# Proiect ARDUINO

Gugulea Ioana-Delia

322AC

1. **Scenariu practic**

În acest proiect am realizat un sistem de monitorizare al nivelului apei. Acesta poate fi utilizat în măsurarea unui rezervor(prevenirea exploziilor încălzitoarelor de apă), în detectarea prezenței apei (precipitații) sau în construirea electronicelor submersibile.

Pentru exemplul prezentat, voi construi un indicator simplu, de nivel al apei, presupunând un senzor de apă, introdus într-un recipient. Pe măsură ce nivelul apei variază, LED-urile indică starea acesteia (nivel scăzut, mediu sau ridicat), iar display-ul 4-digit TM1637 va afișa valorile senzorului. Pentrul nivelul scăzut, se va aprinde LED-ul verde, pentru cel mediu, se va aprinde LED-ul albastru, iar pentru cel ridicat, se va aprinde LED-ul roșu. Informațiile vor fi afișate și prin intermediul portului serial către un PC conectat. De asemenea, voi folosi un buzzer activ, care va emite semnale acustice în funcție de nivelul apei (de exemplu, un sunet de avertizare în caz de nivel ridicat).

1. **Diagramă bloc simplificată**

Active Buzzer



USB



LEDs

PC

TMP1637 Digit Display

Breadboard

µC

Water Sensor

1. **Funcționalitate**

Senzorul de apă are zece urme expuse de cupru, cinci sunt urme de putere, iar restul cinci urme senzoriale (atenție: pentru a evita rata de coroziune, senzorul se va utiliza doar în timpul citirii). Între fiecare două urme, există o urmă de simț, care se va conecta cu puterea în momentul atingerii apei. Aceste urme formează un rezistor variabil (la fel ca un potențiometru) a cărui rezistență variază în funcție de cât de mult sunt expuse la apă.

Această rezistență variază invers proporțional cu adâncimea de scufundare a senzorului în apă:

* Cu cât senzorul este scufundat în mai multă apă, cu atât conductivitatea este mai bună și rezistența este mai mică.
* Cu cât senzorul este scufundat în mai puțină apă, cu atât conductivitatea este mai slabă și rezistența este mai mare.

Senzorul generează o tensiune de ieșire proporțională cu rezistența; prin măsurarea acestei tensiuni se poate determina nivelul apei.

1. **Mod de lucru**

Voi prezenta modul de realizare al circuitului, rezultatele implemetării codului, precum și valorile citite.

A circuit board with wires and a red and white led

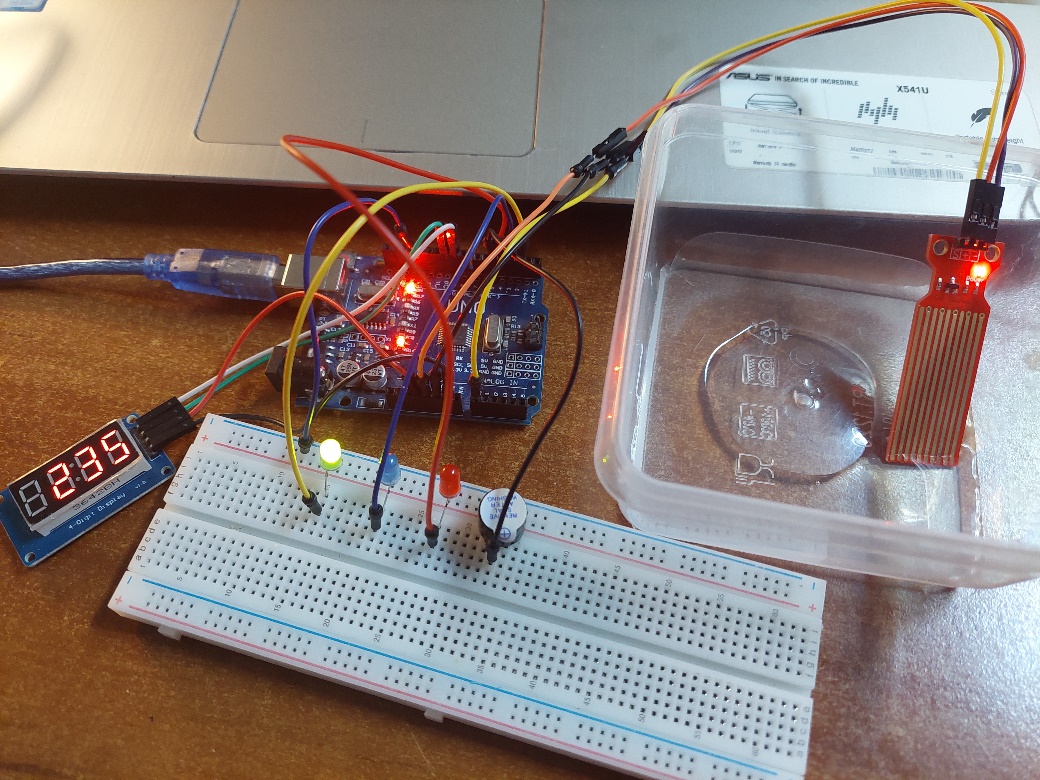
Description automatically generated with medium confidence

