

Ultrasound Ranging with Selective Zone

Libo Deng

CCTEG Xi'an Research Institute, Xi'an, China

Email: 122744403@qq.com

How to cite this paper: Deng, L.B. (2024)

Ultrasound Ranging with Selective Zone.
World Journal of Engineering and Technology, 12, 339-361.

<https://doi.org/10.4236/wjet.2024.122022>

Received: March 2, 2024

Accepted: May 10, 2024

Published: May 13, 2024

Copyright © 2024 by author(s) and
Scientific Research Publishing Inc.
This work is licensed under the Creative
Commons Attribution International
License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

Abstract

The technology of ultrasonic measurement has been widely applied in many fields in recent years. It has already been researched for almost 40 years and its application in both residential use and military are mature and well-developed. One of the advanced uses is for reversing a car and it is based on sending sound waves through a specific medium and observing the returning echoes to measure the distance from the device to the obstacle. In this report, a simple ultrasonic distance measurement system will be introduced including the design of hardware and software, the simulating results and the final product. This system was built to be cheaper, requires less power and delivers better performance.

Keywords

Ultrasound, Selective Zone

1. Introduction

With the rapid development of automobile industry, especially in recent years, the world has begun to step into the era of private cars. As the number of vehicles has increased year by year, the traffic has become increasingly crowded in some places, for example, highways, streets, parking lots and garages. The driver feels worried about the safety of the car and one of the most important saturation is the reversing. Due to the driver cannot know and see clearly the distance from the end of the car to barrier which can be a wall, stone, pedestrian or animal, lots of accidents happen every year. A research shows that there are approximately more than 10,000 traffic accidents per year in reversing-car [1]. This problem has been paid more attention by the society and traffic department.

To find a way to solve the problem above, the reversing radar was born. This system can help driver to know how far between the object and the car, and reduce the frequency of accident to a large extent. Most vehicles installed in this

system have a new name “smart car” [2]. The development of “smart car” requires new sensors that are able to measure distances in the range of a few centimeters to a few meters. Parking aids, as well as intelligent suspensions and headlight leveling, are some examples of features that require a distance measurement to be performed with contactless sensors. Several different physical principles can be employed to measure the distance, but price limits greatly restrict the actual choice.

An interesting possibility, which has been investigated by several authors, is the use of ultrasonic sensors based on the well-known time of flight technique [3]. Such sensors are reasonably cheap and work for ranges of up to a few meters, even though problems arise regarding both their accuracy and their behavior in noisy open-air conditions.

In this paper, the author describes a low-cost ultrasonic distance measurement system, which is about design and development of an ultrasound ranging with selective zone. It is concerned with the design of an ultrasound distance measuring system. Two ultrasound transducers will be used to measure the distance from a target, one is for transmit and the other one is for receive. If the measured distance falls with a set range, for example 3.5 - 4 meters, the system will generate an alarm signal and produce an audio signal via a small speaker.

1.1. The Aim of This Project

The objective of this project is that realizing the principle of ultrasound and how to solve the real application using the ultrasound distance measuring system. Indeed, it can be used in a car or alarm when somebody passes or goes through the active range that is set up before. In this project, the microcontroller will be used to control the signal of transmit and receive, also create an alarm signal when the detect distance is lesser than the fixed value which is set by the programming. Consequently, this is a good opportunity to practice the MCU and programming. Moreover, the communicative competence should be improved to a large extent. For example, it is necessary to ask for help from supervisor or classmates in the period of doing the project.

In another word, by completing this individual project, it is a useful training for the further professional career. The main objective of this project is illustrated as below.

- Understanding the operating principle and structure function of MCU in detail.
- Absorb the process of wire arrangement in PCB.
- Grasp the method for testing and correcting the error of circuit.

In a word, the ability of putting the theory applies to the practice will be developed at the end of the project. It is meaningful to lay a solid foundation for job in the future.

1.2. Summary of the Report Structure

This report will describe a simple ultrasonic distance measurement system. It is

divided into 6 main chapters, including introduction, the review of commercial systems, hardware design and software design, the final results and products, and conclusion, respectively. Specifically, the hardware can be separated by 4 parts which are transmit circuit, receive circuit, display circuit and alarm circuit. Meanwhile, the software is using the C language to program the MCU for control the hardware. And then, according to test the circuit, the simulating result will be given. After that, the picture of final product will be pasted on this report.

2. Individual Design

To achieve the requirement of this project, a circuit based on 51 series MCU (This component will be introduced on the next chapter) will be introduced in this part. The basic block diagram of this system is shown in **Figure 1**.

The central MCU is for controlling the drive circuit to transmit a 40 KHz sine wave. This signal will arrive in an object, such as a wall or rocks. Then it would return back to another sensor, which is for receiving. The period of time will be measured by the MCU and calculated the final distance at the same time. After that, the final distance will be displayed on the 7-segment LED displays. If the distance is lower than the range, which is set before, an alarm signal will be produced by the MCU. In addition, due to the temperature could make a little different to the speed of ultrasonic transmission, the actual temperature will also be provided to figure out the distance by temperature measurement circuit. In a word, all of the specific process will work correctly by programming the MCU.

