Al-Balqa' Applied University
Faculty of Engineering Technology

Electrical Engineering Department
Final Project:
Embedded Systems

Project Topic:
Water Level Detection

GROUP MEMBERS:

ليث قاسم مفيد عمرو

INSTRUCTOR: Eng.Mazen Abu Zaher

Lecture Time: SUN, TUE, THU 12:30—1:30PM

DATE: THURSDAY 2 / 1 / 2025

Water Level Project

❖ Introduction

*1. **Objectives:***

The primary objective of this project is to design a water level monitoring system using an ultrasonic sensor. The system measures the water level in a tank and displays the value on an LCD screen.

*2. **Overview:***

This project employs a PIC16F887 microcontroller to measure the distance to the water surface using the HC-SR04 ultrasonic sensor. Based on the distance, it calculates the water level and displays it on an LCD. The water tank's height is defined such that the empty tank corresponds to 200 cm, and the full tank corresponds to 5 cm. This system is useful for managing water usage in residential, agricultural, or industrial applications.

❖ System Design

*1. **Block Diagram:***

a. Input: Ultrasonic Sensor (HC-SR04).
b. Processing: PIC16F887 microcontroller.
c. Output: LCD screen displaying the water level.

2. ***Circuit Diagram:***

The circuit schematic was designed and simulated in Proteus. The key components include:

a. PIC16F887 microcontroller.
b. HC-SR04 ultrasonic sensor.
c. 16x2 LCD.
d. Supporting components (crystal oscillator, resistors, capacitors).

Water Level Project

❖ Components Used

1. ***Hardware:***
   a. **PIC16F887 Microcontroller: Processes distance data and controls the LCD output.**
   b. **Ultrasonic Sensor (HC-SR04): Measures the distance to the water surface by emitting ultrasonic pulses and calculating the time for the echo to return.**
   c. **16x2 LCD: Displays the water level in percentage or distance units.**
   d. **Crystal Oscillator (8 MHz): Provides a stable clock signal for the microcontroller.**
   e. **Resistors (330 Ω, 10 kΩ): Used for current limiting and pull-up configurations.**
   f. **Capacitors (22 pF): Used for stabilizing the crystal oscillator.**
2. ***Software:***
   a. **Proteus 8 Pro: For circuit simulation and testing.**
   b. **mikroC Pro for PIC: For writing and compiling embedded C code.**

Water Level Project

❖ Working Principle

1. ***Water Level Measurement:***
    a. The HC-SR04 sensor emits ultrasonic pulses and calculates the time taken for the echo to return.
    b. The microcontroller converts this time into a distance measurement.
    c. The system defines the full tank level at 5 cm and the empty tank level at 200 cm.
2. ***Microcontroller Processing:***
    a. The distance is compared to the tank's maximum height (200 cm) and minimum height (5 cm) to calculate the water level as a percentage.
3. ***Display on LCD:***
    a. The calculated water level is displayed on the LCD in real-time, showing the level as a percentage, ranging from 0% (empty tank at 200 cm) to 100% (full tank at 5 cm).

Water Level Project

❖ Implementation

1. Code Description:

a. The system uses the HC-SR04 sensor to measure the distance to the water surface.

b. The microcontroller processes this distance and calculates the water level as a percentage.

c. The LCD is used to display the water level on the screen in real-time, providing a clear visualization of the current water level in the tank.

