International Journal of Computer
Science and Engineering (IJCSE)
ISSN(P): 2278-9960; ISSN(E): 2278-9979
Vol. 8, Issue 1, Dec - Jan 2019; 1-12

International Academy of Science, Engineering and Technology
Connecting Researchers; Nurturing Innovations

SONIC SENSEINTERFACING HC04 SONIC SENSOR WITH PIC16F452 TO ASSIST VISUALLY DISABLED PERSON NAVIGATE THROUGH AN OBSTACLE COURSE.

Sneha Kale[1] & Omkarkachare[2]

[1]Research Scholar, Computer Science, California State University,
Sacramento, California, United States of America
[2]Research Scholar, Electronics and Electrical Engineering,
California State University, Sacramento, California, United States of America

ABSTRACT

This project is designed to guide a visually impaired person to walk and avoid bumping into obstacles. Low-cost ultrasonic rangefinders along with a microcontroller are used to measure the distance to obstacles and if they are close enough to provide feedback to the user in form of beeps or vibrations.

The project is made on a small single layer PCB. It is a portable project which is mounted on a metal plate such that it can be fixed on any belt type structure. The sensors are not mounted on the PCB but they are mounted in the front of the plate and connected to the main board using wires. It's designed exactly like a mobile carrying case used on belts.

KEYWORDS: *Visually Impaired Person, Interfacing with Microcontroller, Microcontroller and Echo*

Article History

Received: 02 Jan 2019 | Revised: 08 Jan 2019 | Accepted: 12 Jan 2019

INTRODUCTION

Background

In order to understand project background so we will see first the history of HC-SR04, this is the main sensor of our project. For sensing distance to obstacles, an HC-SR04 sensor module has been used. It has an ultrasonic transducer which generates the ultrasonic waves, an ultrasonic receiver and control circuitry built on a small PCB.

For interfacing with microcontroller, it provides two lines namely TRIGGER and ECHO. The trigger pin is an input pin; the MCU sends a 10uS high pulse on this line to tell the HC-SR04 to start a taking a measurement.

As soon as the HC-SR04 receives this pulse it sends out ultrasonic waves and waits for it to go to the obstacle and come back to the sensor. The sensor then emits a pulse on the ECHO line whose width is equal to this time. By simple calculation, we can find the distance to the obstacle.

Two such sensors are used in this project to find an obstacle in front and downward of the user.

BLOCK DIAGRAM

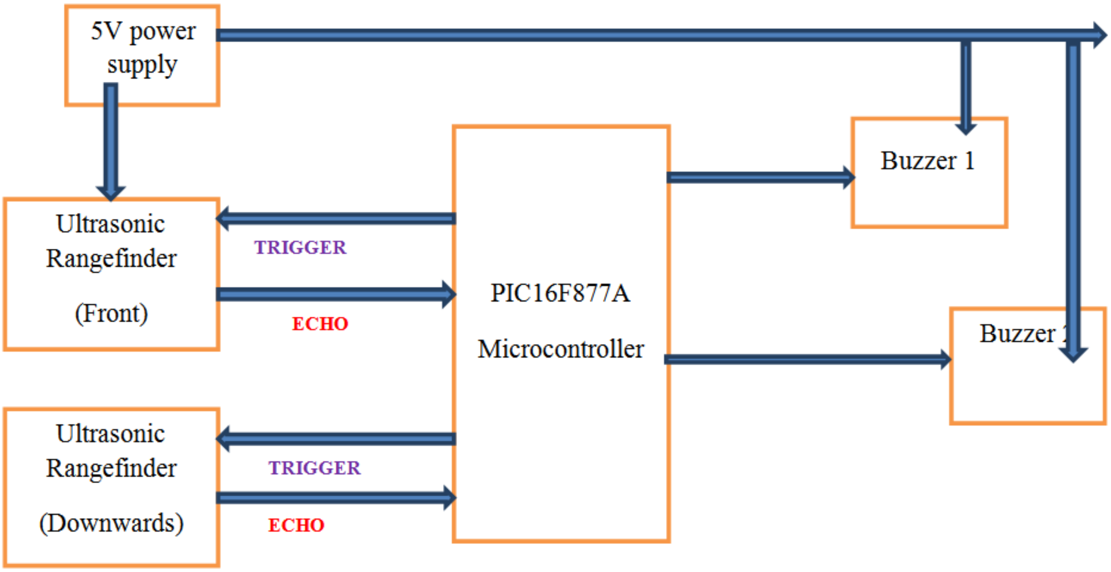

Figure 1: Block Diagram Description

Microcontroller

We've used PIC16F877A in this project. Trigger given as input to the sensor by the microcontroller and Echo is the input to the microcontroller from the sensor that is used to sense the obstacle. Timer1 is used as a counter which is initialized to 0 as soon as the trigger is detected. It starts incrementing unless and until an echo is detected.