The project contains two main modules, which are hardware circuit design and software design. Indeed, a variety of software should be used for circuit design and test in some situation. For example, the principle circuit can be design by the Multisim, the PCB board can be created by the Protel 99, and the Keil uvision 2 can be used to program the MCU. Meanwhile, the hardware includes transmit circuit, receive circuit, display circuit and alarm circuit. All of the software and hardware will be described more specifically in next 2 chapters.

2.1. The Basic Brief of This Project

The Ultrasonic wave propagation velocity in the air is approximately 340 m/s at

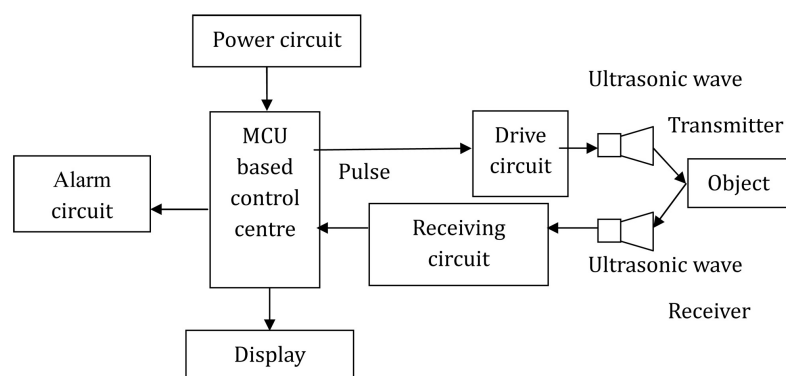


Figure 1. The basic block diagram of this system.

15°C of air or atmospheric temperature, the same as sonic velocity. To be precise, the ultrasound velocity is governed by the medium and its temperature, hence the velocity in the air is calculated using the formula below.

$$V = 340 + 0.6(t - 15) \text{ m/s}$$

t temperature, °C [4].

In this project, a room temperature of 20°C is assumed, thus the velocity of ultrasound in the air is taken as 343 m/s. Due to the travel distance is very short, the travel time is little affected by temperature. For example, it takes approximately 29.15 µsec for the ultrasound to propagate waves through 1 cm distance [5]. Hence, this system has about 1 cm resolution when measuring the distance.

The fixed range will be set approximately 1.0 meter by using the MCU. When the real-distance fall within 1 meter, the speaker that is located in alarm circuit module will make a piece of audio. As the real-distance is gradually close, the frequency of sound is increasing continuously. In other words, taking a car-parking as an example, this different frequency indicates the different gap between the obstruction and the vehicle. It is a useful warning for drivers to realize how close the distance is. Without doubt, the MCU determines this fixed range by using its internal timer (This process will be described in the next chapter).

2.2. The List of Components

All the components which have been used in this project are shown as the **Figure 2**.

Components	Amount	price(£)
AT89C51	1	2.5
Switch	1	1.5
74HC04	1	1.5
7-SEG Digital Tube	4	0.8
12 MHz Crystal Oscillator	1	1
RES 500	8	0.2
RES 1 K	2	0.2
RES 4.7 K	5	0.2
RES 10 k	1	0.2
CAP 3300 pF	1	0.2
Electronic CAP 10 uF	1	0.5
Capacitor 3.3 uF	2	0.2
Ultrasound T	1	1.0
Ultrasound R	1	1.0
CX20106A	1	1.0

Figure 2. All the components.

3. Hardware Design

The detail of the main hardware design would be described in this chapter, including the transmit circuit, receive circuit, display circuit and alarm circuit, respectively. For understanding easily, some components will be introduced if it is necessary. Most of the circuit pictures are picked from the Protel99.

3.1. Transmit Circuit

For generating the transmit ultrasonic signal, the circuit could get a signal waveform by editing the program of MCU and it will work as a driving circuit which is propose to provide a high enough current to drive the ultrasonic wave transmitter. If a hardware circuit will be used to generate this signal, it should be designed as the specified ultrasonic generating circuit with a fixed frequency. This circuit could drive the transmitter directly.

As the **Figure 3** and **Figure 4** shows, the way of using software to generate the

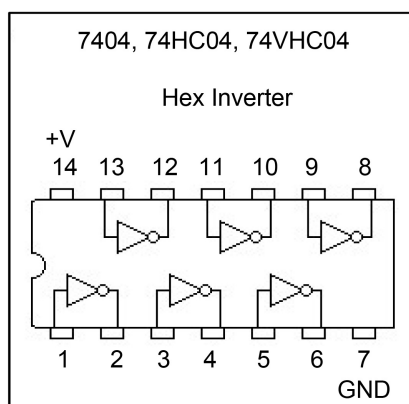


Figure 3. The datasheet of 74HC04.

