

PRACTICA 5 : Buses de comunicación I (introducción y I2c)

A: ESCÁNER I2C

Código

```
#include <Arduino.h>
#include <Wire.h>

void setup()
{
    Wire.begin();

    Serial.begin(115200);
    while (!Serial);           // Leonardo: wait for serial monitor
    Serial.println("\nI2C Scanner");
}

void loop()
{
    byte error, address;
    int nDevices;

    Serial.println("Scanning...");

    nDevices = 0;
    for(address = 1; address < 127; address++ )
    {
        // The i2c_scanner
        // the Write.endTransmission to see if
        // a device did acknowledge to the address.
        Wire.beginTransmission(address);
        error = Wire.endTransmission();

        if (error == 0)
        {
            Serial.print("I2C device found at address 0x");
            if (address<16)
                Serial.print("0");
            Serial.print(address,HEX);
            Serial.println("  !");

            nDevices++;
        }
        else if (error==4)
        {
            Serial.print("Unknown error at address 0x");

```

```

        if (address<16)
            Serial.
            Serial.print("0");
            Serial.println(address,HEX);
        }
    }
    if (nDevices == 0)
        Serial.println("No I2C devices found\n");
    else
        Serial.println("done\n");

    delay(5000);          // wait 5 seconds for next scan
}

```

Funcionamiento del programa

En primer lugar, necesitaremos la librería "Wire.h". A continuación vemos que declaramos la función Setup, esta inicializa la conexión con el puerto série.

```

#include <Arduino.h>
#include <Wire.h>

void setup()
{
    Wire.begin();

    Serial.begin(115200);
    while (!Serial);          // Leonardo: wait for serial monitor
    Serial.println("\nI2C Scanner");
}

```

Ahora procedemos a definir la función Loop, en primer lugar definimos dos variables de tipo "byte" para guardar información sobre el dispositivo I2C y a continuación inicializamos el contador de dispositivos "nDevies" en 0.

Declaramos un bucle FOR que nos permitirá leer cualquier dispositivo I2C y este bucle nos retornará un valor byte entre 0 y 4 que nosotros entenderemos como:

· 0: El scan se ha realizado con éxito. · 1: Se han recibido datos demasiado largos y no caben en el búfer de transmisión. · 2: Se ha recibido NACK al transmitir la dirección. · 3: Se ha recibido NACK al transmitir datos. · 4: Otro error que no conocemos.

Una vez se ha cerrado el bucle Loop, el dispositivo I2C no se mantendrá conectado a la placa ESP32 ya que se habrá ejecutado la operación de finalizar la transmisión.

```

void loop()
{
    byte error, address;

```

```

int nDevices;

Serial.println("Scanning...");

nDevices = 0;
for(address = 1; address < 127; address++ )
{
  // The i2c_scanner
  // the Write.endTransmission to see if
  // a device did acknowledge to the address.
  Wire.beginTransmission(address);
  error = Wire.endTransmission();

  if (error == 0)
  {
    Serial.print("I2C device found at address 0x");
    if (address<16)
      Serial.print("0");
    Serial.print(address,HEX);
    Serial.println("  !");

    nDevices++;
  }
  else if (error==4)
  {
    Serial.print("Unknown error at address 0x");
    if (address<16)
      Serial.
        Serial.print("0");
    Serial.println(address,HEX);
  }
}
if (nDevices == 0)
  Serial.println("No I2C devices found\n");
else
  Serial.println("done\n");

delay(5000);          // wait 5 seconds for next scan
}

```

Salida por el Terminal

Si la conexión funciona correctamente, el programa imprimirá lo siguiente por pantalla:

```

Scanning...
I2C device found at address 0x3C  !
done

```

El nombre del dispositivo aparecería en lugar de 3C.