

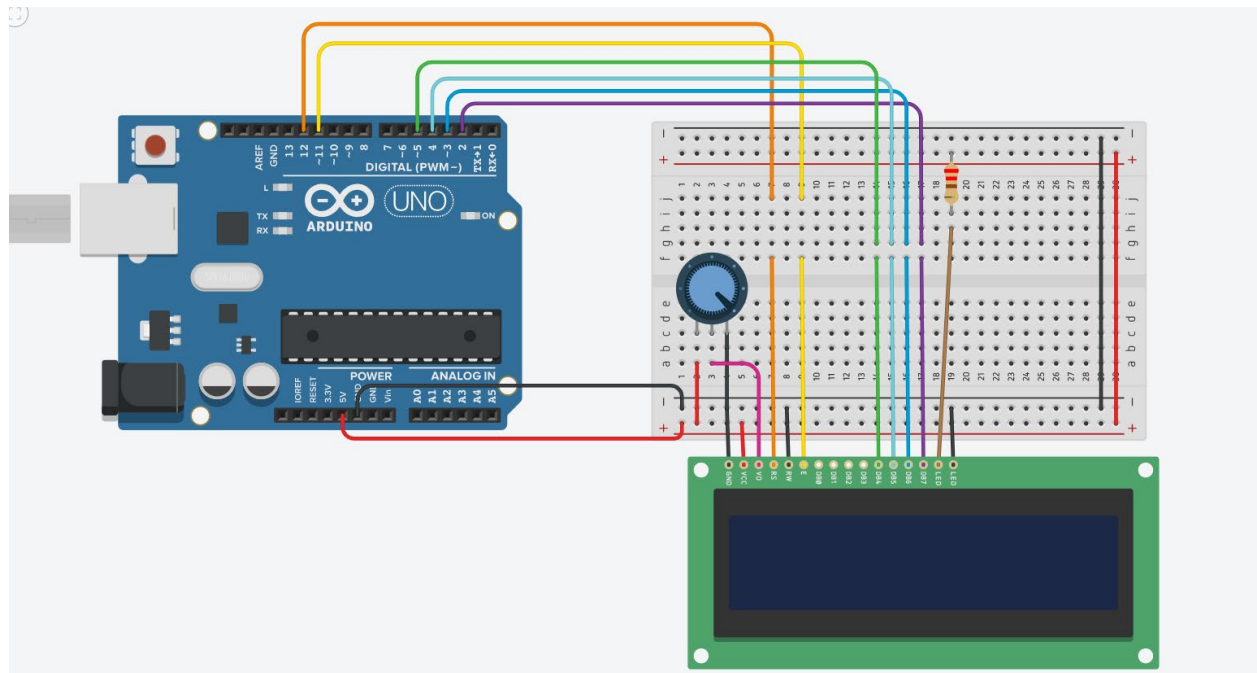
Laborator 4. Folosirea unor componente în Sisteme Încorporate

Display LCD

Se poate folosi pentru afișarea unor texte sau valori pe display.

Conectarea firelor la display:

- * LCD RS pin to digital pin 12
- * LCD Enable pin to digital pin 11
- * LCD D4 pin to digital pin 5
- * LCD D5 pin to digital pin 4
- * LCD D6 pin to digital pin 3
- * LCD D7 pin to digital pin 2
- * LCD R/W pin to ground
- * LCD VSS pin to ground
- * LCD VCC pin to 5V
- * 10K resistor:
 - * ends to +5V and ground
 - * wiper to LCD VO pin (pin 3)



Codul sursă:

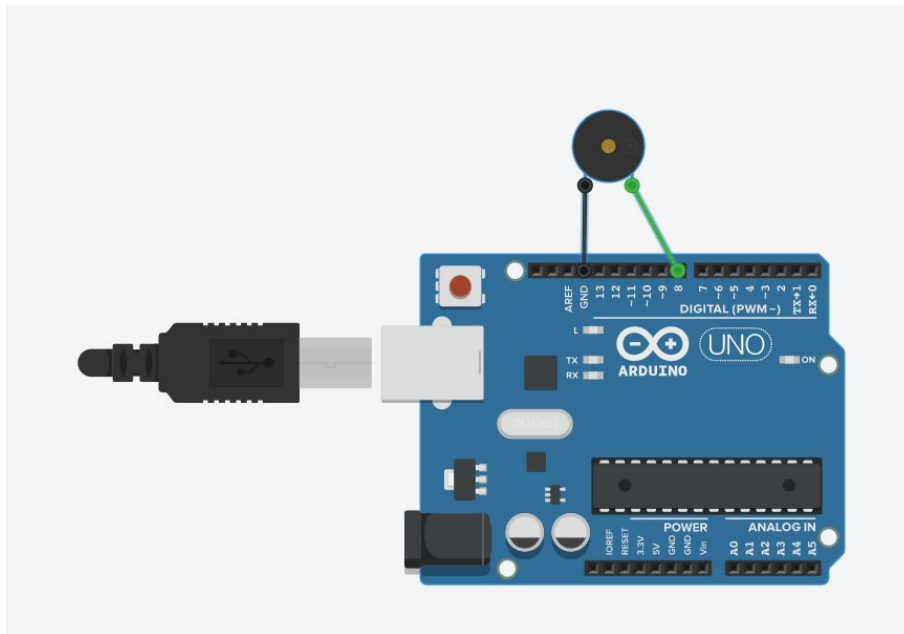
```
// include the library code:
#include <LiquidCrystal.h>

// initialize the library with the numbers of the interface pins
LiquidCrystal lcd(12, 11, 5, 4, 3, 2);

void setup() {
  // set up the LCD's number of columns and rows:
  lcd.begin(16, 2);
  // Print a message to the LCD.
  lcd.print("hello, world!");
}

void loop() {
  // set the cursor to column 0, line 1
  // (note: line 1 is the second row, since counting begins with 0):
  lcd.setCursor(0, 1);
  // print the number of seconds since reset:
  lcd.print(millis() / 1000);
}
```