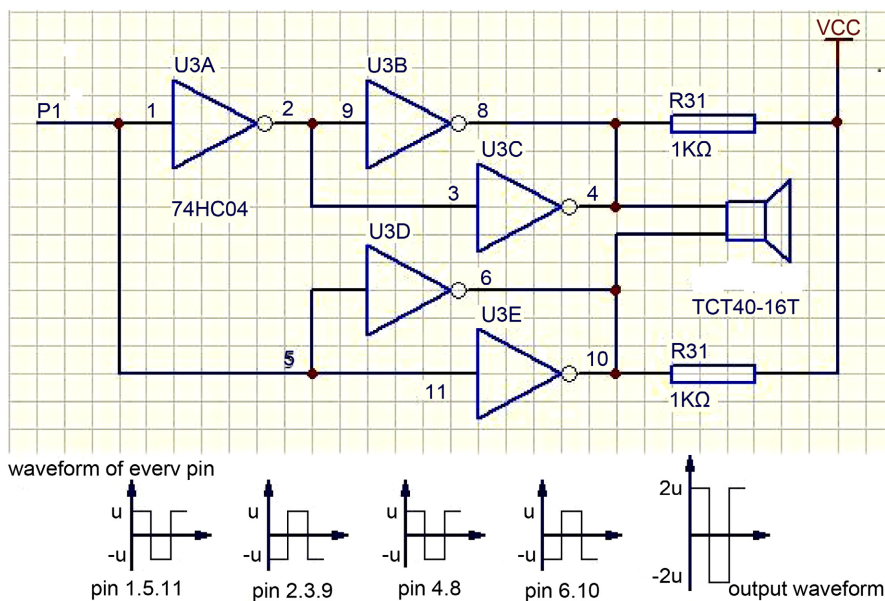


Figure 4. The waveform of pins [7].

signal has the advantage of higher flexibility. So in this project, the P1.0 of AT89c51 will generate a 40 KHz pulse and then go through the 74HC04 [6] to amplify the output waveform. The 74HC04 is a high-speed CMOS inverter with amplification, symmetrical propagation delay and transition time. Compared with the LSTTL logic IC, it has low power consumption and the operation voltage is 2 - 6 V. In this circuit, the output waveform is stable by using the 74HC04.

3.2. Receive Circuit

The receiving circuit consists of three parts: the ultrasonic receiver, the simple filter circuit, the amplifier circuit and the waveform converting circuit. The ultrasonic receiver should be matched with the transmitter for they must be sensitive to a same frequency. Otherwise, the effects of receiver will be influenced.

As the Figure 5 and Figure 6 shows, for the strength of received signal always

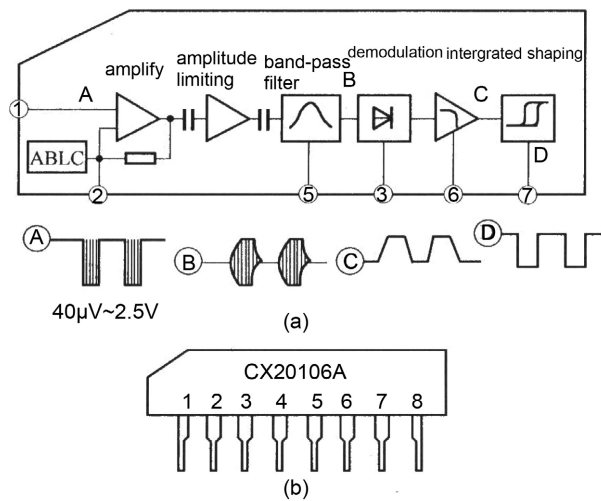


Figure 5. The CX20106A chip.

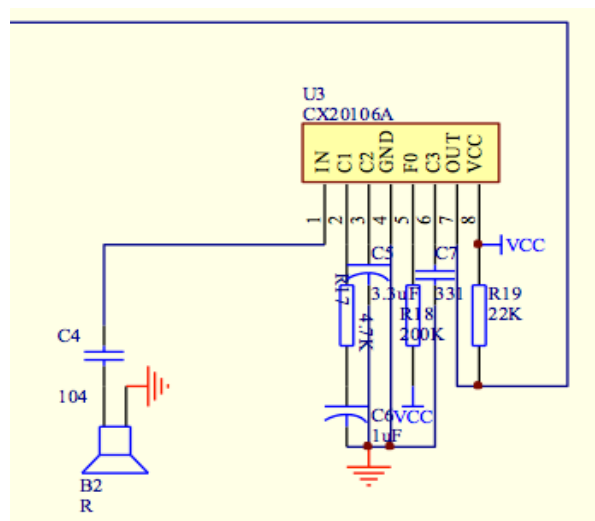


Figure 6. Receiving circuit design.

be very weak, sometimes the voltage is only around 10 mV, it is quite easy to be influenced by the interference signal, especially the 40 KHz electrified wire netting signal. So a simple high-pass filter, which has a 40 KHz pass frequency. In addition, the amplification factor could be accessible to control for it must be changed in diver environment.