Water Level Project

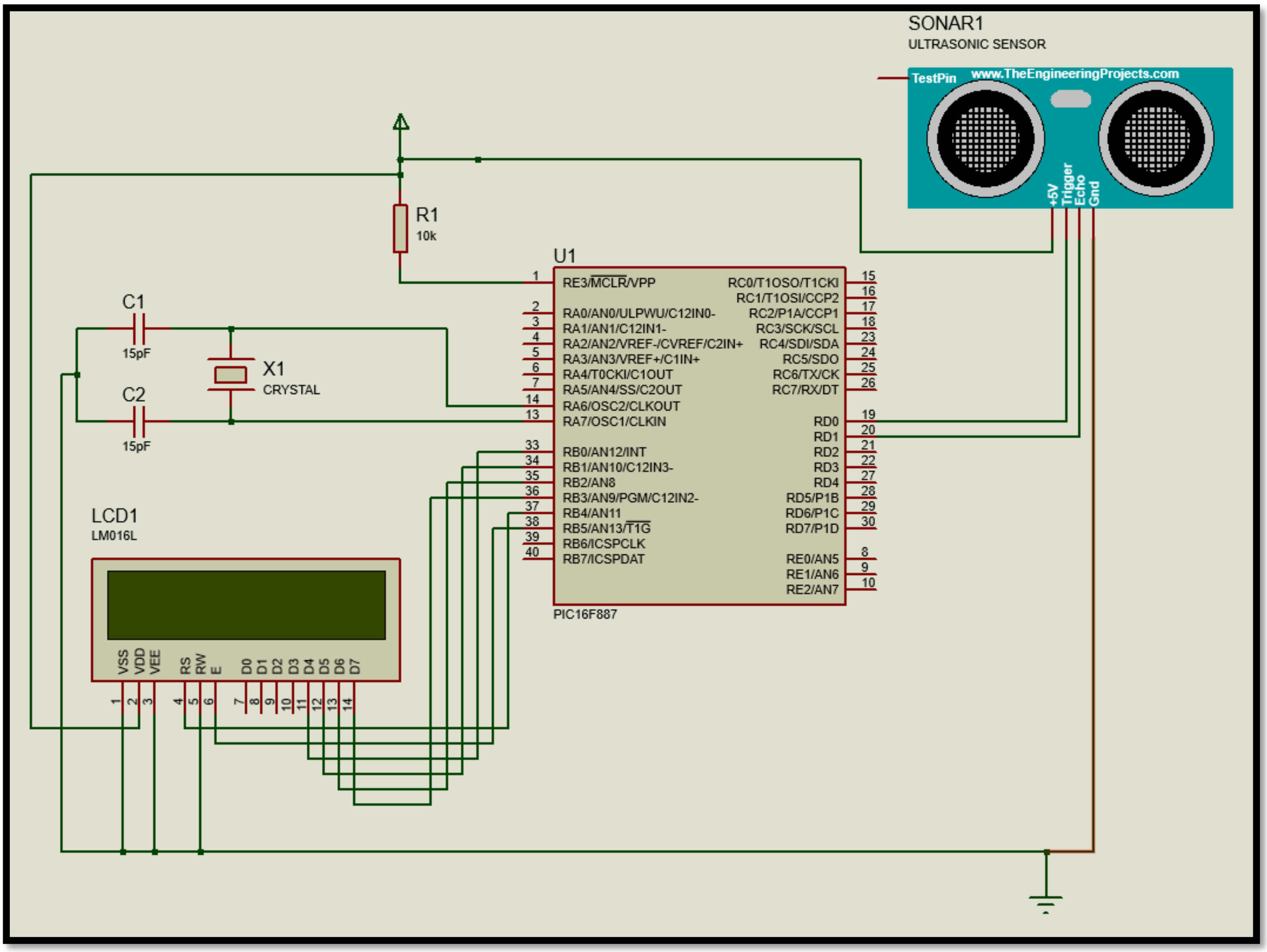
2. Sample Code Snippet:

```
#LCD_instructions.
#define MAX_HEIGHT 200   // Maximum height of the tank in cm (empty tank = 0% at 200 cm)
#define MIN_HEIGHT 5     // Minimum height of the tank in cm (full tank = 100% at 5 cm)
void main() {
    float D, t;
    int distance;
    ansel = 0;
    anselh = 0;

    Lcd_Init();  // Initialize LCD
    Lcd_Cmd(_LCD_CLEAR);  // Clear LCD screen
    Lcd_Out(1, 5, "Water Level:");  // Center "Water Level:" on line 1 (start at position 5)
    Lcd_Out(2, 5, "Level: ?");      // Center "Level: ?" on line 2 (start at position 5)

    delay_ms(2500);  // Delay to show the initial messages

    TRISD = 0b00000010;  // RD0 (Trigger) as output, RD1 (Echo) as input

    OPTION_REG = 0b00000110;  // Prescaler 1:128
    TMR0 = 68;                // Initial count from 68

    while(1) {
        PORTD.F0 = 1;    // Set RD0 high (Trigger)
        Delay_us(10);     // Wait for 10 microseconds
        PORTD.F0 = 0;    // Set RD0 low (Trigger)

        while(PORTD.F1 == 0);  // Wait for echo to go high

        TMR0 = 68;   // Clear Timer0 register before starting to measure time

        while(PORTD.F1 == 1);  // Wait for echo to go low

        t = (128.0 * (TMR0 - 68)) / 1000000.0;  // Time in microseconds (TMR0 is 8-bit timer)

