Obstacle Detection Using Ultrasonic Sensor with PIC Microcontroller

Introduction:

Our project will detect any obstacle in front of it and alert via a buzzer. We implemented it using a microcontroller. In this project, we use an ultrasonic sensor which is capable of detecting an obstacle and gives us the signal. It can be measured the distance between the sensor and the obstacle but here, we are dealing with the detection part only. We used the ultrasonic sensor to interface with the PIC Microcontroller.

Equipment:

1. PIC Microcontroller 16F877A.

2. Ultrasonic Sensor HC-SR04.

3. ARDUINO UNO.

4. Buzzer.

5. Breadboard.

6. 8 MHz Crystal Oscillator.

7. Jumper Wires.

Ultrasonic Sensor HC-SR04: HC-SR04 Ultrasonic sensor is a popular and low cost solution for non-contact distance measurement function. It is able to measure distances from 2cm to 400cm. It has two sensor portion. One is used for transmitting the wave and another one is used for receiving the wave. The transmitter emits an US wave at a frequency of 40Hz, this wave travels through the air and gets reflected back when it senses an object. The returning waves are observed by the receiver. HC-SR04 module has 4 pins and they are VCC, Trigger, Echo and Ground. The VCC pin is used for the power supply to the sensor and work with 5V because the sensor use less current than 15 amp. The ground pin is basically connected with the ground part of the device which we want to interface with it. The trigger pin works as an input pin to the sensor. We need to give 10 microsecond pulse to start the ranging. After this, the module emits ultrasonic waves in pulse of 8 cycles at a frequency of 40,000 Hz. At the same time, the module makes the echo pin high, once the wave get reflected by the object and receive signal by the receiver, at that time the echo pin will be low. The echo pin works as an output pin for the ultrasonic sensor. This is functionalities of Ultrasonic Sensor HC-SR04.

Estimated Cost:

|  |  |
| --- | --- |
| **Component** | **Price** |
| PIC16F877A | 182.8 Taka |
| HC-SR04 Ultrasonic Sensor | 89.9 Taka |
| Breadboard(Big) | 84.9 Taka |
| Jumper Wires (M to M) | 29 Taka |
| Jumper Wires (M to F) | 29 Taka |
| Ceramic Capacitor (22 pF) | 2.37 Taka |
| 33K Ceramic Capacitor | 6.90 Taka |
| 0.1 UF 50V Capacitor | 9.9 Taka |
| Multi-Color LED | 5.6 Taka |
| Red LED | 2.62 Taka |
| Green LED | 2.62 Taka |
| 16 MHz Crystal Oscillator | 16.2 Taka |
| 20 MHz Crystal Oscillator | 15.7 Taka |
| 40 MHz Crystal Oscillator | 13.8 Taka |
| 8 MHz Crystal Oscillator | 13.8 Taka |
| 150 Ohm Resistor | 4.6 Taka |
| 1K Ohm Resistor | 5 Taka |
| 10K Ohm Potentiometer | 13.8 Taka |
| 10K Ohm Resistor | 5 Taka |
| Buzzer(5V) | 16.1 Taka |
| 0.22 uF(50V) | 11.7 Taka |
| **Total** | **561.31 Taka** |

Code:

// LCD module connections

sbit LCD\_RS at RD2\_bit;

sbit LCD\_EN at RD3\_bit;

sbit LCD\_D4 at RD4\_bit;

sbit LCD\_D5 at RD5\_bit;

sbit LCD\_D6 at RD6\_bit;

sbit LCD\_D7 at RD7\_bit;

sbit LCD\_RS\_Direction at TRISD2\_bit;

sbit LCD\_EN\_Direction at TRISD3\_bit;

sbit LCD\_D4\_Direction at TRISD4\_bit;

sbit LCD\_D5\_Direction at TRISD5\_bit;

sbit LCD\_D6\_Direction at TRISD6\_bit;

sbit LCD\_D7\_Direction at TRISD7\_bit;

// End LCD module connections

void main()

{

Lcd\_Init();

Lcd\_Cmd(\_LCD\_CLEAR); // Clear display

Lcd\_Cmd(\_LCD\_CURSOR\_OFF); // Cursor off

TRISB0\_bit=0; //Set RB0 as output

TRISB4\_bit=1; //Set RB4 as input

TRISA0\_bit=0; //Set RA0 as output for the BUZZER/LED

RA0\_BIT = 1; //Making the Buzzer off

Lcd\_Out(1,1,"Obstacle");

Lcd\_Out(2,1,"Detector");

Delay\_ms(2000);

Lcd\_Cmd(\_LCD\_CLEAR);

while(1)

{

PORTB.F0 = 1; //TRIGGER HIGH

Delay\_us(10); //10uS Delay for the wave to generate

PORTB.F0 = 0; //TRIGGER LOW

if(PORTB.F4 = 0) //Check if the ECHO is off(meaning) wave has recieved

{

Lcd\_Cmd(\_LCD\_CLEAR);

Lcd\_Out(1,1,"Obstacle!!!");

RA0\_BIT = 0; //Buzzer ON

}

else

{

Lcd\_Cmd(\_LCD\_CLEAR);

Lcd\_Out(1,1,"No Obstacle");

