

Índice de tablas

1. Lista de componente utilizados en el laboratorio. . . . . 9

# 1. Resumen

En esta práctica de laboratorio se desarrolla una incubadora de huevos automática utilizando una placa integrada Arduino UNO, la cual posee un microcontrolador ATmega328P. El sistema mide la temperatura a la que se encuentra utilizando un termistor y se buscará controlarla de tal forma que se mantenga en un rango de operación es de  $[30,42]^{\circ}\text{C}$ . El valor de operación deseado se selecciona por medio de la operación de un potenciómetro. Para controlar la temperatura, se utilizará un circuito con una resistencia alimentado por 12 V cuya potencia variará de 0 a 5 W según una señal de control emitida por el Arduino. Dicha señal de control funciona bajo los principios de un control PID, para lo cual se utiliza la librería PID. También se mide la humedad, para luego desplegar el valor de tanto esta como de la temperatura en una pantalla LCD PCD8544. Finalmente, se incluye una función de comunicación por medio del protocolo USART con una PC que se habilita con un switch, de tal forma que se puedan enviar los datos de temperatura y humedad para guardar un registro de estos datos en un archivo csv. De esta forma, se logra implementar exitosamente el uso de un microcontrolador como un controlador PID de un proceso, así como establecer una comunicación entre este y otro dispositivo, siendo en este caso la computadora en uso.

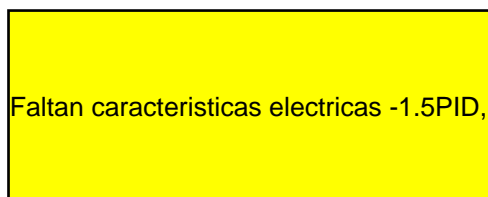
**Palabras clave:** *Arduino, comunicación, control, temperatura, microcontrolador, PID, SimulIDE, temperatura, USART*

## 2. Nota teórica

En esta sección se describe el microcontrolador utilizado en esta ocasión, así como cada periférico utilizado (registros e instrucciones/funciones), componente electrónico complementario y el diseño del circuito final. El proceso de pruebas realizadas para llegar al circuito final se abarca en la sección "Desarrollo y Análisis de Resultados".

### 2.1. Microcontrolador ATmega328P

Para la práctica de laboratorio 3 se trabajó con el microcontrolador ATmega328P, el cual un microcontrolador CMOS de 8 bits basado en la arquitectura RISC AVR y es desarrollado por la compañía Atmel. Posee un total de 23 GPIOs, donde 8 corresponden a canales con salida PWM que permite tener una salida analógica. Trabaja con 32 registros de propósito general y además posee 3 contadores (dos de 8 bits y uno de 16 bits). Además, opera entre los 1.8 a 5 V y la salida de sus pines llega hasta los 5 V (cuenta con un pin de 3 V). En la figura 1 se observa la distribución de los pines, mientras que en la figura 2 se observa su respectivo diagrama de bloques estructural.



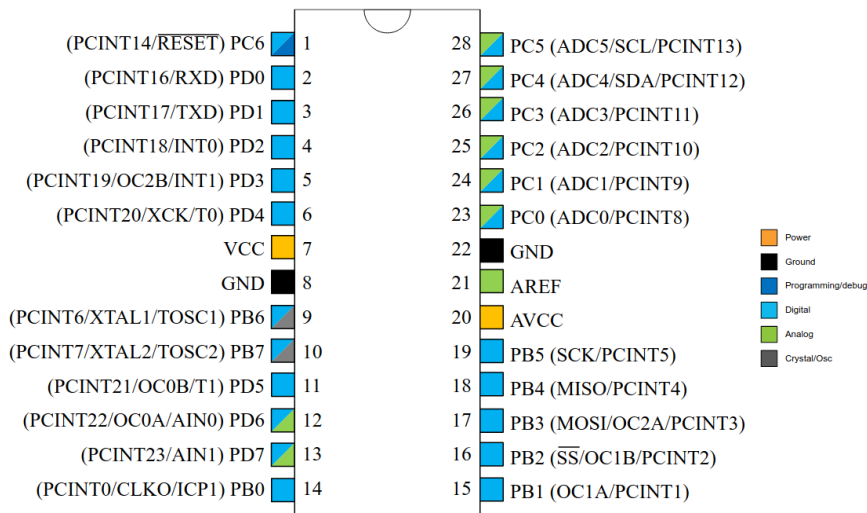


Figura 1: Distribución de pines del ATmega328P [1]

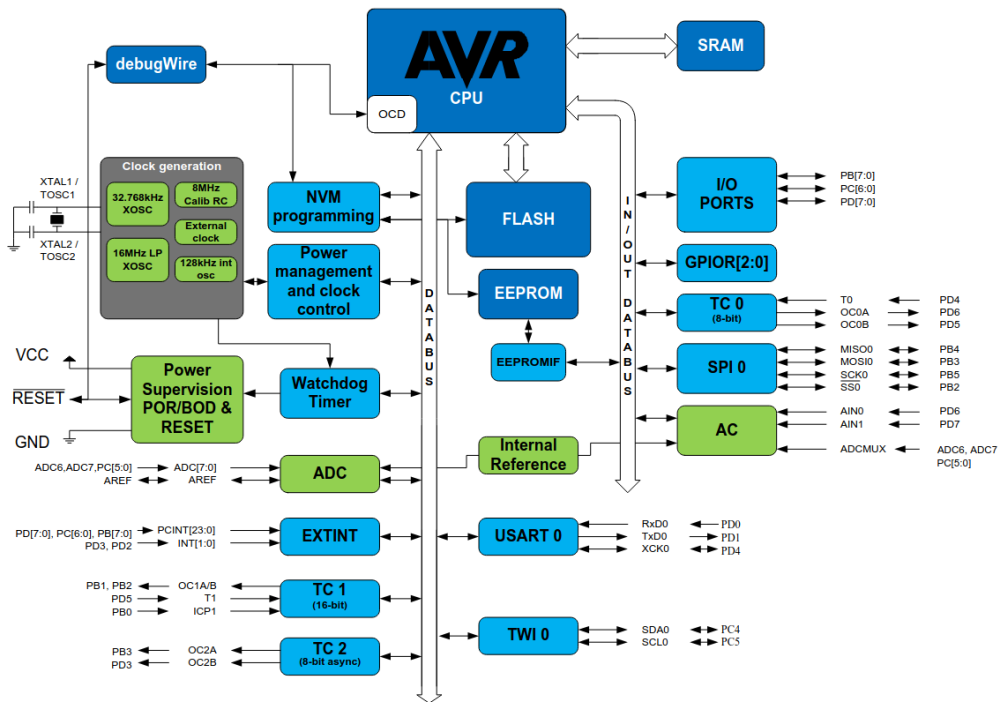


Figura 2: Diagrama de bloques del ATmega328P [1]

## 2.2. Arduino Uno

Corresponde a una placa con un microcontrolador ATmega328P, así como un ATmega16, y diversos periféricos, como puertos de alimentación y para comunicarse con otros dispositivos. Es de uso libre y se caracteriza por su gran accesibilidad en cuestión a precio y disponibilidad, así como la facilidad que hay en cuanto a su uso, ya que cuenta con una interfaz de desarrollo propia. En las siguientes figuras se muestran los pines de salida de la placa Arduino Uno, así como la topología de la placa.