        D = (t * 39000.0) / 2.0;  // Distance (speed of sound = 39000 cm/s)

        // Check for valid distance and calculate the water level as a percentage
        if (D <= MAX_HEIGHT && D >= MIN_HEIGHT) {
            distance = ((MAX_HEIGHT - D) * 100) / (MAX_HEIGHT - MIN_HEIGHT);  // Valid distance
        } else if (D > MAX_HEIGHT) {
            distance = 0;  // Tank is empty or distance is out of range
        } else if (D < MIN_HEIGHT) {
            distance = 100;  // Tank is full
        } else {
            distance = -1;  // Invalid distance, set to -1 to indicate no valid reading
        }

        Lcd_Cmd(_LCD_CLEAR);  // Clear the LCD before updating the display
        Lcd_Out(1, 5, "Water Level");  // Center "Water Level" on line 1
        if (distance == -1) {
            Lcd_Out(2, 5, "Level: ?");  // Center "Level: ?" on line 2
        } else {
            Lcd_Out(2, 5, "Level: ");  // Display "Level: " on line 2
            Lcd_Chr(2, 12, (distance / 10) + '0');  // Tens digit of the distance (percentage)
            Lcd_Chr(2, 13, (distance % 10) + '0');  // Ones digit of the distance (percentage)
            Lcd_Chr(2, 14, '%');  // Display percentage symbol
        }

        Delay_ms(1000);  // Delay before the next measurement
    }
}
```