A waveform converting circuit is necessary because the received signal is an AC sinusoidal signal while the input of MCU must be a pulse signal. In this situation, The CX20106A [8] is an infrared receiver dedicated chip, commonly used in television infrared remote control. Its carrier frequency is 38 KHz which is more similar to 40 KHz. So, the CX20106A should be a center of the receiving circuit.

3.3. Display Circuit

The common anode seven-segment LED will be used in this circuit and controlled by the MCU. The PNP transistor drive the LEDs, (8550 is used in this design). The drive capability of the I/O port is very small and P0 should be pull-up by the external resistor to increase drive capacity. There is one thing should be noticed, when the voltage level is low, it possibly burn microcontroller due to the current flows into the I/O is too large. So the pull-up resistor is generally 600 Ω to 1000 Ω . The experiments show that the digital tube is bright by using this range of value. Its schematic diagram is shown in **Figure 7**.

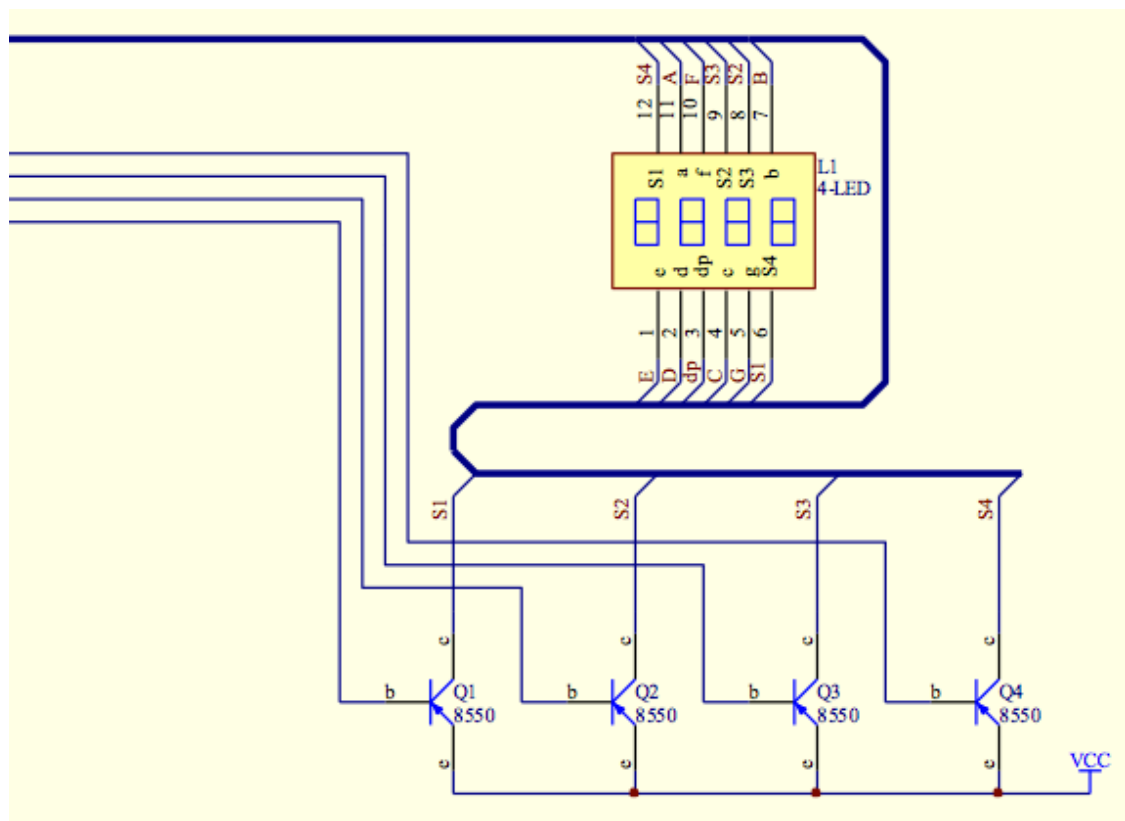


Figure 7. Display circuit design.

3.4. Alarm Circuit

According to the requirements of the project, the system just needs to provide an alarm audio. So, the PNP transistors should be used in driving the speaker. However, the frequency of the audio signal is controlled by the MCU. It has been described that the frequency of the signal will be rise as the distance is close. The circuit design is shown as **Figure 8**.

3.5. The Circuit around AT89C51

The AT89C51 is a low-power, high-performance CMOS 8-bit microcontroller with 4 K bytes of In-System Programmable Flash memory. The device is manufactured using Atmel's high-density nonvolatile memory technology and is compatible with the industry-standard 80C51 instruction set and pinout. The on-chip Flash allows the program memory to be reprogrammed in-system or by a conventional nonvolatile memory programmer. By combining a versatile 8-bit CPU with In-System Programmable Flash on a monolithic chip, the Atmel AT89C51 is a powerful microcontroller which provides a highly-flexible and cost-effective solution to many embedded control applications. **Figure 9** shows the AT89C51 on the software.

3.5.1. The Internal Structure of the Microcontroller

As the **Figure 10** shows, the AT89C51 consists of the following components: central processing unit (CPU), oscillation circuit, the internal bus, program memory and data memory, timer/counter, I/O port, serial port and interrupt system.