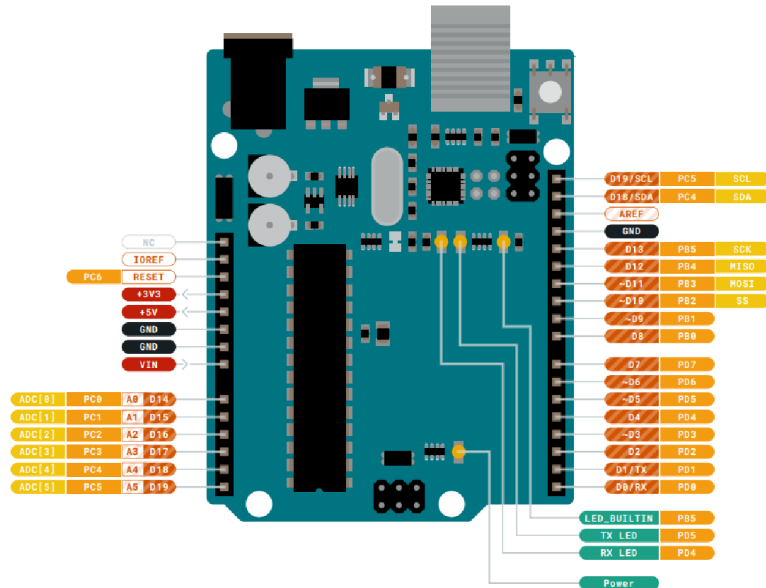
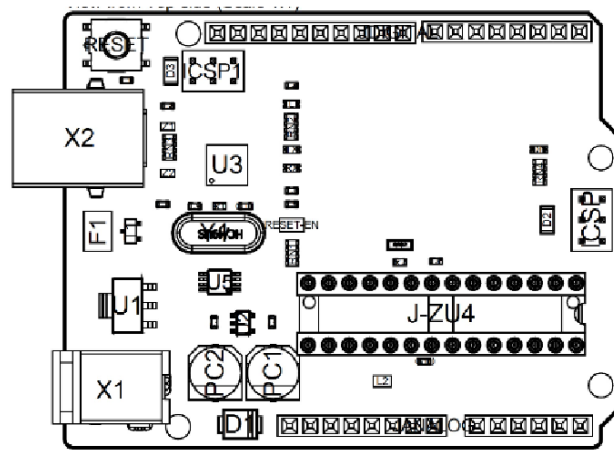


Figura 3: Pines de salida del Arduino UNO [2]



Board topology

Ref.	Description	Ref.	Description
X1	Power jack 2.1x5.5mm	U1	SPX1117M3-L-5 Regulator
X2	USB B Connector	U3	ATMEGA16U2 Module
PC1	EEE-1EA470WP 25V SMD Capacitor	U5	LMV358LIST-A.9 IC
PC2	EEE-1EA470WP 25V SMD Capacitor	F1	Chip Capacitor, High Density
D1	CGRA4007-G Rectifier	ICSP	Pin header connector (through hole 6)
J-ZU4	ATMEGA328P Module	ICSP1	Pin header connector (through hole 6)
Y1	ECS-160-20-4X-DU Oscillator		

Figura 4: Topología del Arduino UNO [2]

## 2.3. Termistor TTC05

Los termistores son dispositivos electrónicos que varían su resistencia en función a la temperatura en que se encuentran. Para este caso se decidió utilizar el termistor TTC05. Este tiene un tamaño de 5 mm, cuenta con un rango de temperaturas que va desde los -30 °C hasta los 125 °C. La versión seleccionada posee una resistencia de 10 kΩ a una temperatura de 25 °C. Puede ser utilizado para aplicaciones domésticas, aplicaciones automotrices, computadores y sensores digitales [6].

## 2.4. Sensor de humedad DHT-22

El DHT-22 es un sensor de humedad de bajo costo que está calibrado y no requiere componentes adicionales para obtener una medición de la humedad relativa. Además, también puede medir la temperatura, por lo que si se llega a implementar el proyecto de laboratorio con este sensor sería suficiente y no se requeriría adquirir el termistor previamente explicado. Entre las características de este sensor se encuentran el hecho que requiere una alimentación de 3.3 a 6 V y opera desde los -40 °C hasta los 80 °C.

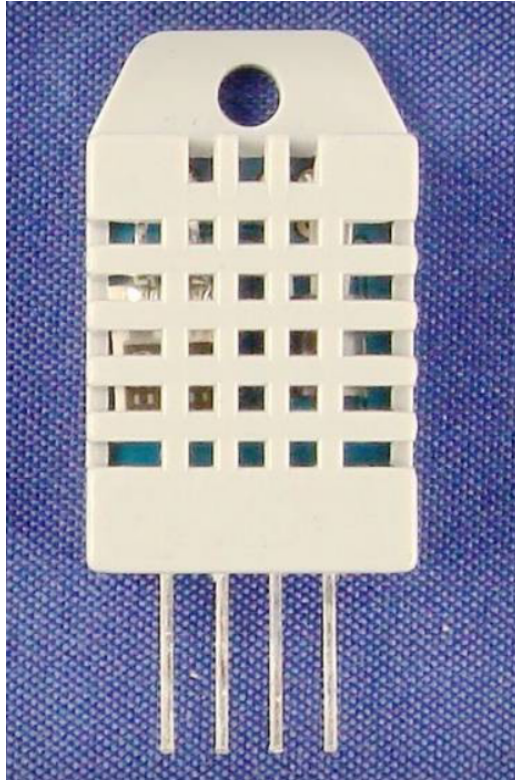


Figura 5: Sensor DHT-22 [3].

## 2.5. Pantalla LCD PCD8544

La pantalla LCD PCD8544 es un dispositivo monocromo fabricado por Nokia originalmente para sus celulares [4]. Es de un bajo consumo de potencia y se programa utilizando la librería de Arduino *Adafruit*. Cuenta con 5 pines con los cuales se conectan a los pines de salida digitales del Arduino para así enviar el formato y la información a mostrar en la pantalla.



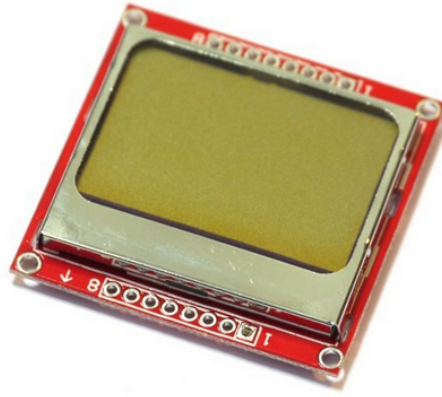


Figura 6: Pantalla LCD PCD8544 [4].

## 2.6. Circuito generador de calor

Se busca diseñar un circuito que alimente una resistencia con una tensión variable de 0 a 12 V, de tal forma que la potencia máxima que pueda alcanzar sea de 5 W. Debido a que la salida máxima del arduino es de 5 V, se propone utilizar un transistor BJT donde su base esté controlada por una señal variable del arduino. Se utiliza específicamente el transistor P2N2222A, el cual posee una ganancia de  $\beta = 100$ . En la siguiente figura se presenta la propuesta del circuito calentador.

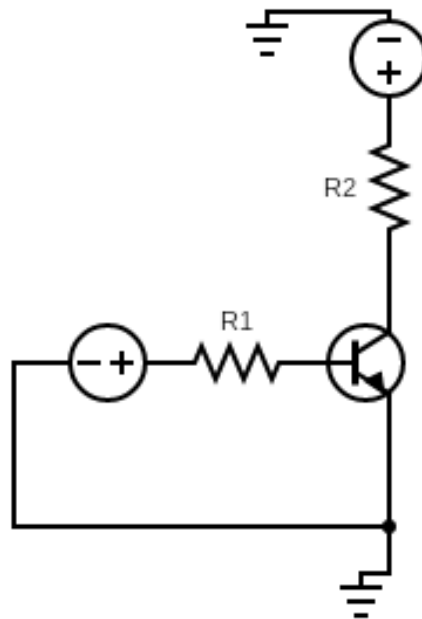


Figura 7: Circuito calentador propuesto con BJT

Para el cálculo de la resistencia  $R_2$ , se parte de que la potencia máxima es de 5 W. Teniendo presente que:

$$P = \frac{V^2}{R}$$

Es posible determinar un valor para  $R_2$ :

$$R_2 = \frac{(12\text{ V})^2}{5\text{ W}} = 28,8\ \Omega$$

El valor comercial disponible más cercano al calculado corresponde a  $27\ \Omega$ , por lo que se selecciona dicho valor para  $R_2$ . Para calcular  $R_1$ , primero se calcula la corriente del colector en el circuito.

$$I_C = \frac{12\text{ V}}{27\ \Omega} = 444,4444\text{ mA}$$

Ahora, se utiliza la siguiente relación aproximada entre la corriente de la base y la corriente de colector para un transistor BJT.

$$I_C = I_B * \beta \rightarrow I_B = \frac{I_C}{\beta} = \frac{444,4444\text{ mA}}{100} = 4,4444\text{ mA}$$

Sabiendo que la tensión de salida máxima del Arduino es de 5 V, se procede a calcular  $R_1$ :

$$R_1 = \frac{V}{I} = \frac{5\text{ V}}{4,4444\text{ mA}} \approx 1,12\text{ k}\Omega$$

Como detalle final, se agrega un capacitor de considerable capacitancia (10 mF) en paralelo a la resistencia del colector debido a que la señal es de PWM, por lo que para poder medir un valor de tensión se utiliza un filtro. Esto nos da como resultado el siguiente circuito.