Water Level Project

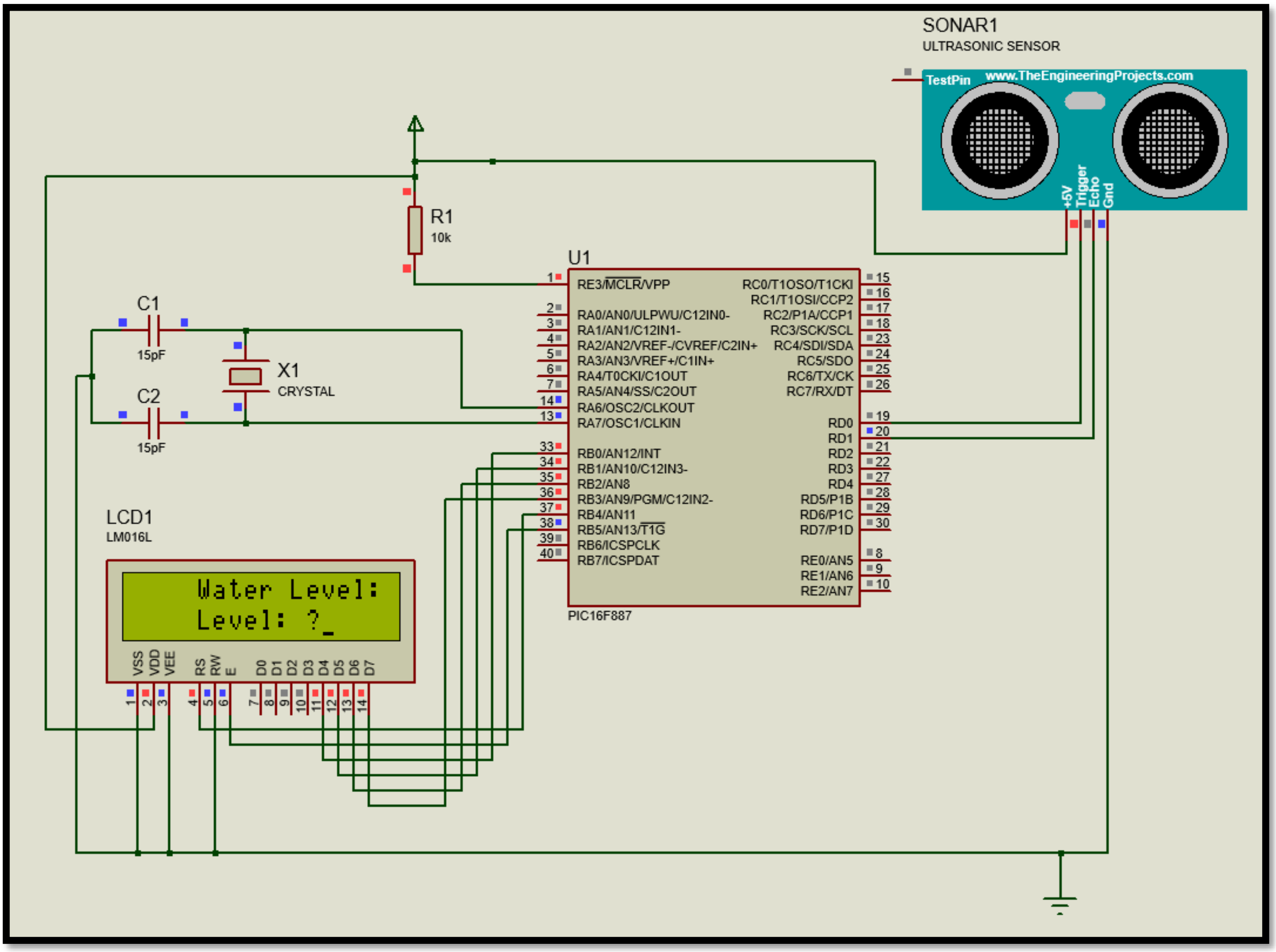
3. Simulation and Testing:

The system was tested in Proteus by simulating the water level and ultrasonic sensor input signals. The LCD correctly displayed the water level, and the microcontroller processed the distance data as expected. The system was able to calculate the water level percentage and display it in real-time. The circuit configuration was verified, and the system showed accurate readings based on the simulated input, confirming its functionality.



