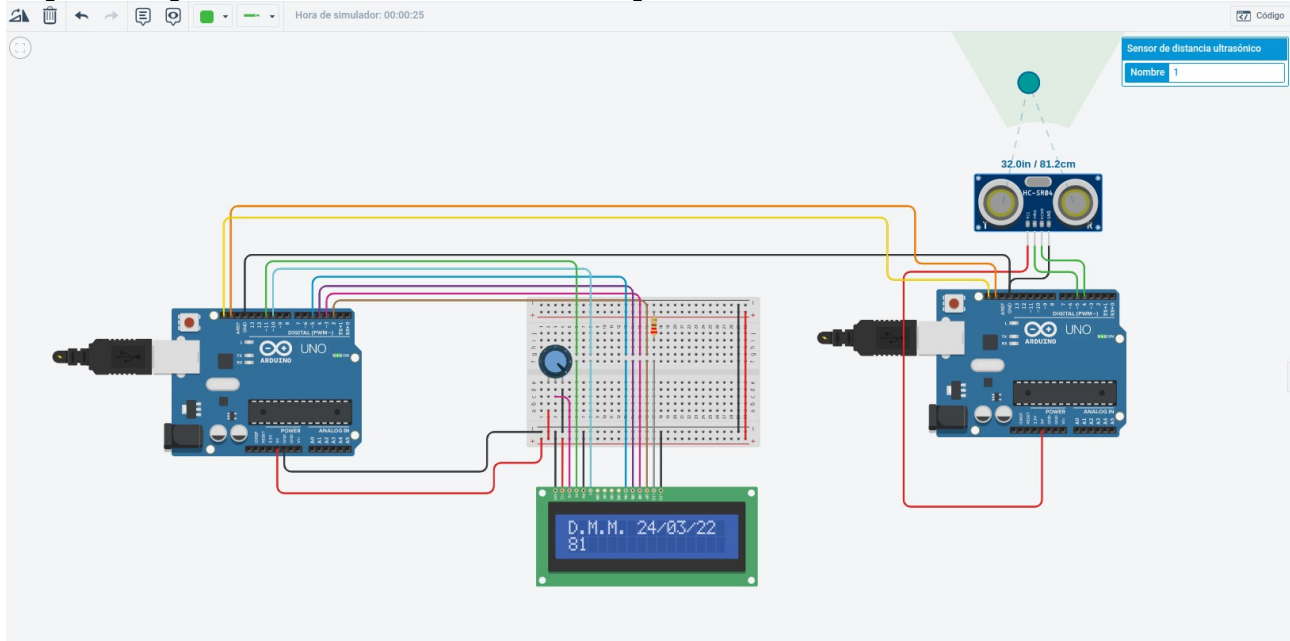


SISTEMAS CON MICROPROCESADORES: PRACTICA 4

Dirección web de los proyectos Tinkercad:

- <https://www.tinkercad.com/things/7B2X0OnIwjz>

Captura de pantalla del funcionamiento de la pantalla LCD



Código fuente (maestro):

```
// C++ code
//

#include <Wire.h>
#include <LiquidCrystal.h>

#define DIRECC_I2C_ESCLAVO 3
#define PIN_RS 11
#define PIN_EN 10
#define PIN_D4 5
#define PIN_D5 4
#define PIN_D6 3
#define PIN_D7 2

LiquidCrystal pantalla(PIN_RS, PIN_EN, PIN_D4, PIN_D5, PIN_D6, PIN_D7);
void setup()
{
  Wire.begin();
```

```

    pantalla.begin(16,2);
    pantalla.print("D.M.M. 24/03/22");
}

void leerEsclavo(){
    pantalla.setCursor(0,1);
    int cm1, cm2;

    Wire.requestFrom(DIRECC_I2C_ESCLAVO, 2);
    cm1 = Wire.read();
    cm2 = Wire.read();

    if(cm1 != -1 && cm2 != -1){
        uint16_t cm;
        cm = (cm2 << 8) + cm1;
        pantalla.print(cm);
    }

    else{
        pantalla.print("ERROR");
    }
}

void borrarLinea(){
    pantalla.setCursor(0,1);

    for(int i=0; i < 16; i++){
        pantalla.print(" ");
    }
}

void loop()
{
    pantalla.setCursor(0,1);
    leerEsclavo();
    delay(1000);
    borrarLinea(); }

```

Código fuente (esclavo):

```
// C++ code
//

#include <Wire.h>

#define PIN_DISPARDOR 5
#define PIN_ECO 4
#define TIMEOUT_ECO 23200
#define DIRECC_I2C 3

uint16_t cm;
void setup()
{
    pinMode(PIN_DISPARDOR, OUTPUT);
    pinMode(PIN_ECO, INPUT);
    digitalWrite(PIN_DISPARDOR, 0);
    Wire.begin(DIRECC_I2C);
    Wire.onRequest(callback_peticion_i2c);
    cm = 0;
}

unsigned long leeDistancia(uint8_t triggerPin, uint8_t ecoPin)
{
    delayMicroseconds(2); // Por seguridad
    digitalWrite(triggerPin, HIGH); // Activa el trigger 10 us
    delayMicroseconds(10);
    digitalWrite(triggerPin, LOW);
    return pulseIn(ecoPin, HIGH, TIMEOUT_ECO); // Lee la anchura del pulso en
us
}

void callback_peticion_i2c()
{
    uint8_t b[2];
    b[0]=lowByte(cm); // cm & 0xFF;
    b[1]=highByte(cm); // (cm & 0xFF00) >> 8
    Wire.write(b, 2);
}
```

```
void loop()
{
  cm = leeDistancia(PIN_DISPADOR, PIN_ECO) / 58;
  delay(500);
}
```