Difuzorul



Codul sursa

```

#define NOTE_C4 262
#define NOTE_G3 196
#define NOTE_A3 220
#define NOTE_B3 247
#define NOTE_C4 262

// notes in the melody:
int melody[] = {
  NOTE_C4, NOTE_G3, NOTE_G3, NOTE_A3, NOTE_G3, 0, NOTE_B3, NOTE_C4
};

// note durations: 4 = quarter note, 8 = eighth note, etc.:
int noteDurations[] = {
  4, 8, 8, 4, 4, 4, 4, 4
};

void setup() {
  // iterate over the notes of the melody:
  for (int thisNote = 0; thisNote < 8; thisNote++) {

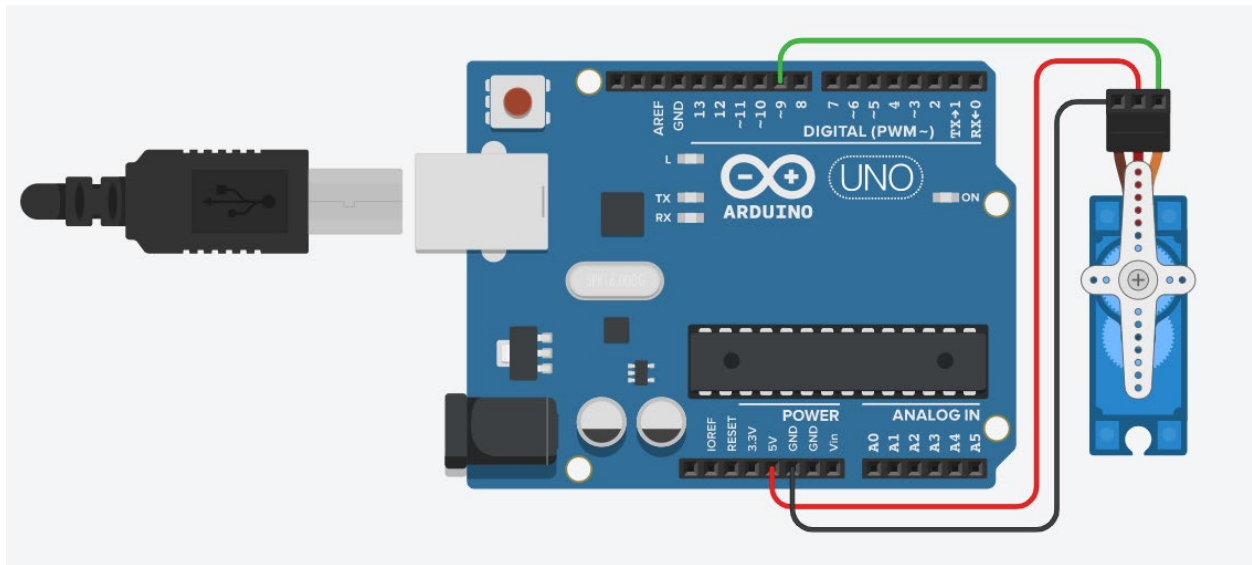
    // to calculate the note duration, take one second
    // divided by the note type.
    //e.g. quarter note = 1000 / 4, eighth note = 1000/8, etc.
    int noteDuration = 1000 / noteDurations[thisNote];
    tone(8, melody[thisNote], noteDuration);

    // to distinguish the notes, set a minimum time between them.
    // the note's duration + 30% seems to work well:
    int pauseBetweenNotes = noteDuration * 1.30;
    delay(pauseBetweenNotes);
    // stop the tone playing:
    noTone(8);
  }
}

void loop() {
  // no need to repeat the melody.
}

```

Servomotor



Codul sursa

```
#include <Servo.h>

int pos = 0;

Servo servo_9;

void setup()
{
    servo_9.attach(9, 500, 2500);
}

void loop()
{
    // sweep the servo from 0 to 180 degrees in steps
    // of 1 degrees
    for (pos = 0; pos <= 180; pos += 1) {
        // tell servo to go to position in variable 'pos'
        servo_9.write(pos);
        // wait 15 ms for servo to reach the position
        delay(15); // Wait for 15 millisecond(s)
    }
    for (pos = 180; pos >= 0; pos -= 1) {
        // tell servo to go to position in variable 'pos'
        servo_9.write(pos);
        // wait 15 ms for servo to reach the position
        delay(15); // Wait for 15 millisecond(s)
    }
}
```

Exercitiu

Componente:

- Arduino Uno
- 2 led-uri
- 1 LCD
- 1 servomotor
- 1 difuzor
- 3 butoane

Cerințe

Când se alimentează sistemului se va aprinde LED-ul roșu.

Când se apasă pe primul buton sistemul va face următoarele:

- Se va aprinde LED-ul verde
- Pe LCD se va afișa un mesaj de pornire
- Când se va apăsa pe al doilea buton servomotorul se va deplasa spre stanga, iar difuzorul va cânta un anumit cântec
- Când se va apăsa pe al treilea buton servomotorul se va deplasa spre dreapta, iar difuzorul va cânta un alt cântec

Se va crea un document Word cu următoarele informații:

- Prima pagină va conține numele studentului
- A doua și a treia pagină va conține:
 - o Cerințele exercițiului
 - o O figură realizată cu simulatorul Arduino
 - o Codul sursă, vor fi și comentarii în cod
 - o Link-ul cu sistemul de pe Tinkercad