Ultrasonic Sensor

Ultrasonic ranging module HC - SR04 provides 2cm - 400cm non-contact measurement function, the ranging accuracy can reach to 3mm. The modules include ultrasonic transmitters, receiver and control circuit. The basic principle of work:

- Using IO trigger for at least 10us high-level signal,
- The Module automatically sends eight 40 kHz and detect whether there is pulse signal back.
- IF the signal back, through high level, time of high output IO duration is the time from sending ultrasonic to returning.

Wire Connecting Directly as Follows

- 5V Supply
- Trigger Pulse Input
- Echo Pulse Output
- 0V Ground

Buzzer

Buzzer and LED's are used as obstacle indicator in the circuit. As soon as an obstacle is detected buzzer is activated.

ELECTRONICS AND HARDWARE DESIGN ASPECTS

PIC Selection

Table 1: Specifications of PIC Family Microcontroller

Parameter	PIC16F877A	PIC18F4520
Pin Count	28/40	28/40/44
Operating voltage	2V – 5.5V	2V – 5.5V
Operating frequency	0 to 20MHz	0 to 40MHz
Inbuilt ADC	8	13
ADC bits	10	10
No of timers(8/16)	3/0	1/3
Program Memory	8K	32K
Data memory	368(SRAM),256(EEPROM)	1536(SRAM),256(EEPROM)

- We require only 1 Timer i.e. TIMER1 for our sensor interfacing of 8 bit.
- Operating voltage up to 5V.
- We needed program memory as less as possible.
- We needed digital output for the sensor.
- The cost of the PIC should be least.
- Hence we have selected as PIC16F877A as our projects brain.

RESET CIRCUIT

We are connecting an RC circuit to the MCLR (pin1) of the microcontroller (16f877A). The resistor and the capacitors values from the datasheet are 1kohm and 10uF respectively. The resistor-capacitor combination gives the RC time delay for the microcontroller for it to operate properly. The PIC has an active low reset, the capacitor initially at 0V charges via the supply through a –ohm resistance in series. Therefore the reset time of our circuit is:

R*C=1K*0.1*10^-6=0.1msec

Crystal Circuit

Here we are connecting two capacitors which are basically of values 15pF each. In other words to give a pure square wave to the microcontroller. A basic rule for placing the crystal on the board is that it should be as close as possible to avoid any interference in the clock. The crystal we used is of 8MHZ.

POWER BUDGET

- Pic16F877A=25mA
- Buzzer=30mA

- Led=30mA
- Sensor=40mA
- LM7805=50mA
- Total current=17775mA
- VOLTAGE REQUIREMENT=5V
- Total power =0.875W

POWER SUPPLY DESIGN

Our project is designed for visually impaired people so we have given supply through a 9V rechargeable battery. As circuit needed 5V dc regulated supply we supplied 9V regulated dc through a voltage regulator LM7805 which provides an output as fixed 5V supply as per our requirement.

Battery=9V LM7805=5V.

SOFTWARE ASPECTS

PROTEUS

It is software for microprocessor simulation, schematic capture. It is developed by Lab Center electronics. The Proteus design suite includes schematic capture tool with the Possibility to simulate programmable ICs like Microchip PIC, Atmel AVR etc. ARES for PCB layouts.

ALTIUM 15

Altium15 software is used for routing tracks of PCBS. It is useful software as it has many In-built libraries.so we do not have to separately create our new library and draw a component according to its specification on the datasheet. We just have to install the library and use the appropriate components.

MIKRO C PRO

MIKRO C pro is a full-featured ANSI C compiler for PIC devices from Microchip. It is the best solution for developing code for PIC devices. It features intuitive IDE, powerful compiler with advanced optimizations, lots of hardware and software libraries, and additional tools that will help you in your work. The compiler comes with a comprehensive Help file and lots of ready-to-use examples designed to get you started in no time. Compiler license includes free upgrades and a product lifetime tech support, so you can rely on our help while developing.

ALGORITHM

- START.
- Initialize the PORT A.4 as input.
- Initialize the PORT B.4 as input.
- Initialize Timer1 module.
- Set the initial Timer values to 0.