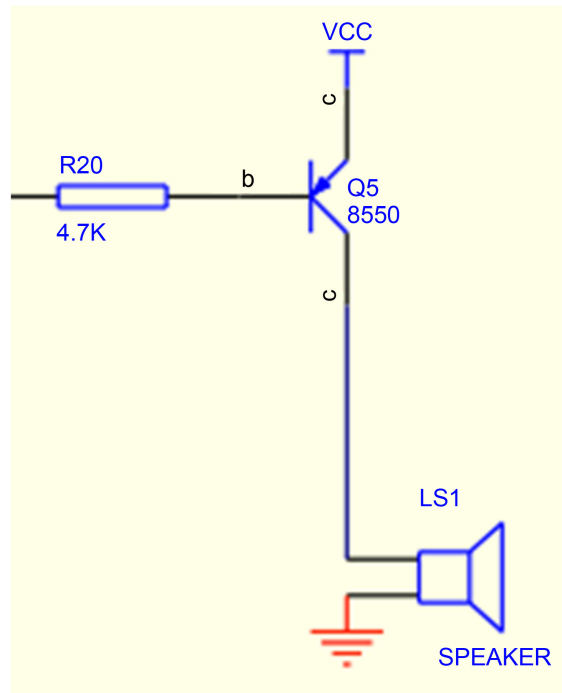


Figure 8. Alarm circuit design.

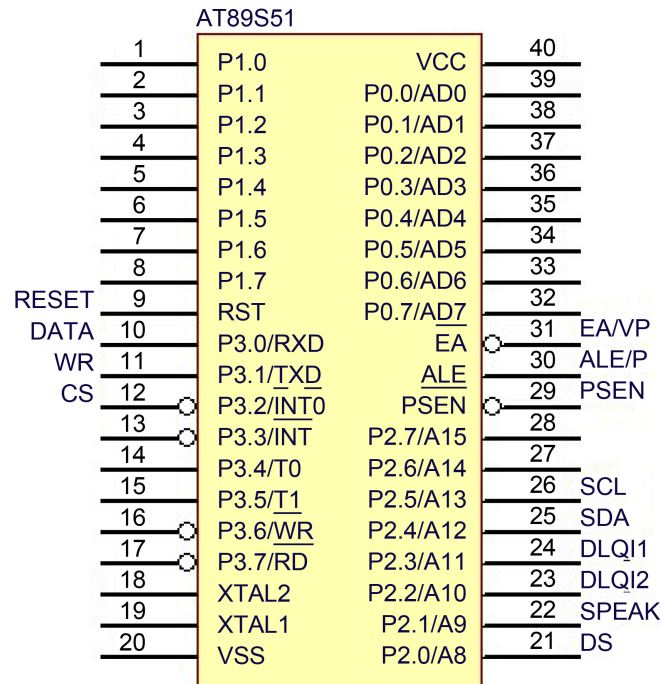


Figure 9. The AT89C51.

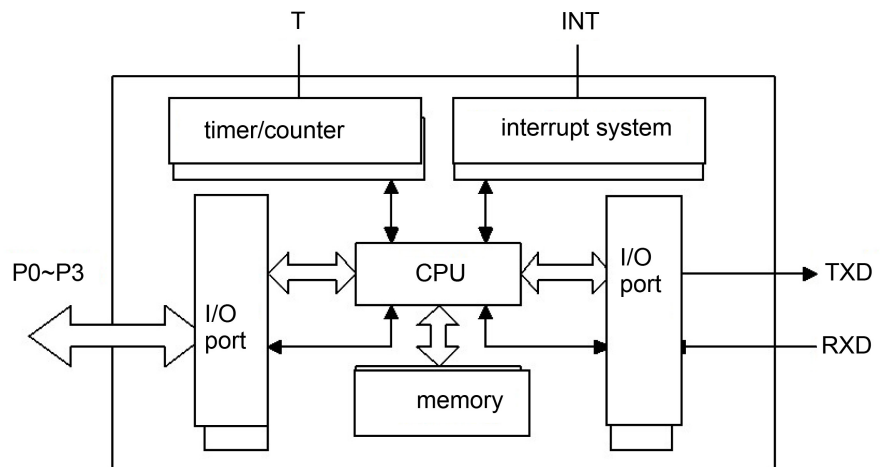


Figure 10. Inside of the AT89C51.

3.5.2. Crystal Oscillator Characteristics

There is a high-gain inverting amplifier used to constitute the internal oscillator. The XTAL1 and XTAL2 are input and output of the inverting amplifier.

External quartz crystal (12.0 MB) and C1, C2 constitute a parallel oscillating circuit connected to the amplifier feedback loop. Although there are not very stringent requirements on external capacitors C1, and C2, if using a quartz crystal, capacitors should be $30 \text{ pF} \pm 10 \text{ pF}$. The circuit is shown in **Figure 11**.