RA0\_BIT = 1; //Buzzer Off

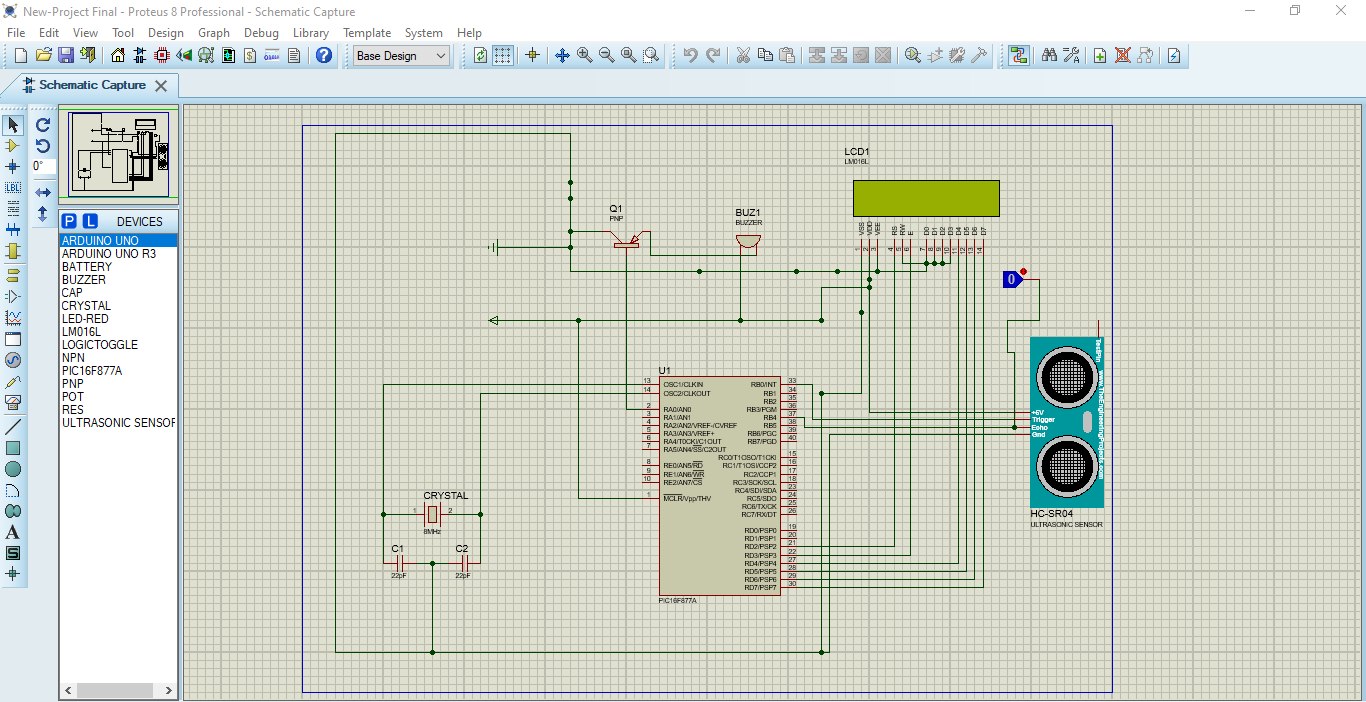
}

Delay\_ms(400);

}

}

Simulation diagram:



Discussion:

In this part, we want to discuss what we had done with the project. First, we were working with the simulation portion. In the simulation portion, we implemented the MCU, Sensor, LCD display, Oscillator and Capacitor successfully. But our main goal is to connect the buzzer which will indicate us. At first, we connected the buzzer’s VCC pin with the pin no. 37 of the MCU, but the buzzer didn’t working properly. So, we used a PNP transistor which would fixed the problem. We also connected the LCD display with the microcontroller and sensor. We used a POT to reduce the brightness of the LCD but there are some issues with the POT, that’s we remove the POT from the simulation. We connected the VEE port directly onto the ground. There was no option to reduce the brightness.

We burn the code in the MCU using the pickit 3 software. I did not face any problem at time burning. After burning, we were going to implement the project on the breadboard. Then, we took a breadboard. Then we put the PIC Microcontroller 16F877A and gave the connection of the MCU with the VCC. We used ARDUINO UNO as a power supply for the circuit. We connected the pin no.1 of the MCU with the ARDUINO UNO’s 5V VCC connection.

Next, we connected the pin no. 13, 14 with a crystal oscillator of 8MHz. The Oscillator works as a clock. Next, we connected the Ultrasonic Sensor with the MCU to interact with each other. As, we know that there are four pins in the sensor. We power up the sensor after connecting VCC. We also connected the GND pin with the common ground of the breadboard. Then, I connected the pin no.33 of the MCU with the trigger pin of the Ultrasonic Sensor. I also connected the pin no. 37 of the MCU with the Echo pin of the Sensor. The MCU established a connection through pin no.33 with Trigger pin which produce 10 microsecond pulse for ranging. After finding any obstacles, the Echo pin produced a signal as an output. So, we need to show the output signal, that’s why we used a buzzer for visualizing the signal.

Next, we took a buzzer and connected one portion with the pin no. 37 of MCU and another portion was connected with the common ground. We also connected the 16\*2 LCD display with the microcontroller to see the output but here we faced some issues. In the display, we connected the pin of LCD (D0-D3) with the common ground because we did not need this ports. We connected the pin (D4-D7) of the LCD with the pin of the microcontroller (27-30). We connected the RS, E pin of the LCD with the microcontroller 21 & 22 no. pin. W connected the anode pin of the LCD with VCC and cathode pin with the common ground. We took a POT which helped us to reduce the brightness of the LCD and helped us to consume less power. I connected the VDD pin of the LCD with the VCC pin of the POT and VSS pin of the LCD with the ground pin of the POT. The Connection of the POT was going to the common VCC and another portion on the common ground. After connecting the POT, the display didn’t work properly, so we removed the POT and tried to use the display directly. The display was light up but there were nothing output on the display. We also tried with the buzzer but the buzzer was not working properly. It would produce some error signal.