

Universidad de la Sabana

Automatización y Control de Procesos

Workshop 7

Juan David Baldion 0000191147
Hadden Carolina Diaz 0000156276

May 19, 2022

1 Introducción

Para la implementación del workshop 7, se dispone a la solución de una práctica con un circuito de Arduino haciendo uso de una configuración master-slave que se comunica por medio del protocolo I2C. Teniendo en cuenta lo anterior, se realizó una simulación en Tinkercad del circuito que posteriormente se implementara de forma física.

2 Modelo Tinkercad

Se realizó una implementación del modelo anteriormente mencionado, con ayuda de la herramienta Tinkercad, el cual muestra en una simulación funcional la conexión master-slave de dos arduinos y el funcionamiento de un sensor de temperatura.

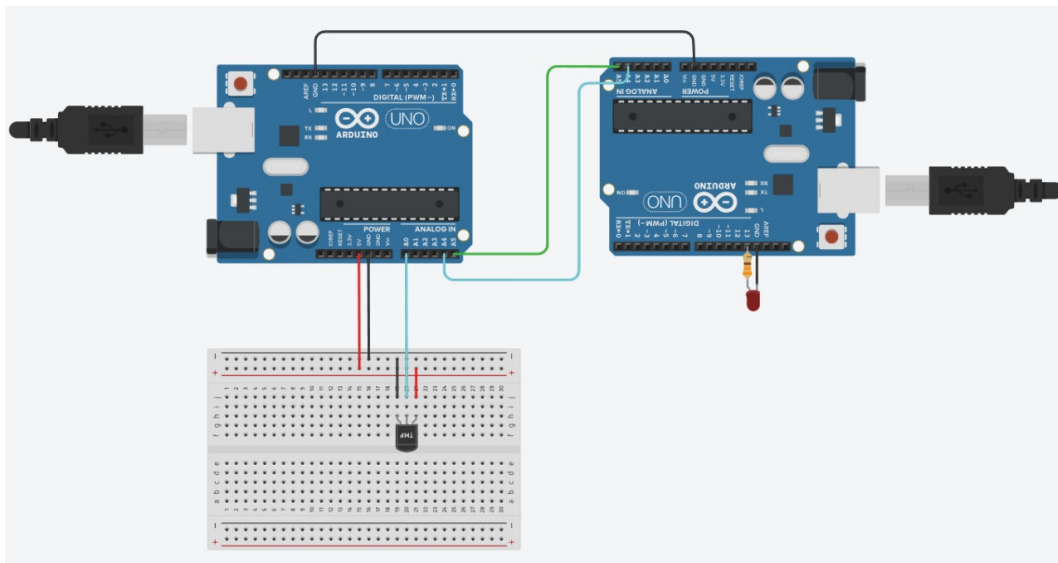


Figure 1: Simulación en Tinkercad.

Tras haber realizado una simulación exitosa del circuito en Tinkercad, se procedió compilar y subir el código a los Arduino UNO R3 por medio Arduino IDE, para posteriormente realizar las conexiones entre estos. La comunicación por I2C se uso por medio de la librería Wire.h de Arduino, la cual permitió realizar la conexión.

3 Implementación física

3.1 Materiales

Para la implementación de la simulación, se ocuparon los siguientes materiales:

- Arduino UNO
- Sensor de temperatura LM35
- LED
- Resistencia
- Protoboard
- Cable de cobre

3.2 Código Arduino

A continuación, se presenta el código ejecutado en Arduino IDE para la implementación de la simulación.

```
//Variables y Librerias
#include <Wire.h>
int sensor = A0;
int x = 0;
void setup()
{
    Serial.begin(9600);
    //Inicia Wire
    Wire.begin();
    pinMode(sensor, INPUT);
}

void loop()
{
    //Inicia Transmision de datos
    Wire.beginTransmission(1);
    //Lee lo que detecta el sensor
    x = analogRead(sensor);
    //Manda el dato x
    Wire.write(x);
    //Termina transmision
    Wire.endTransmission();
    //Esto se repite cada 1s
    delay(1000);
}
```

Figure 2: Código master-esclavo.

```

//Variables y Librerias
#include <Wire.h>
int sensor = A0;
int led = 7;
int x = 0;
long temp = 0;
void setup()
{
    Serial.begin(9600);
    //Inicia Wire y se usa metodo onReceive para que ejecute
    //el bloque de codigo cada vez que recibe datos
    Wire.begin(1);
    Wire.onReceive(receiveEvent);
    pinMode(7, OUTPUT);
}

void loop()
{
    //Inicia Transmision de datos
    Wire.beginTransmission(1);
    //Validacion de temperatura para el encendido del LED
    if(temp>30){
        digitalWrite(led, HIGH);
    }else{
        digitalWrite(led, LOW);
    }
}

//Este metodo se ejecuta cada vez que se reciben datos
//Se utiliza para calcular la temperatura apartir de lo que manda el sensor
void receiveEvent(int howMany){
    x = Wire.read();
    temp = ((long)x*5000)/1023;
    temp = temp/10;
    Serial.println(temp);
}

```

Figure 3: Código master-dominante.