- Give a pulse of 10us or more as a trigger.
- Start the timer as soon as the trigger is detected.
- Start the timer.
- Wait for the Echo to be detected.
- Stop the timer after the echo is detected.
- Read the timer value.
- Convert time to distance.
- Buzz if the distance is less than 100cm.
- Repeat Steps 5-12 for sensor 2.
- Buzz with a delay of 500ms if the distance is less than 250cm.
- Buzz with a delay of 10ms if the distance is less than 100cm.
- LOOP.

Mikro C Code

```
void main()
 {
int a;
int b;
//TRISB.F0=0;        //TRIGGER OP
//TRISB.F6=0;        //BUZZER OP
//TRISA.F5=0;        //BUZZER
TRISA = 0b00010000;      //RA4 as Input PIN (ECHO)
TRISB = 0b00010000;
delay_ms(100);
T1CON = 0x10;        //Initialize Timer Module
while(1)
{
// SENSOR 1
TMR1H = 0;       / /Sets the Initial Value of Timer
TMR1L = 0;       //Sets the Initial Value of Timer
```

```
PORTB.F0 = 1;      //TRIGGER HIGH
delay_us(15);       //10uS Delay
PORTB.F0 = 0;      //TRIGGER LOW
while(!PORTB.F4);    //Waiting for Echo
T1CON.F0 = 1;       //Timer Starts
while(PORTB.F4);    //Waiting for Echo goes LOW
T1CON.F0 = 0;       //Timer Stops
 a = (TMR1L | (TMR1H<<8)); //Reads Timer Value
a = a/58.82;              //Converts Time to Distance
a = a + 1;                //Distance Calibration
if(a>2 && a<400)
 {
delay_ms(10);
if(a<=100)
{
PORTB.F6=1;
delay_ms(3000);
PORTB.F6=0;
delay_ms(50);
}
}
delay_ms(10);
 // SENSOR 2
TMR1H = 0;       //Sets the Initial Value of Timer
TMR1L = 0;       //Sets the Initial Value of Timer
PORTB.F0 = 1;       //TRIGGER HIGH
delay_us(15);       //10uS Delay
PORTB.F0 = 0;       //TRIGGER LOW
while(!PORTA.F4);     //Waiting for Echo
T1CON.F0 = 1;       //Timer Starts
```

```
while(PORTA.F4);      //Waiting for Echo goes LOW
T1CON.F0 = 0;        //Timer Stops
b = (TMR1L | (TMR1H<<8));  //Reads Timer Value
b = b/58.82;         //Converts Time to Distance
 b = b + 1;         //Distance Calibration
if(b>2 && b<400)
{
delay_ms(10);
if(b<=250)
{
PORTA.F5=1;
delay_ms(500);
 PORTA.F5=0;
if(b<=100)
{
 PORTA.F5=1;
delay_ms(10);
PORTA.F5=0;
 } }
}
delay_ms(10);
 }
 }
```