4. Software Design

In this chapter, some softwares which are used in this system will be described

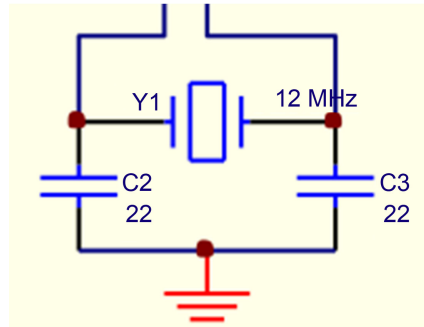


Figure 11. Crystal oscillator characteristics.

and the programming for AT89C51 will be explained. For example, the Protel99 is used in two part, simulation circuit diagram and PCB design.

4.1. The Application of Protel99

4.1.1. Introduction of Protel99

Protel 99 SE—a complete 32-bit electronic design system for Windows and provides a completely integrated suite of design tools that lets us easily take your designs from concept through to final board layout. Some of the powerful features of Protel 99 SE are used in this project. Not only capturing a schematic and running a transient simulation on the design, but also designing and routing a PCB, all with the integrated tools provided in it [9].

With faster logic switching and design clock speeds, the quality of the digital signals becomes more important. Protel 99 SE includes a sophisticated signal integrity analysis tool that can accurately model and analyze your board layout. The signal integrity requirements such as impedance, overshoot, undershoot, and slope are defined as PCB design rules, and then tested during the standard design rule check. If there are nets that you need to analyze in more detail you can select Tools and then Signal Integrity to pass the design to the Signal Integrity Analyzer, where you can perform reflection and cross talk simulations. The results are displayed in an oscilloscope-like waveform analyzer, where you can examine the performance and take measurements directly from the waveforms.

4.1.2. The Schematic Circuit and the PCB Diagram

Figure 12 shows that schematic diagram and final PCB diagram of this project.

For capturing the correct schematic circuit, 4 basic steps should be finished. Firstly, open the Protel99 SE and create a new schematic sheet. Secondly, set the schematic options. For example, the size is A4. Then, draw the schematic. Finding every component needed in the library and connecting them as the design on the schematic diagram. Finally, check the electrical properties of the circuit. In detail, selecting the tools and clicking the ERC button, it will generate an ERC report from the circuit.

For achieving the PCB diagram, there are 7 steps should be completed. It will be described in in order, creating a new PCB file, adding the PCB footprint libraries and transferring the design, defining the layer stack and other