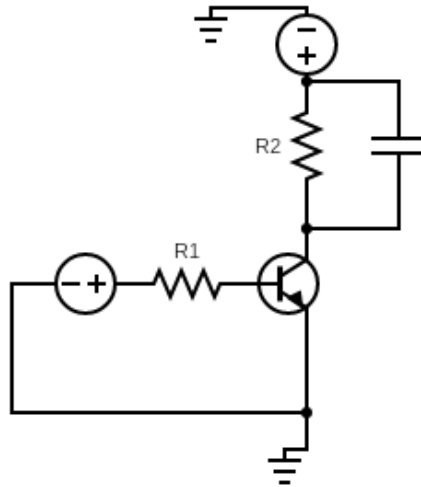


Figura 8: Circuito calentador propuesto con BJT y capacitor

## 2.7. Componentes electrónicos adicionales

### 2.7.1. LEDs de colores

Para este laboratorio se utilizaron diodos de tipo LED de colores rojo y azul para dar la señal de advertencia cuando se llegara a ciertas temperaturas medidas por el sensor. El funcionamiento de estos es bastante simple, se cuenta con dos terminales: el cátodo y el ánodo. Si se crea una diferencia de voltaje en el cual el cátodo se encuentra a un mayor voltaje que el ánodo, el diodo va a emitir luz. Caso contrario se va a encontrar apagado. Dependiendo del color que se quiera emitir estos van a tener diferentes rangos de operación en el voltaje.

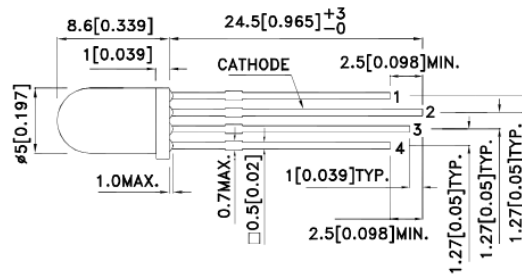


Figura 9: Diagrama de pines de LED básico [5]

### 2.7.2. Capacitores

Los capacitores son dispositivos electrónicos bastante comunes, estos se utilizan para almacenar energía dentro de un campo eléctrico interno. Es un componente de tipo pasivo y su estructura se compone de dos placas conductoras separadas por un material dieléctrico.

## 2.8. Diseño de Circuito

Luego de realizar diversas pruebas y cambios en el diseño, se llega al diseño del circuito que se muestra en la siguiente figura.

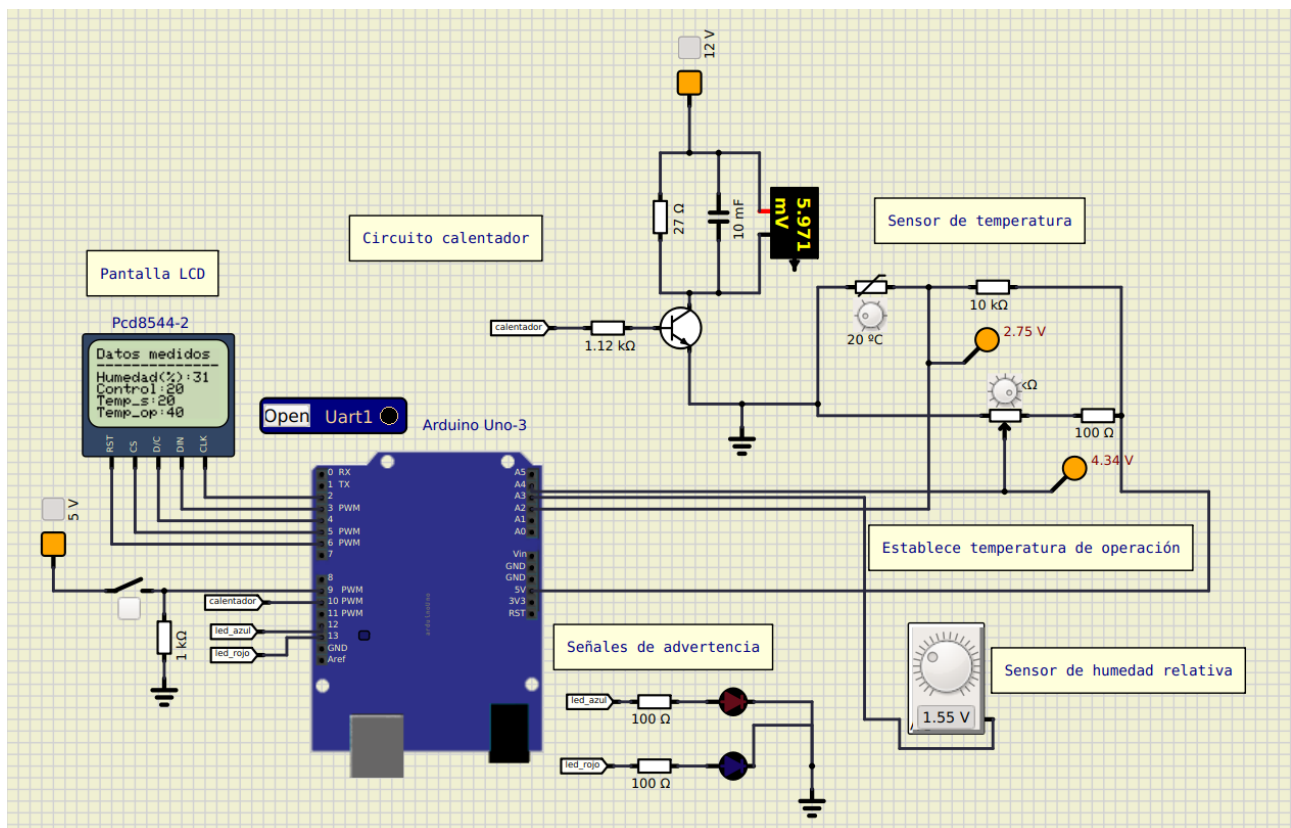


Figura 10: Diseño de circuito final en SimulIDE

Para lograr que la temperatura mostrada en la pantalla LCD fuera la misma que la medida por el termistor se utilizó la ecuación de Steinhart-Hart [7]. Primeramente, se debe realizar una división de tensión en el nodo que se conecta al Arduino y recolectar tres valores de resistencia.