3.3 Sensor LM35



Figure 4: Código master-dominante.

Teniendo en cuenta las características del sensor LM35, este genera un valor de 0 a 1023. Es por ello, que es necesario el establecimiento de la siguiente fórmula matemática para calcular la temperatura:

$$x = \frac{SENSOR * 5000mV}{1023} \quad (1)$$

$$TEMP = \frac{x}{10} \quad (2)$$

3.4 Prototipo físico

Por último, se implementó la solución descrita, en la cual, al encender el prototipo, el sensor captura la temperatura en tiempo real que se encuentre en el ambiente y si esta excede el mínimo de 30 grados centígrados, se enciende el led.

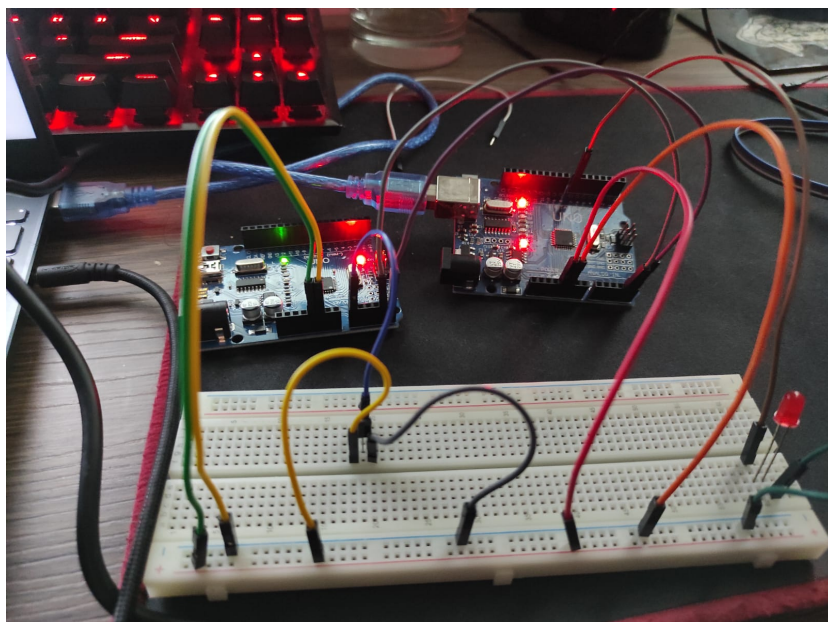


Figure 5: Prototipo físico led apagado.

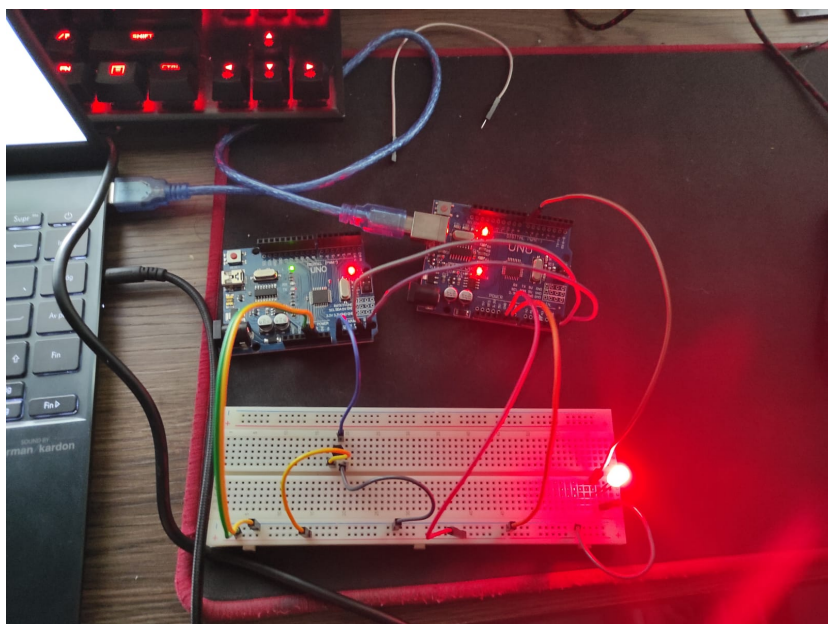


Figure 6: Prototipo físico led encendido.