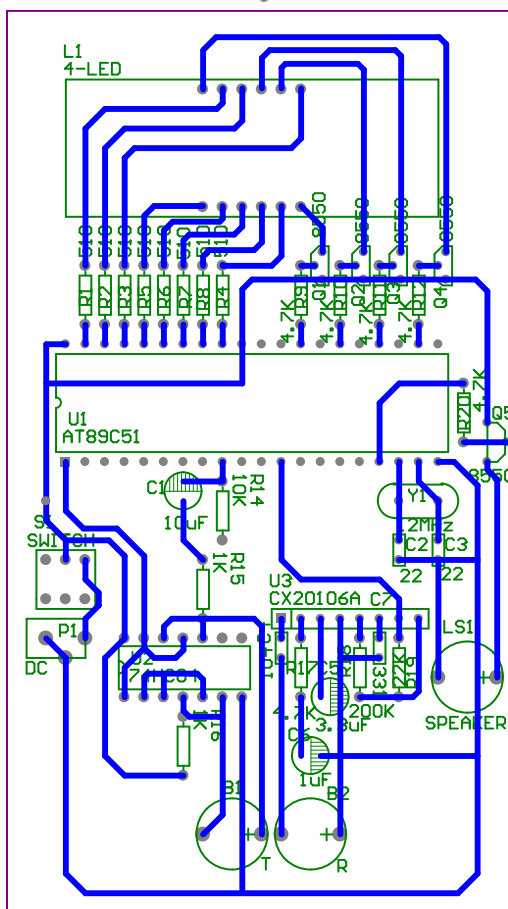
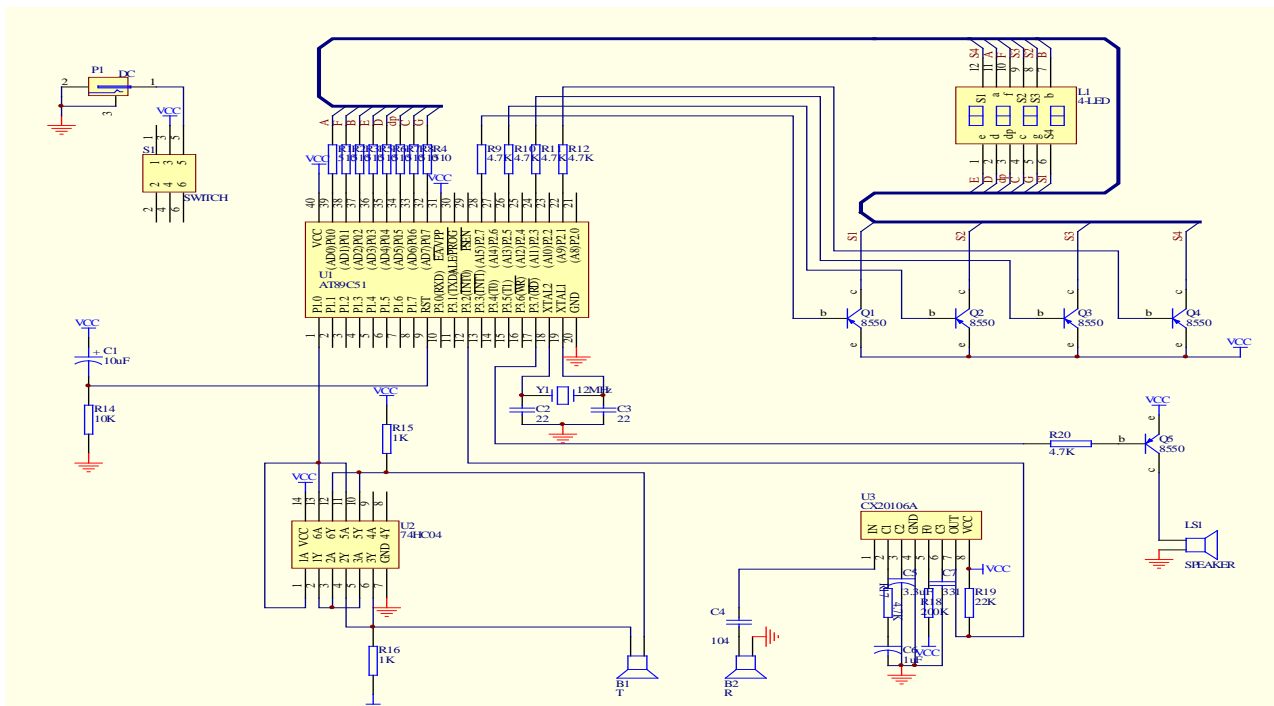
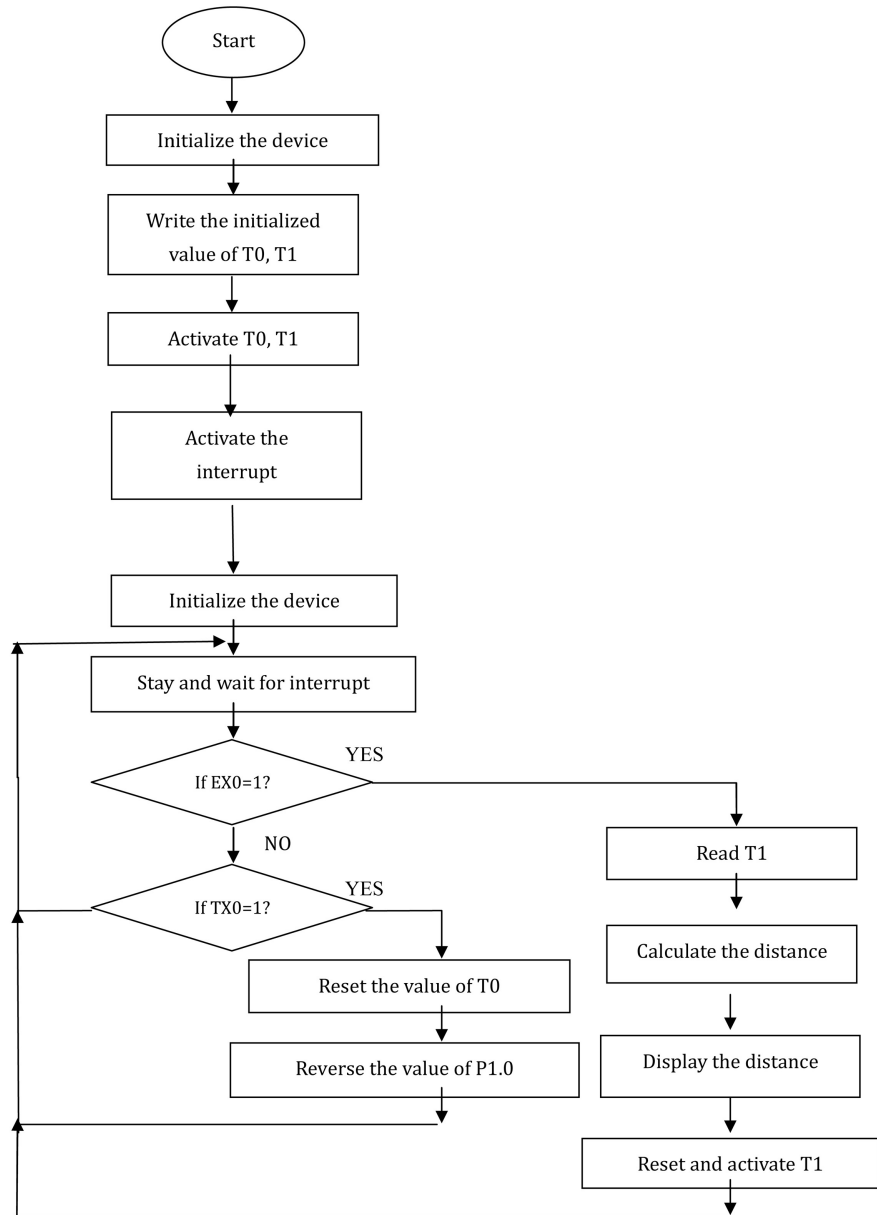


Figure 12. Schematic circuit and the PCB.

non-electrical layers, setting up new design rules, positioning the components on the PCB, automatically routing the board, verifying your board design, printing to a Windows printing device.