CIRCUIT DESIGN

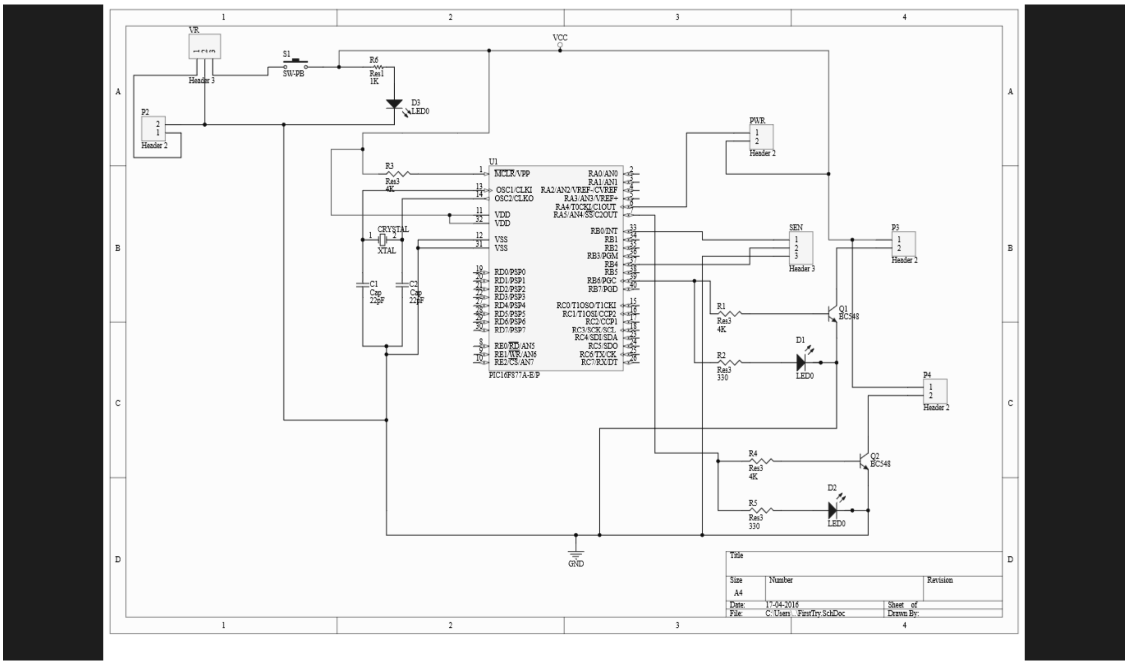

Figure 2: Circuit Diagram

SIMULATION RESULTS

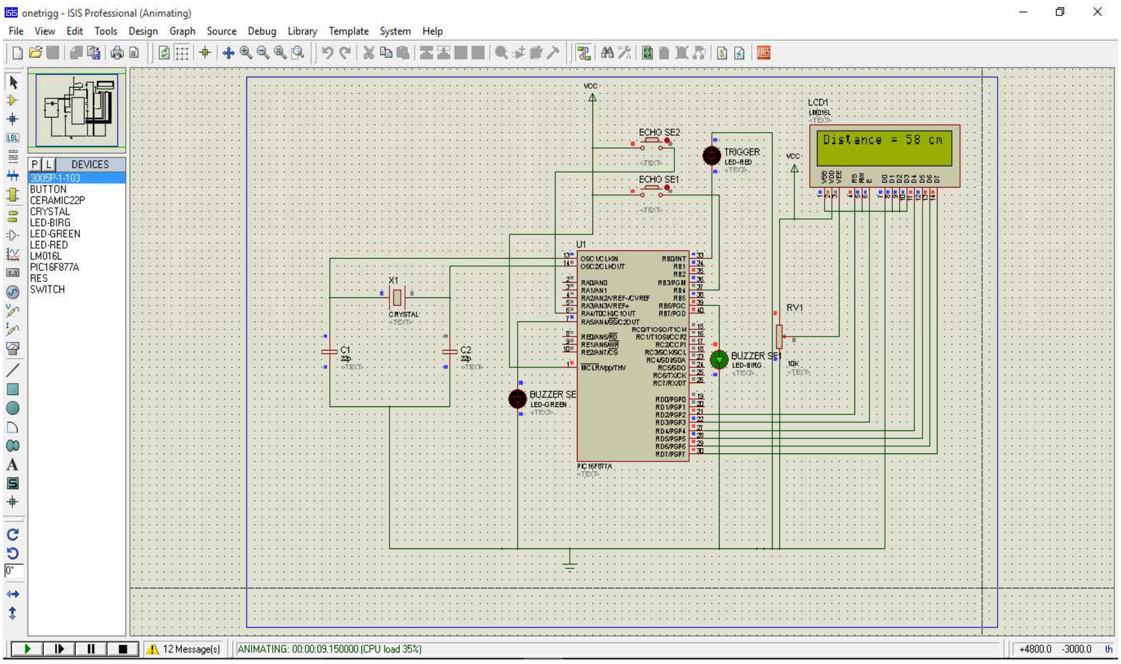

Figure 3: Working Circuit

PCB LAYOUT

In order to design the printed circuit board for our circuit, we have used Altium software. We've only one circuit board which is of our microcontroller. Except for the Sensor, everything is mounted on the same PCB.

Rules for Microcontroller Board

- Clearance=0.5mm
- Track width = Tracks Preferred width
- For VCC and GND 0.8mm
- Other Tracks 0.6mm
- Layer=bottom layer
- Hole size=0.9mm
- PCB size=66 SQUARE CM

Microcontroller Board

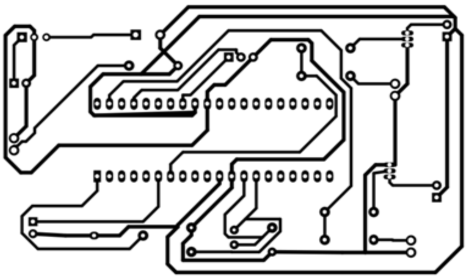

Figure 4: PCB Layout

TESTING OF MODULES

PCB and Soldering Test

We checked the PCB tracks for continuity and track layers. After soldering the components we checked if there is any short circuit scenario.

