

## PINES SCL GPIO22

PINES DA GPIO21 `#include <Wire.h>`

`#include <Adafruit_Sensor.h>`

`#include <Adafruit_BME280.h>`

`#include <LiquidCrystal_I2C.h>`

`#define SDA_PIN 21`

`#define SCL_PIN 22`

`LiquidCrystal_I2C lcd(0x27, 16, 2);`

`Adafruit_BME280 bme;`

`// tiempos (ms)`

`const unsigned long TIEMPO_APAGADO = 40000; // 40 segundos`

`const unsigned long TIEMPO_ENCENDIDO = 10000; // 10 segundos`

`unsigned long marcaTiempo = 0;`

`bool mostrando = false;`

`void setup() {`

`Serial.begin(115200);`

`Wire.begin(SDA_PIN, SCL_PIN);`

`lcd.init();`

`lcd.backlight();`

`lcd.clear();`

`if (!bme.begin(0x77)) {`

`lcd.print("BME280 ERROR");`

`while (1);`

`}`

`marcaTiempo = millis();`

`}`

`void loop() {`

`unsigned long ahora = millis();`

`if (!mostrando && (ahora - marcaTiempo >= TIEMPO_APAGADO)) {`

`// pasar a estado MOSTRANDO`

`mostrando = true;`

`marcaTiempo = ahora;`

```
float temp = bme.readTemperature();
float hum  = bme.readHumidity();

lcd.clear();
lcd.setCursor(0, 0);
lcd.print("Temp: ");
lcd.print(temp, 1);
lcd.print(" C");

lcd.setCursor(0, 1);
lcd.print("Hum:  ");
lcd.print(hum, 1);
lcd.print(" %");

Serial.println("MOSTRANDO 10s");
}

if (mostrando && (ahora - marcaTiempo >= TIEMPO_ENCENDIDO)) {
    // volver a estado APAGADO
    mostrando = false;
    marcaTiempo = ahora;
    lcd.clear();

    Serial.println("APAGADO 40s");
}
}
```

### **BME280**

- VCC → 3.3V
- GND → GND
- SDA → GPIO 21
- SCL → GPIO 22

### **Pantalla LCD**

- VCC → 5V
  - GND → GND
  - SDA → GPIO 21
  - SCL → GPIO 22
-