Uno a la menor temperatura (0 °C), otro a una temperatura intermedio (62 °C) y de último, uno a la mayor temperatura soportada (125 °C).

$$R_0 = \frac{3,71}{\frac{(5-3,71)}{10000}} = 28759,68 \, \Omega \quad (1)$$

$$R_{62} = \frac{1,09}{\frac{(5-1,09)}{10000}} = 2787,72 \, \Omega \quad (2)$$

$$R_{125} = \frac{0,26}{\frac{(0,26-1,09)}{10000}} = 518,52 \, \Omega \quad (3)$$

Posteriormente se pasa a calcular una serie de parámetros útiles:

$$\begin{aligned} L_1 &= \ln R_0 = 10,267 \\ L_2 &= \ln R_{62} = 7,933 \\ L_3 &= \ln R_{125} = 6,251 \\ Y_1 &= \frac{1}{T_1} = 3,661E - 3 \\ Y_2 &= \frac{1}{T_2} = 2,984E - 3 \\ Y_3 &= \frac{1}{T_3} = 2,512E - 3 \\ \gamma_2 &= \frac{Y_2 - Y_1}{L_2 - L_1} = 2,896E - 4 \\ \gamma_3 &= \frac{Y_3 - Y_1}{L_3 - L_1} = 2,858E - 4 \end{aligned} \quad (4)$$

Con lo anterior calculado se procede a calcular los factores  $A$ ,  $B$  y  $C$  que se sustituyen directamente en la ecuación de Steinhart-Hart:

$$A = Y_1 - (B + L_1^2 C) L_1 = 8,2485E - 4 \quad (5)$$

$$B = \gamma_2 - C (L_1^2 + L_1 L_2 + L_2^2) = 2,665E - 4 \quad (6)$$

$$C = \left( \frac{\gamma_3 - \gamma_2}{L_3 - L_2} \right) (L_1 + L_2 + L_3)^{-1} = 9,239E - 8 \quad (7)$$

Ya con todo listo solo basta sustituir el valor de resistencia a la temperatura que se quiere encontrar junto con los parámetros calculados en la ecuación 8 para encontrar la temperatura de operación del termistor.

$$\frac{1}{T} = A + B \ln R + C (\ln R)^3 \quad (8)$$

Cabe resalta que el valor de temperatura  $T$  se encuentra en Kelvin por lo cual se debe hacer la respectiva sustitución a las unidades que se quiera mostrar.

**Nota importante:** En caso de simular el circuito, para que funcione tal y como se diseñó es necesario desconectar ambos cables del capacitor y volver a conectarlos. Se desconoce el porqué esto es necesario cada vez que se cierra y se vuelve a abrir el circuito en SimulIDE.

### 2.8.1. Lista de componentes

La lista de componentes utilizados en el experimento se muestra en la Tabla 1.

Componente	Tipo	Precio	Cantidad
Pantalla LCD	PCD8544	\$ 8.95	1
Termistor	TTC05	\$ 0.75	-
Microcontrolador	Arduino UNO	\$ 28.50	1
Led	Azul	\$ 0.03	1
Led	Rojo	\$ 0.03	1
Sensor de humedad	DHT22	\$ 9.89	1
Interruptor simple	-	\$ 0.98	1
Capacitor	10 mF	\$ 0.1	1
Resistor	100 $\Omega$	\$0.07	3
Resistor	10 K $\Omega$	\$ 0.02	2
Resistor	1.12 K $\Omega$	\$ 0.02	1
Transistor BJT	2N3906	\$ 0.02	1
Total	-	\$49.24	16

Tabla 1: Lista de componente utilizados en el laboratorio.

### 3. Desarrollo y Análisis de Resultados

En esta sección se comenta de forma detallada el desarrollo del proyecto. Primero se explica el desarrollo del circuito como tal, y luego el diseño del programa a partir del programa deseado.

#### 3.1. Desarrollo del circuito

El circuito propuesto para desarrollo del presente trabajo de laboratorio se puede dividir en dos secciones principales. Una de estas se compone principalmente por sensores y actuadores. Esto con el fin de tomar datos de la incubadora y con base a estos proceder con la acción de control indicada. El circuito se puede apreciar en la Figura 11.

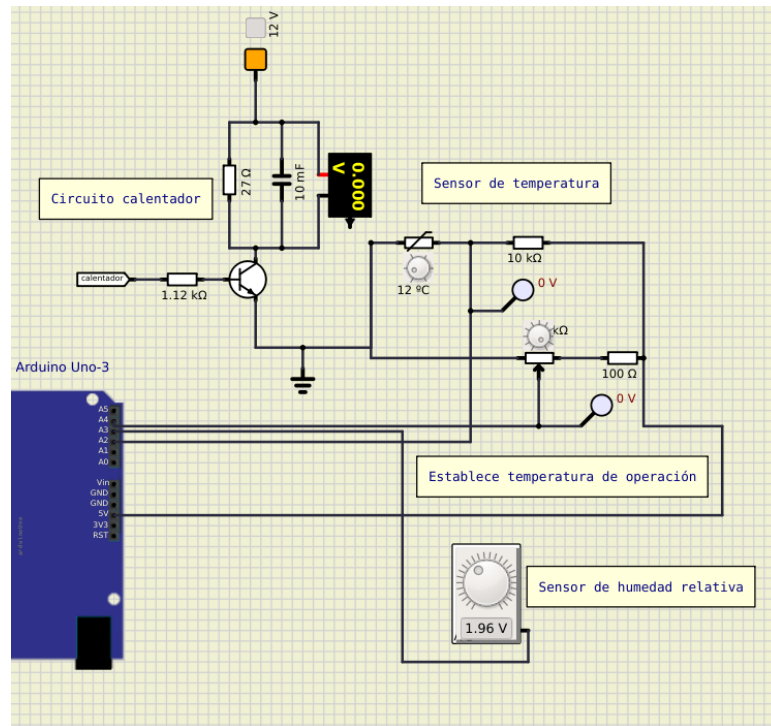


Figura 11: Sensores y actuador en circuito propuesto como solución

Según lo indicado en el enunciado del laboratorio, el sensor de humedad relativa se modela con una fuente de voltaje variable de 0 V a 5 V. El sensor de temperatura se modela con un termistor que puede operar de los 0  $^{\circ}\text{C}$  y los 125  $^{\circ}\text{C}$ , el cual se coloca en serie con una resistencia de 10 k $\Omega$ . Por otro lado, para establecer el punto de operación se utiliza un potenciómetro en serie con una resistencia de 100  $\Omega$ , de tal forma que se varía la tensión desde los 0 V hasta los 5 V, valores que luego se limitan al rango de temperaturas de operación, el cual va desde los 30  $^{\circ}\text{C}$  hasta los 42  $^{\circ}\text{C}$ . Para la parte del calentador se utilizó una aproximación con un resistor en el cual al variar el voltaje entregado en la base de transistor BJT con una salida PWM del Arduino, se logra variar el voltaje en el resistor. Con esto se puede simular el efecto de estar calentándose cuando así se requiera en la acción de control.

La otra parte del circuito propuesto se caracteriza por la visualización de los datos medidos y señales de indicación para el usuario. Esta sección se puede observar en la Figura 12. Se incluye una pantalla de tipo LCD la cual muestra: la señal de control, la temperatura medida, la temperatura de operación y la humedad relativa en la incubadora. Por otro lado, se añade al diseño la utilización de dos diodos LED como forma de indicar al usuario el momento en que la temperatura medida se encuentra por debajo de los 30  $^{\circ}\text{C}$  o por arriba de los 42  $^{\circ}\text{C}$ . Además,

al pin 9 del Arduino se conectó un interruptor como forma de habilitar y/o deshabilitar la comunicación serial para el envío de los datos medidos a la computadora.