Microcontroller Board Testing

We checked the PCB for continuity. Then we checked the voltages at point VDD, VCC and MCLR pin. We checked for reset circuitry is it working properly or not? Also, we programmed microcontroller with a test program which blinks the test led on B port. We checked the Sensor test program on PIC board.

HC-SR04 Testing

We tested HC-SR04 on the breadboard for its Working and proper supply to its trigger, Echo, VCC, and Ground terminal.

CABINET DESIGN AND ASSEMBLY

The cabinet mainly consists of the following elements:

- Mounting Plate
- Main PCB

- Sensor
- Batteries
- Buzzers

As this is a portable project we've mounted our PCB and sensors on a metal plate which is bended and carried on a belt so as to get proper sensing of an obstacle.

There is slot provided on the metal plate for the interfacing wires of both the sensors.

PROJECT PHOTOS

Figure 5: Photo 1

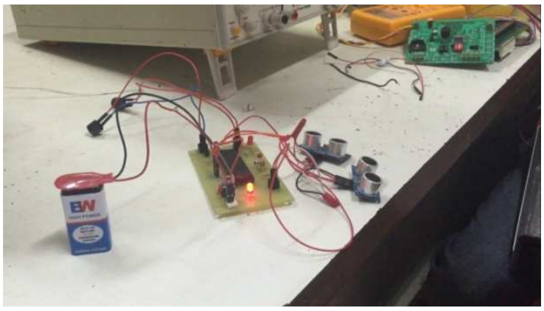

Figure 6: Photo 2

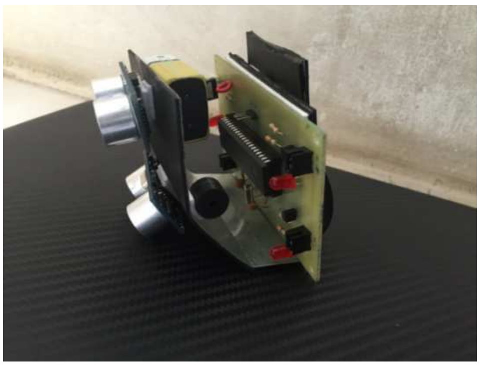

Figure 7: Photo 3

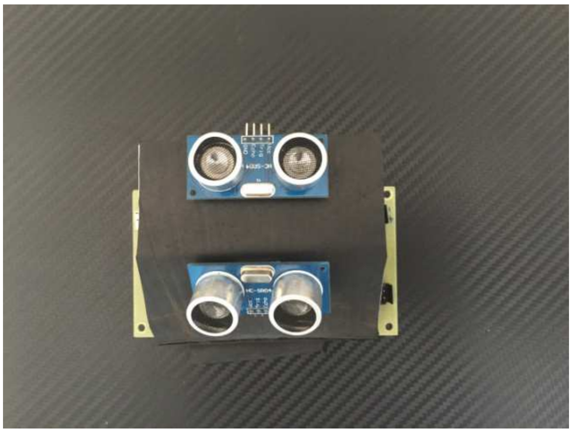

Figure 8: Photo 4

CONCLUSIONS

Thus we conclude that through completion of the project we imbibed a lot of design aspects like PCB design, C programming of sensors, generating footprints for unavailable components.

We came across new concepts like "Trigger" and "Echo" of an Ultrasonic sensor and also the working of the sensor and the importance of components testing.

We have also learned to have an outlook to view the project as a "product" and its development in the mere future.

REFERENCES

1. *"PIC MICROCONTROLLER AND EMBEDDED SYSYTEM"*- *Muhammad Mazidi, Pearson Education, 2nd edition.*
2. *"Datasheet of PIC16F877- http://www.micropik.com/PDF/PIC16F877.pdf"*
3. *"Datasheet of HC-SR04- http://www.micropik.com/PDF/HCSR04.pdf"*
4. *"Datasheet of LM7805- https://www.sparkfun.com/datasheets/Components/LM7805.pdf"*
5. *"Datasheet of BC548- http://www.futurlec.com/Datasheet/Transistor/BC548.pdf"*
6. *http://digitalwizard.net*
7. *http://nevonprojects.com*