**4-Digit Display -> UNO**: CLK ->D9; DIO -> D10; VCC -> 3.3V; GND -> GND

BreadBoard -> UNO; LED verde -> D2; LED albastru -> D3; LED rosu -> D4; buzzer active ->D5; **senzor:** S (signal) -> A0, **+**(VCC) -> 5V, **-** -> GND

**Nivel scăzut al apei**

* **LED-ul verde se aprinde când valorile senzorului > 0 si <320**

 (LOW)

**Nivel mediu al apei**

* **LED-ul albastru se aprinde când valorile senzorului > 320 si <390**

A circuit board with wires and a thermometer

Description automatically generated (MEDIUM)

**Nivel ridicat al apei**

* **LED-ul roșu se aprinde când valorile senzorului > 390 si <420**
* **atunci când valorile cresc peste 390, buzzer-ul activ se declanșează și emite un sunet**

1. **Prezentare cod Arduino**
2. //biblioteca integrata pentru display digital
3. #include <TM1637Display.h>
4. //definesc pinii display-ului
5. #define CLK 9
6. #define DIO 10
7. //creez un obiect din clasa TM1637Display
8. //constructorul accepta cele doua intrari: CLK si DIO
9. TM1637Display display(CLK, DIO);
10. //valorile minime si maxime
11. int lowerThreshold = 320;
12. int upperThreshold = 393;
13. //declararea pinilor Arduino la care sunt conectati pinii senzorului
14. #define sensorPower 7
15. #define sensorPin A0
16. //variabila pentru stocarea nivelului curent al apei
17. int val = 0;
18. //declararea pinilor pentru fiecare led
19. int redLED = 2;
20. int blueLED = 3;
21. int greenLED = 4;
22. //pin pentru buzzer
23. #define buzzerPin 5
24. void setup() {
25. //initializare comunicare serial pentru depanare
26. Serial.begin(9600);
27. //se configureaza pinul de putere pentru senzorul de apa
28. pinMode(sensorPower, OUTPUT);
29. digitalWrite(sensorPower, LOW);
30. //configurare LED-uri la OUTPUT
31. pinMode(redLED, OUTPUT);
32. pinMode(blueLED, OUTPUT);
33. pinMode(greenLED, OUTPUT);
34. //inițializarea display-ului TM1637
35. display.setBrightness(0x0a);  //puteți ajusta luminozitatea (0x00 până la 0x0f)
36. //initial toate LED-urile sunt inchise, iar diplay-ul sters
37. digitalWrite(redLED, LOW);
38. digitalWrite(blueLED, LOW);
39. digitalWrite(greenLED, LOW);
40. display.clear();
41. }
42. void loop() {
43. //citirea valorii de la senzorul de apa
44. int level = readSensor();
45. if (level == 0) {
46. Serial.println("Water Level: Empty");
47. digitalWrite(redLED, LOW);
48. digitalWrite(blueLED, LOW);
49. digitalWrite(greenLED, LOW);
50. display.showNumberDec(0, false);
51. } else if (level > 0 && level <= lowerThreshold) {
52. Serial.println("Water Level: Low");
53. digitalWrite(redLED, HIGH);
54. digitalWrite(blueLED, LOW);
55. digitalWrite(greenLED, LOW);
56. display.showNumberDec(level, false);
57. } else if (level > lowerThreshold && level <= upperThreshold) {
58. Serial.println("Water Level: Medium");
59. digitalWrite(redLED, LOW);
60. digitalWrite(blueLED, HIGH);
61. digitalWrite(greenLED, LOW);
62. display.showNumberDec(level, false);
63. } else if (level > upperThreshold) {
64. Serial.println("Water Level: High");
65. digitalWrite(redLED, LOW);
66. digitalWrite(blueLED, LOW);
67. digitalWrite(greenLED, HIGH);
68. display.showNumberDec(level, false);
69. //emite un sunet de avertizare folosind buzzer-ul pentru 1 secundă
70. tone(buzzerPin, 1000, 1000);
71. }
72. delay(1000);
73. }
74. //functie pentru a citi valoarea senzorului de apa
75. int readSensor() {
76. digitalWrite(sensorPower, HIGH);
77. delay(10);
78. val = analogRead(sensorPin);
79. digitalWrite(sensorPower, LOW);
80. return val;
81. }