Rebotes -1.5

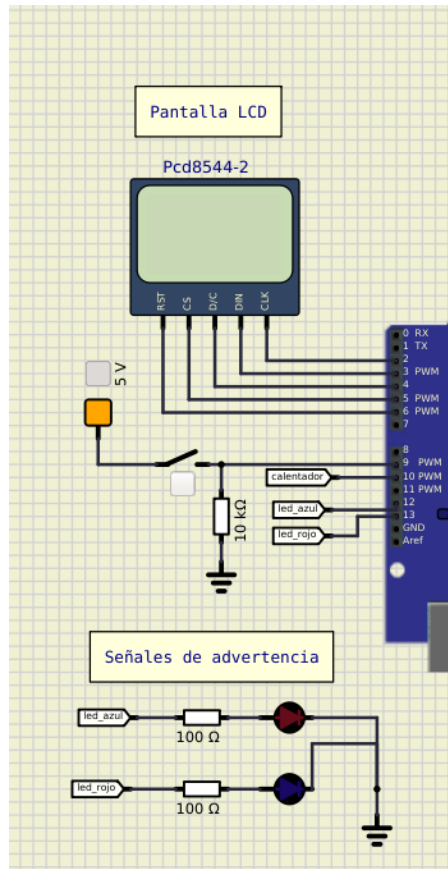


Figura 12: Visualización de resultados, señales de alerta y comunicación serial

### 3.2. Desarrollo del programa

Antes de comenzar con el desarrollo del programa, el cual se elaborará bajo el lenguaje de programación C utilizando el IDE de Arduino, es importante realizar un diagrama del proceso que se desea realizar. En esta ocasión se realizan dos diagramas, donde el primero corresponde a un diagrama de bloques donde se observa el funcionamiento del controlador PID y el siguiente donde se observa el diagrama de flujo del programa a partir de la señal de control emitida.

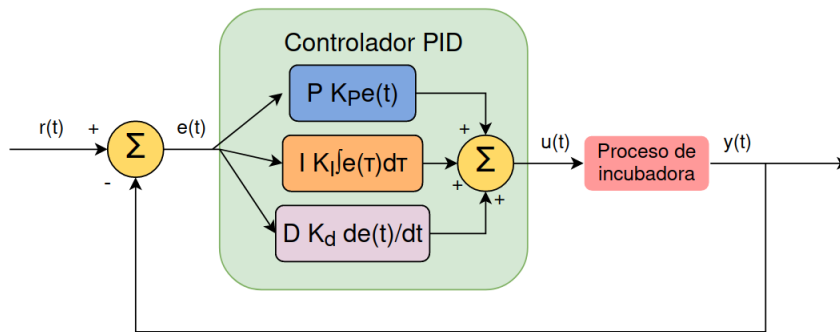


Figura 13: Diagrama de bloques de sistema realimentado

La señal  $r(t)$  corresponde a la referencia que en este caso corresponde al punto de operación

deseado indicado por el potenciómetro. El error  $e(t)$  se obtiene al restarle a la salida el la referencia, ya que se quiere llegar a esta. En esta caso, se utiliza una librería para simular el comportamiento de un controlador PID, donde los parámetros  $K_i$  y  $K_d$  se ajustaron como 0, mientras que al parámetro  $K_p$  se le asignó un valor de 3.3. Con estos valor la señal de control sigue perfectamente a la referencia, obteniendo de esa forma el control deseado.

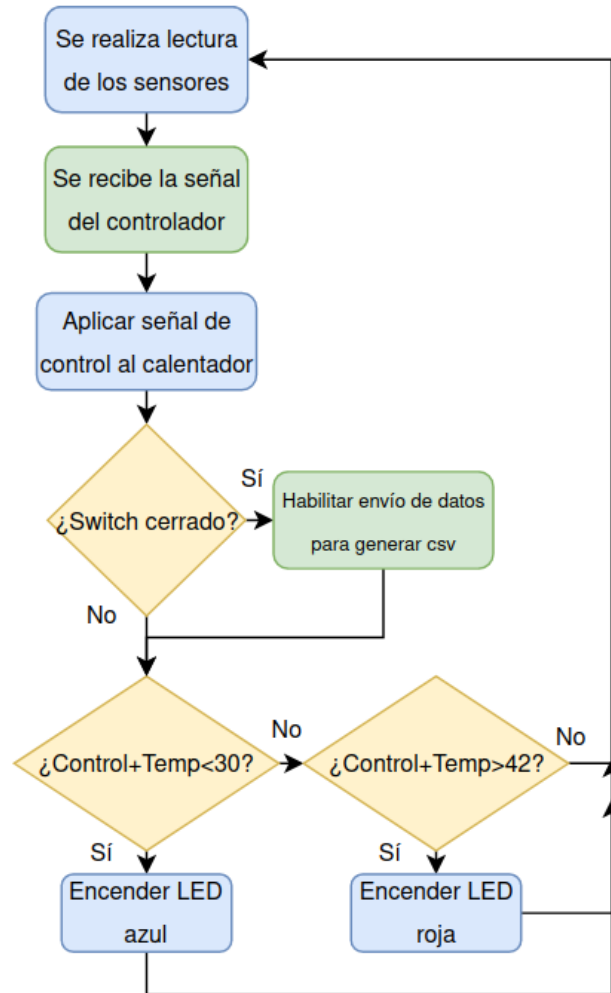


Figura 14: Diagrama de flujo del proyecto

El primer paso del programa es realizar una lectura de todos los sensores para poder de esta forma generar una salida de control. Una vez que se tiene la señal de control, esta se envía al circuito calentador para generar el calor apropiado. El circuito cuenta con un switch que habilita la comunicación del Arduino con una computadora para el envío de datos, por lo que observa el estado de este interruptor para ver si se envían datos. En caso de que se realice el envío de datos, un programa en Python se encargará de recibir los datos y guardarlos en un archivo .csv. Además, el programa debe advertir si la incubadora se encuentra a temperaturas fuera del rango establecido, por lo que si se encuentra a menos de 30 grados se enciende una LED azul, mientras que si se superan los 42 grados se ilumina una LED roja. La temperatura del controlador se estima que será la temperatura sensada, que en el caso del simulador es fija, más la temperatura que aporta la señal de control.



### 3.3. Repositorio de Git

Este proyecto se trabajó en un repositorio de Git para poder tener un control de las diversas versiones de todos los archivos que componen el proyecto. Se utilizó la plataforma de GitHub y el repositorio se encuentran en este link. Este laboratorio se encuentra en el directorio “Laboratorio 3”.

### 3.4. Funcionalidad del circuito y del programa

A continuación se adjuntan diversas capturas donde se puede comprobar el funcionamiento del circuito, ya que se observa el control adecuado y el cambio de la tensión en el calentador, así como el accionamiento de las luces LEDs según corresponde y la lectura de la humedad. Todos estos datos se despliegan en la pantalla LCD. Además, se muestra un extracto del archivo .csv que genera el programa en Python cuando se habilita la comunicación entre el simulador y la computadora.