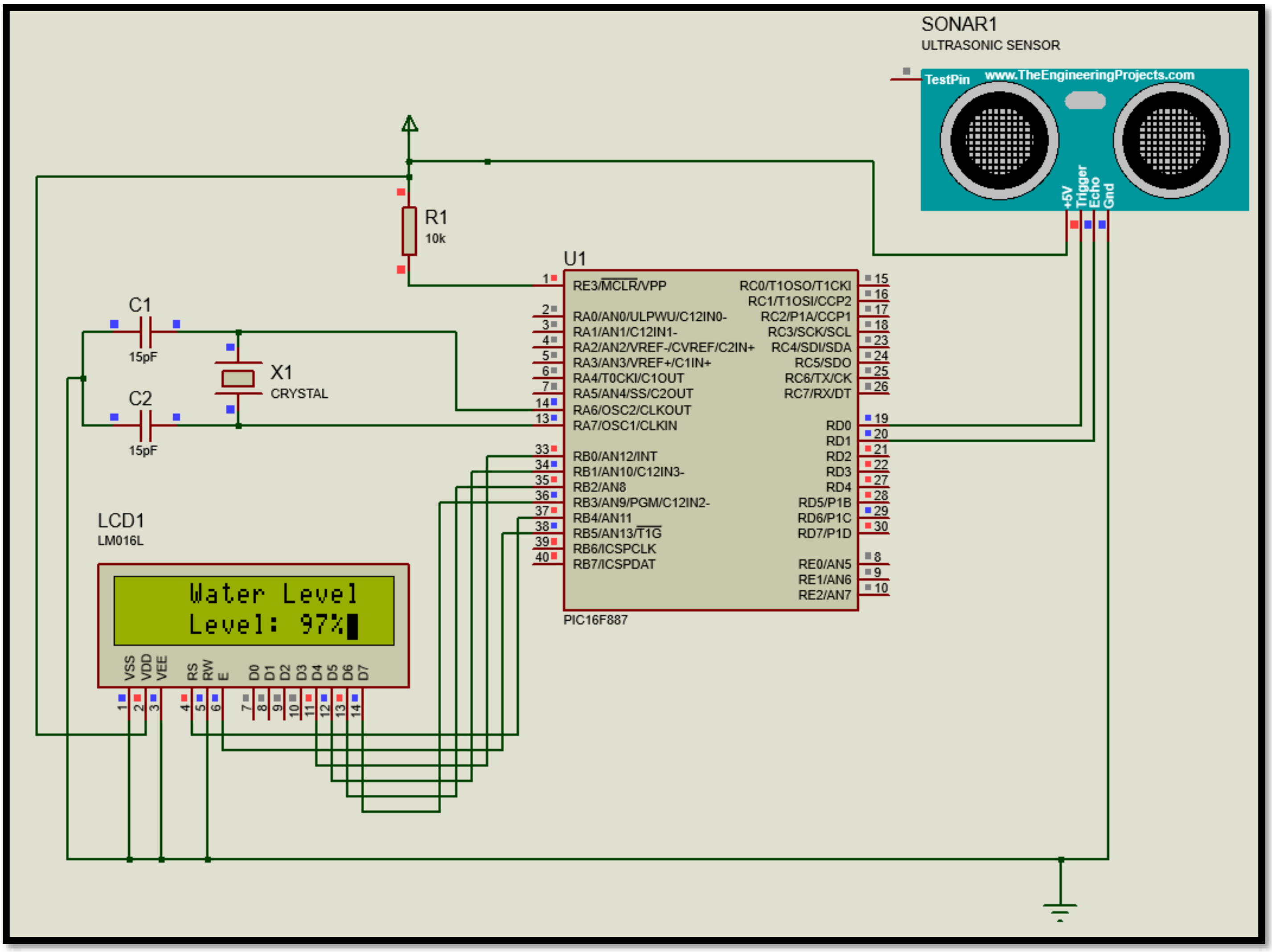
Water Level Project

- **Firstly, when the PIC is on, and the sensor doesn't send any echo.**



Water Level Project

- Secondly, when sensor is given back the echo the LCD shows the percentage of water in the Tank.



Water Level Project

❖ Applications

1. ***Water management** in residential or commercial tanks.*
2. ***Monitoring water levels** in agricultural fields.*
3. ***Industrial applications** requiring precise water level monitoring.*

❖ Challenges and Solutions

1. ***Challenges:***
   a. **Accurate calibration of the HC-SR04 sensor to handle environmental factors.**
   b. **Display synchronization on the LCD.**
2. ***Solutions:***
   a. **Conducted multiple tests to calibrate the sound sensor for appropriate sensitivity.**
   b. **Used precise timing and LCD commands for consistent display updates.**

Water Level Project

❖ Conclusion

The water level monitoring system successfully met its objectives of accurately measuring and displaying water levels. This project demonstrated effective integration of hardware and software, with clear LCD output of real-time water level data. The system defines the full tank level at 5 cm and the empty tank level at 200 cm. The system has potential for further enhancements such as wireless communication or IoT integration.

❖ References

1. PIC16F887 Datasheet
2. Proteus Simulation Software Documentation
3. MikroC Programming Tutorials

- This report was written by Laith Qasem Amro for Embedded Systems course (First semester 2024-2025).

- Instructor : Eng.Mazen Abu Zaher