**6. Bill of materials**

**Arduino UNO -** [*https://ardushop.ro/ro/home/29-placa-de-dezvoltare-uno-r3.html*](https://ardushop.ro/ro/home/29-placa-de-dezvoltare-uno-r3.html)

**Water Sensor –** [*https://ardushop.ro/ro/electronica/46-modul-senzor-nivel-apa.html?search\_query=Senzor+apa&results=519*](https://ardushop.ro/ro/electronica/46-modul-senzor-nivel-apa.html?search_query=Senzor+apa&results=519)

**Breadboard –** [*https://ardushop.ro/ro/electronica/33-breadboard-830.html?search\_query=Breadboard&results=32*](https://ardushop.ro/ro/electronica/33-breadboard-830.html?search_query=Breadboard&results=32)

**LED-uri –** [*https://ardushop.ro/ro/electronica/299-led-5mm.html?search\_query=led&results=238*](https://ardushop.ro/ro/electronica/299-led-5mm.html?search_query=led&results=238)

**Active Buzzer –** [*https://www.sigmanortec.ro/Buzzer-activ-5v-p126421597?gad\_source=1*](https://www.sigmanortec.ro/Buzzer-activ-5v-p126421597?gad_source=1)

**4-Digit Display -** [*https://roboromania.ro/produs/modul-4-digit-led-display-rosu-comandat-de-driverul-tm1637/*](https://roboromania.ro/produs/modul-4-digit-led-display-rosu-comandat-de-driverul-tm1637/)

**7. Resurse**

[*https://arduinogetstarted.com/tutorials/arduino-water-sensor*](https://arduinogetstarted.com/tutorials/arduino-water-sensor)

[*https://lastminuteengineers.com/water-level-sensor-arduino-tutorial/*](https://lastminuteengineers.com/water-level-sensor-arduino-tutorial/)

[*https://lastminuteengineers.com/tm1637-arduino-tutorial/*](https://lastminuteengineers.com/tm1637-arduino-tutorial/)

[*https://github.com/passion-tech/Hello-tech/blob/master/Water\_Level\_Sensor\_with\_LED.ino*](https://github.com/passion-tech/Hello-tech/blob/master/Water_Level_Sensor_with_LED.ino)

[*https://www.youtube.com/watch?v=-HCZY4UoFiA*](https://www.youtube.com/watch?v=-HCZY4UoFiA)