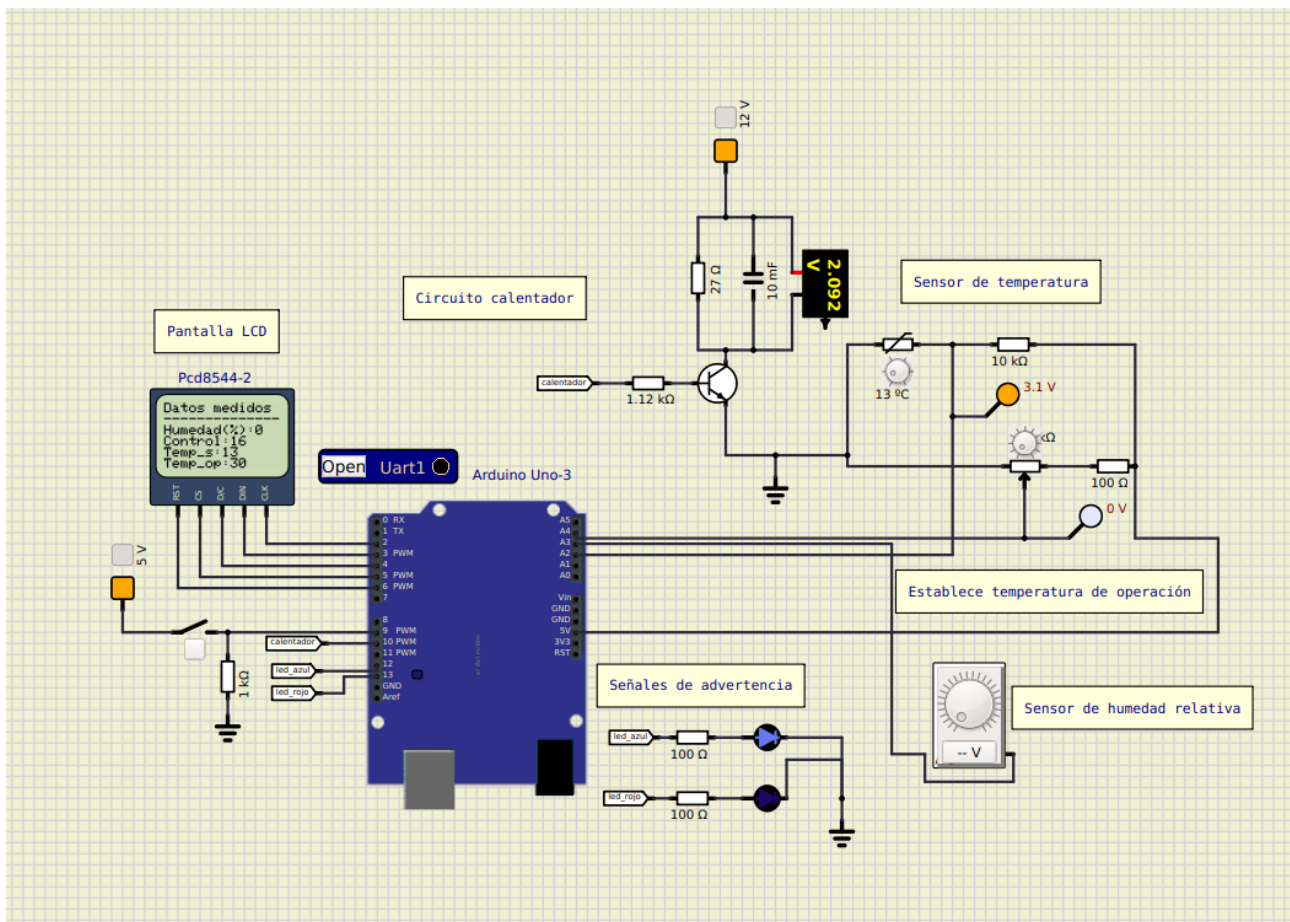


Figura 15: Sistema por debajo de temperatura de operación

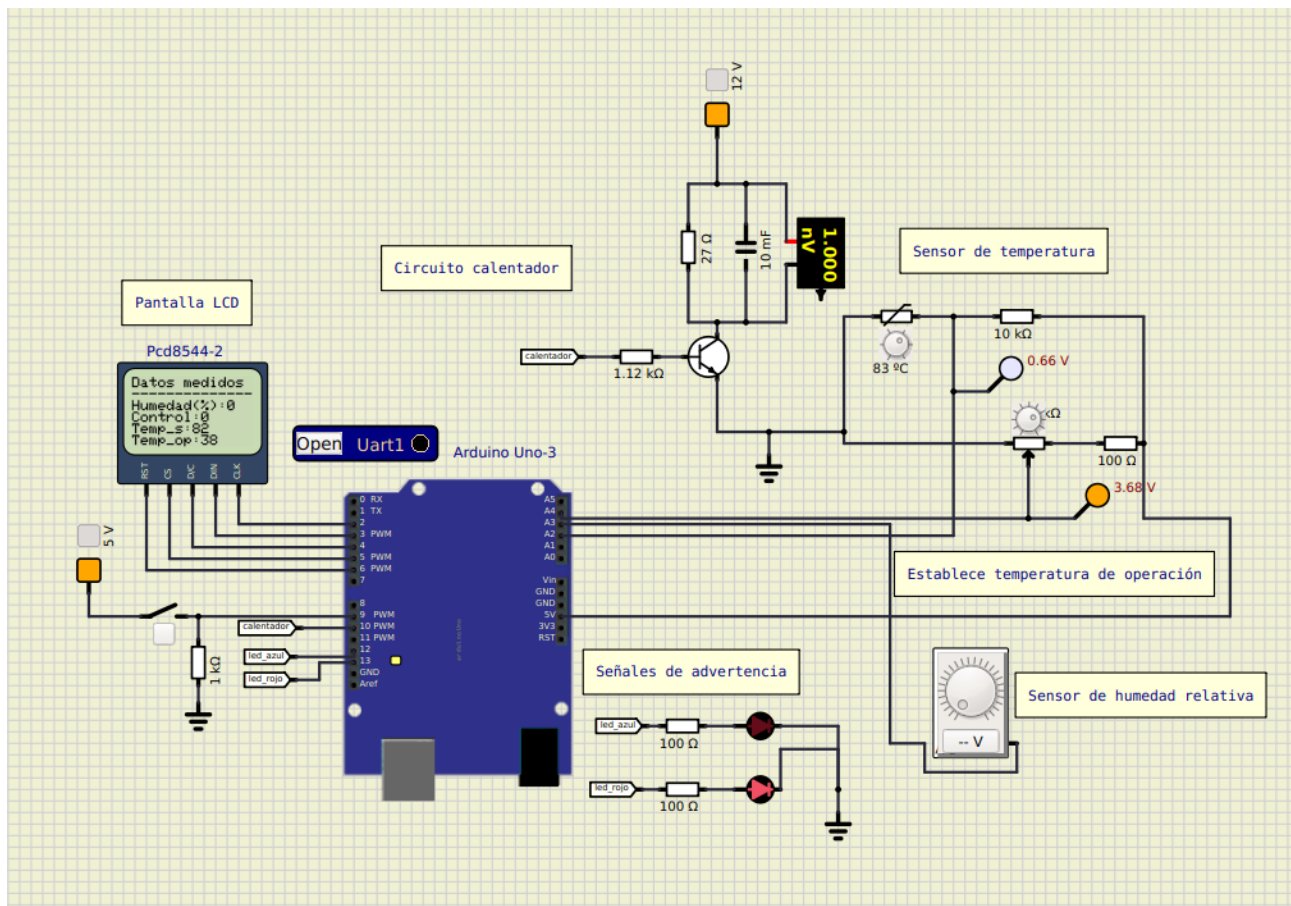


Figura 16: Sistema por encima de temperatura de operación

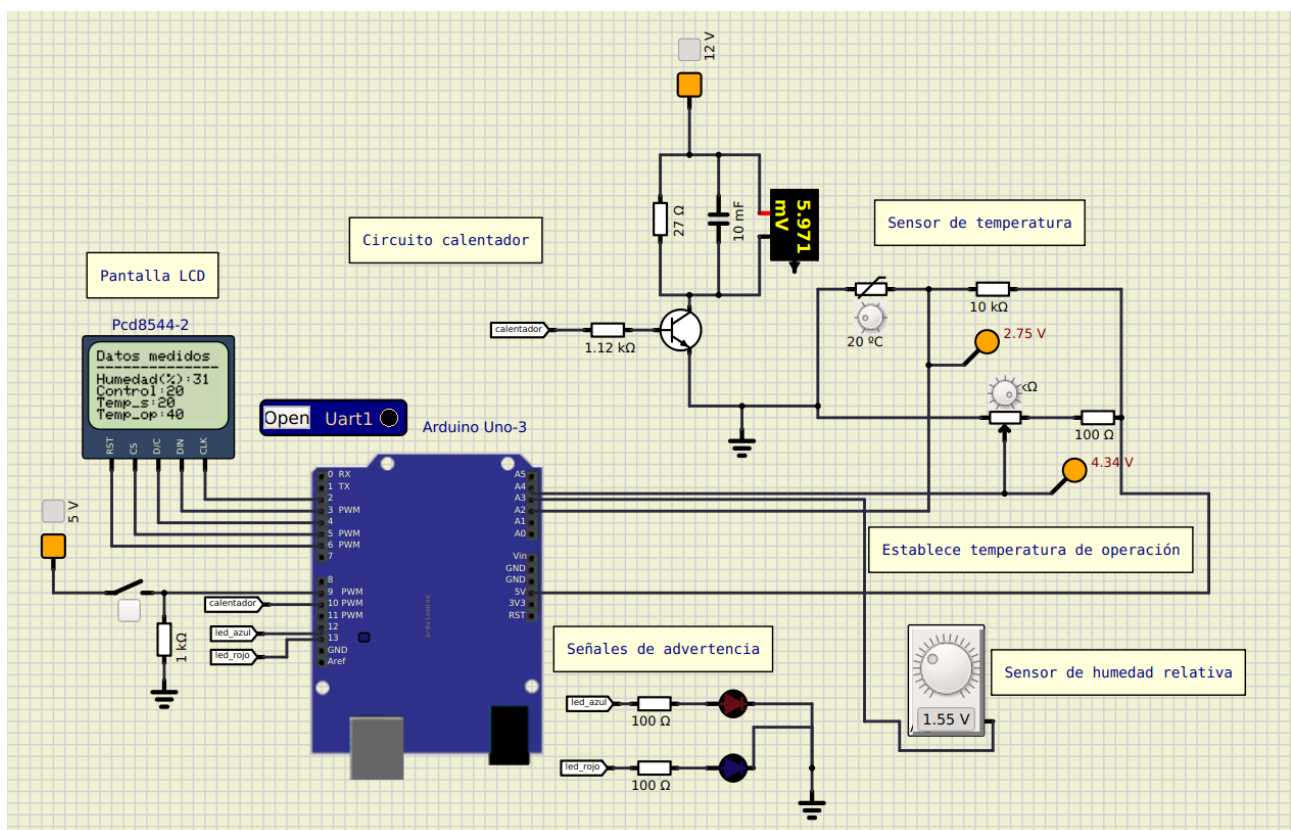


Figura 17: Sistema funcionando con control de 20°C

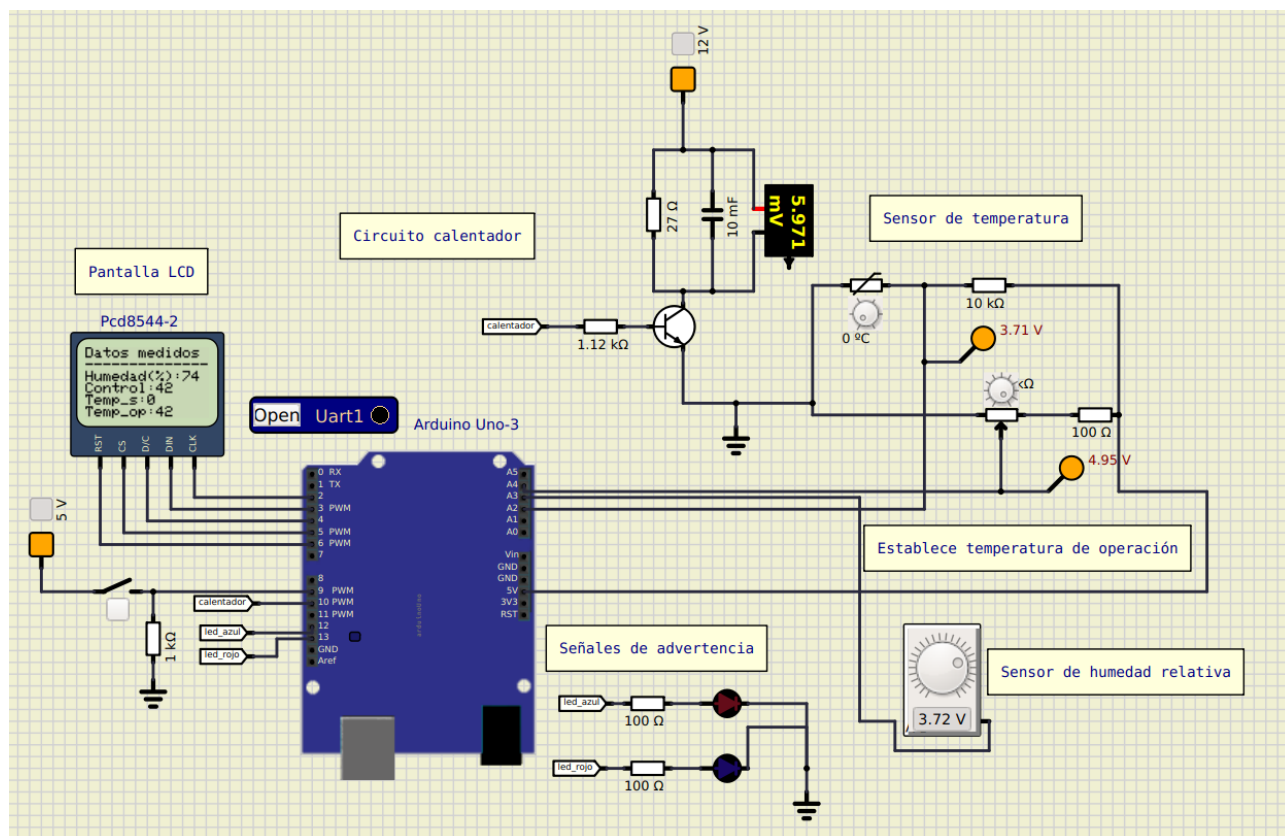


Figura 18: Sistema funcionando con control de 42°C

	A	B	C	D
1	<u>Humedad</u>	Control	Temp_s	Temp_op
2	0	30	0	30
3	0	30	0	30
4	0	30	0	30
5	0	30	0	30
6	0	30	0	30
7	0	30	0	30
8	0	30	0	30
9	0	30	0	30
10	0	30	0	30
11	0	30	0	30
12	0	30	0	30
13	0	30	0	30
14	0	30	0	30
15	0	30	0	30
16	0	30	0	30
17	0	30	0	30
18	0	30	0	30
19	0	30	0	30
20	0	30	0	30
21	0	30	0	30
22	0	30	0	30

Figura 19: Extracto de archivo .csv generafo

## 4. Conclusiones y Recomendaciones

En la elaboración del proyecto se llegaron a conocer las características de la placa Arduino UNO, la forma en que un microcontrolador puede ser utilizado como un controlador para controlar la salida de un sistema a partir de una señal de referencia, y la manera en que se puede establecer el envío de información entre un microcontrolador a otro dispositivo para la manipulación de datos, donde en este caso se guardaron en un archivo .csv utilizando un programa desarrollado bajo el lenguaje de programación Python. También se estudió el funcionamiento del termistor y la forma en que se puede obtener una lectura precisa de la temperatura a partir de la lectura de la tensión de esta. El controlador implementado fue específicamente uno del tipo PID, donde se observó que el ajuste de sus parámetros es un proceso que requiere gran análisis y pruebas.

En futuros trabajos se intentaría buscar alternativas para circuitos calentadores que sean más precisos. La variación de la tensión del calentador con respecto la señal de control no parece muy precisa, pero cumple con su finalidad. Además, si se llegara a implementar este proyecto, se esperaría que funcione satisfactoriamente, ya que el control evita que se lleguen a temperaturas por debajo del rango establecido, y también por encima cuando el ambiente lo permite. En conclusión, se desarrolló satisfactoriamente una incubadora de huevos en SimulIDE utilizando la placa Arduino UNO implementando a su vez un controlador PID y también el envío de datos de un microcontrolador a una computadora.

