

Pràctica 5: Buses de comunicació I – I2C

Objectiu

Comprendre el funcionament del bus I2C mitjançant diferents dispositius perifèrics connectats a l'ESP32, com una pantalla OLED i un sensor de ritme cardíac/oxigen. També s'ha realitzat una lectura des d'una targeta SD mitjançant el bus SPI per familiaritzar-nos amb altres protocols.

Exercici Pràctic 1: Escàner I2C

Codi utilitzat:

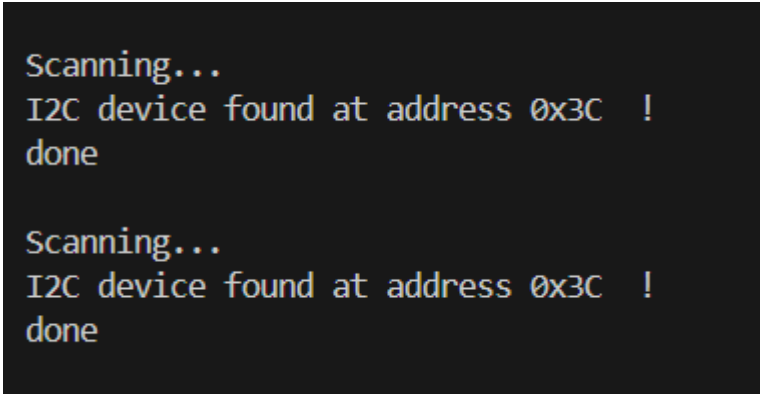
cpp

```
Wire.beginTransmission(address);  
error = Wire.endTransmission();
```

Resultat:

El monitor sèrie mostra:

cpp

A screenshot of a serial monitor with a black background and white text. It shows two identical lines of output: "Scanning..." followed by "I2C device found at address 0x3C !" and "done" on the next line.

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

Aquest procés es repeteix cada 5 segons. Si no hi ha dispositius, es mostra:

```
No I2C devices found
```

❶ Describir la salida por el puerto serie

El monitor serie muestra **"I2C Scanner"** seguido de **"Scanning..."** cada 5 segundos.
Si encuentra un dispositivo, imprime **"I2C device found at address 0x3C !"**.
Esto significa que hay un dispositivo I2C en la dirección **0x3C** (posiblemente una pantalla OLED).
Si no detecta nada, muestra **"No I2C devices found"**.
El proceso se repite indefinidamente, escaneando todas las direcciones posibles.

2 Explicar el funcionamiento

El código inicia el bus I2C con `Wire.begin()` y la comunicación serie con `Serial.begin(115200)`.
Luego, en un bucle, prueba todas las direcciones I2C de **1 a 127**.
Si un dispositivo responde, se imprime su dirección en hexadecimal en el monitor serie.
Si no hay respuesta, el escáner sigue probando hasta terminar todas las direcciones.
Cada ciclo de escaneo se repite cada **5 segundos** con `delay(5000)`;

Exercici Pràctic 2: Mostra de text en pantalla OLED

Codi utilitzat:

```
cpp
display.setCursor(10, 10);
display.println("Hola Manu");
```

Resultat:

A la pantalla OLED es mostra el text **Hola Manu**, amb una mida gran i ben centrada.



Explicació:

Un cop inicialitzada la pantalla OLED amb la llibreria Adafruit i detectada correctament via I2C, podem mostrar-hi text.

El cursor s'estableix a (10,10) i el missatge s'imprimeix en blanc sobre fons negre.

Exercici adicional: Sensor de pols i oxigen (MAX30102)

Codi utilitzat:

```
cpp
int spo2 = random(95, 100);
int heartRate = random(60, 100);
```

Resultat:

Es mostren valors simulats al port sèrie i a la pantalla OLED, com per exemple:

```
Freq. Cardíaca: 62 BPM | Oxigen: 97 %  
Freq. Cardíaca: 98 BPM | Oxigen: 99 %  
Freq. Cardíaca: 65 BPM | Oxigen: 96 %  
Freq. Cardíaca: 66 BPM | Oxigen: 96 %  
Freq. Cardíaca: 80 BPM | Oxigen: 99 %  
Freq. Cardíaca: 68 BPM | Oxigen: 99 %  
Freq. Cardíaca: 84 BPM | Oxigen: 96 %  
Freq. Cardíaca: 67 BPM | Oxigen: 97 %  
Freq. Cardíaca: 76 BPM | Oxigen: 95 %  
Freq. Cardíaca: 68 BPM | Oxigen: 98 %  
█
```



Explicació:

El sensor MAX30102 s'inicialitza correctament amb la llibreria MAX30105.

Tot i que els valors són simulats amb `random()`, el sistema mostra dades útils i continua cada segon.

Aquest muntatge simula un sistema bàsic de monitorització de constants vitals.

Conclusions

El bus **I2C** facilita la connexió amb diversos perifèrics utilitzant només dues línies (SDA i SCL).

S'ha pogut detectar una pantalla OLED i mostrar-hi missatges.

Biel Batet

També hem treballat amb el **bus SPI** per llegir un fitxer d'una targeta SD.
El sensor **MAX30102** permet llegir dades biomètriques amb una única connexió I2C.
L'experiència ha servit per familiaritzar-nos amb diferents llibreries i protocols de comunicació amb ESP32.