## 5. Referencias

- [1] Atmel. 8-bit avr microcontrollers: Atmega328/p. Disponible en: [https://www.laskakit.cz/user/related\\_files/atmel-42735-8-bit-avr-microcontroller-atmega328-328p\\_datasheet.pdf](https://www.laskakit.cz/user/related_files/atmel-42735-8-bit-avr-microcontroller-atmega328-328p_datasheet.pdf), 2016.
- [2] Arduino® UNO R3. Product reference manual. Disponible en: <https://docs.arduino.cc/resources/datasheets/A000066-datasheet.pdf>, 2022.
- [3] Ltd Aosong Electronics Co. Digital-output relative humidity temperature sensor/module dht22. Disponible en: <https://www.sparkfun.com/datasheets/Sensors/Temperature/DHT22.pdf>.
- [4] espruino. Pcd8544 lcd driver (nokia 5110). Disponible en: <https://www.espruino.com/PCD8544>, 2017.
- [5] Kingbright. T-1 3/4 (5mm) full color led lamp. Disponible en: <https://www.arduino.cc/documents/datasheets/LEDRGB-L-154A4SURK.pdf>.
- [6] TKS. 5 mm disc type for temperature sensing/compensation. Datasheet TTC05, TKS, mar 2008.
- [7] Gordon Varney. How to obtain the temperature value from a thermistor measurement. Disponible en: <https://www.allaboutcircuits.com/industry-articles/how-to-obtain-the-temperature-value-from-a-thermistor-measurement/>, 2020.

**6. Anexos**

# NTC Thermistor: TTC05 Series



## Φ5 mm Disc Type for Temperature Sensing/Compensation

### ■ Features

1. RoHS compliant
2. Body size Φ5mm
3. Radial lead resin coated
4. -30 ~ +125°C operating temperature range
5. Wide resistance range
6. Cost effective
7. Agency recognition :UL /CSA/TUV/CQC



### ■ Recommended Applications

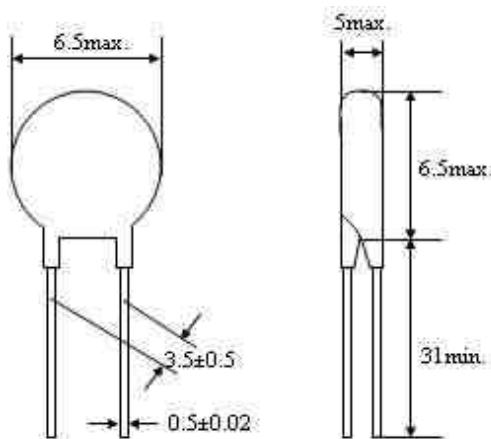
1. Home appliances (air conditioner, refrigerator, electric fan, electric cooker, washing machine, microwave oven, drinking machine, CTV, radio.)
2. Automotive electronics
3. Computers
4. Digital meter

### ■ Part Number Code

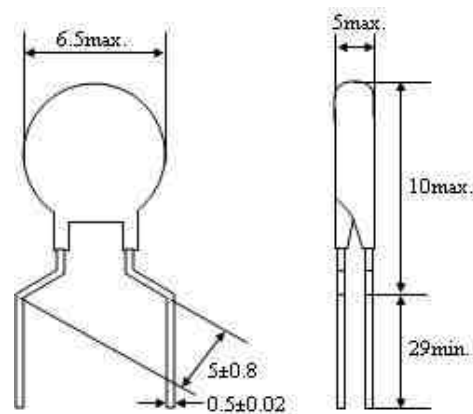
1	2	3	4	5	6	7	8	9	10	11	12	13	14
<b>Product Type</b> TTC Thinking NTC thermistor TTC type		<b>Body Size</b> 05 $\phi$ 5mm		<b>Zero Power Resistance at 25°C (<math>R_{25}</math>)</b> $R_{25} < 100 \Omega$ 005:5 $\Omega$ 015:15 $\Omega$ 050:50 $\Omega$ $R_{25} \geq 100 \Omega$ 101:100 $\Omega$ 682:6800 $\Omega$ 474:470000 $\Omega$				<b>Tolerance of <math>R_{25}</math></b> J $\pm 5\%$ K $\pm 10\%$ L $\pm 15\%$		<b>Appearance</b> S Straight lead I Kink lead		<b>Optional Suffix</b> Y RoHS compliant	

### ■ Structure and Dimensions

S type (Straight lead)

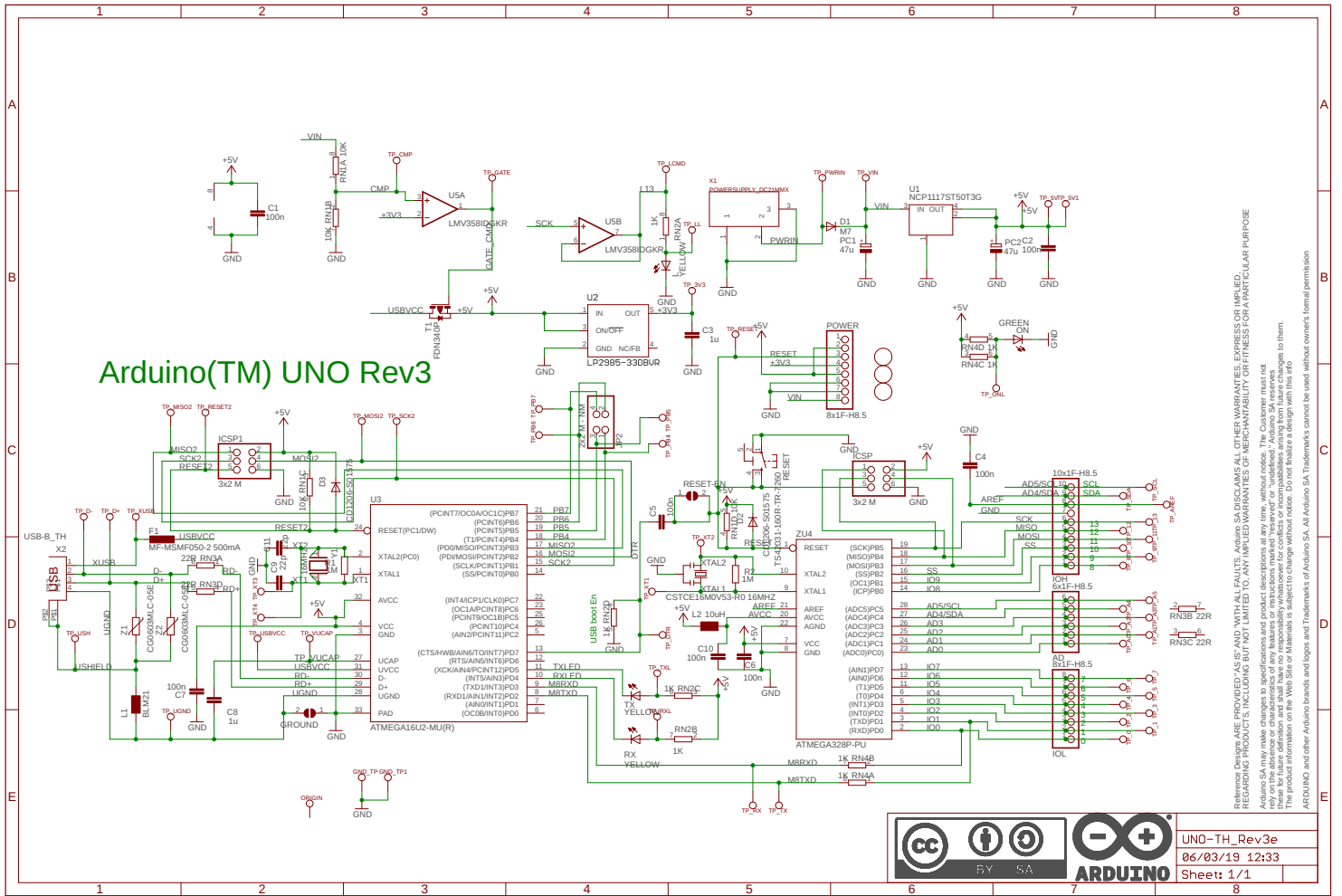


I type (Inner kink lead)



(Unit: mm)

# Arduino(TM) UNO Rev3



UNO-TH\_Rev3e  
06/03/19 12:33  
Sheet: 1/1