4.2. The Programming for AT89C51

The AT89C51 is programmed by C language in the surrounding of Keil version 4, and then the program is downloaded to the device. In this chapter, the flow chart of program will be listed.



5. Error Analysis and Testing Result

In the previous chapters, the simulating results of each parts of circuit have been indicate, which is based on a software test. In this part, the practical circuit

making process and testing result will be shown in the form of photograph, which is completed in the lab.

1) The handmade circuit board (testing) making process (Figure 13 and Figure 14)

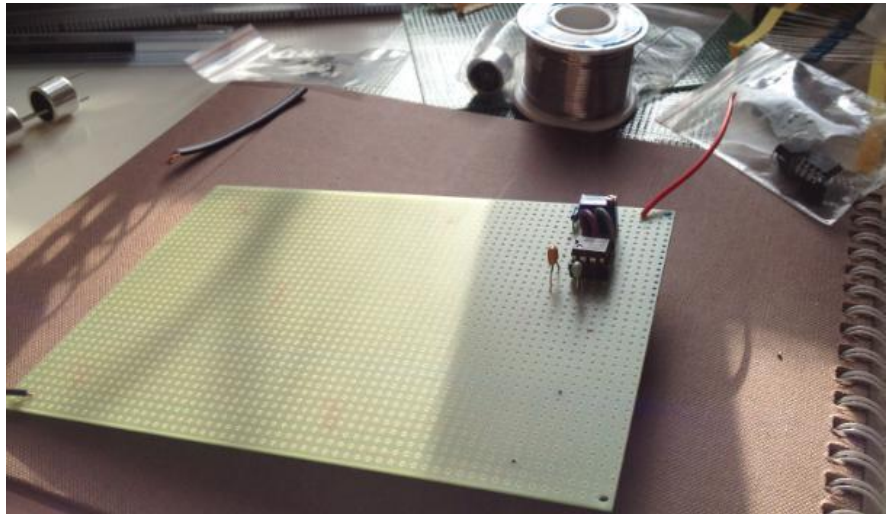


Figure 13. The beginning of board.

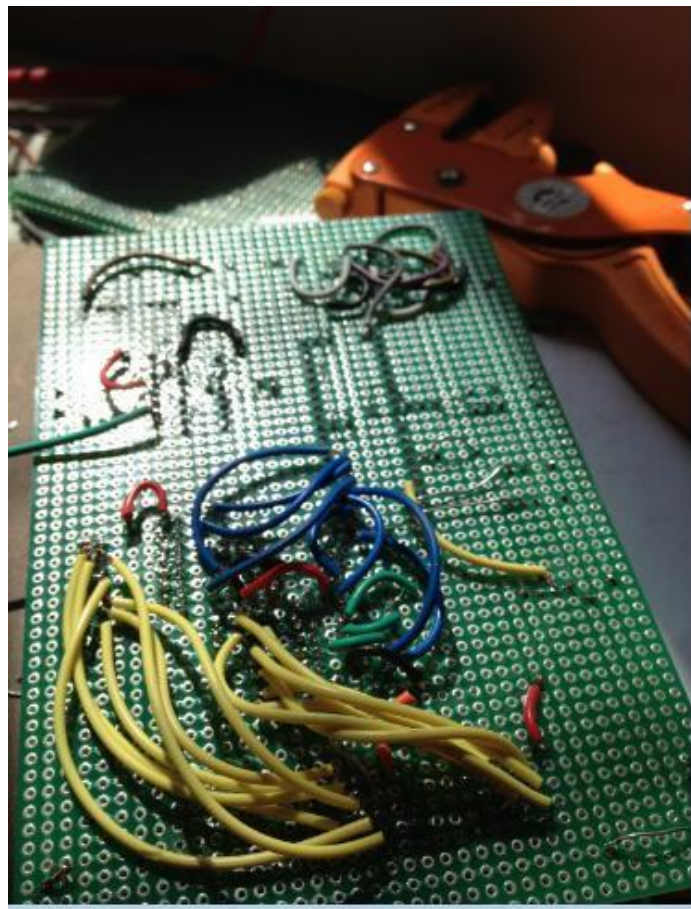


Figure 14. The solder side of the board.

2) Testing and debugging the hardware and software

In this system, the design of hardware and software are modularized, which makes the system debugging extremely convenient. Every module should be detected first, then put them together. This can improve the efficiency of debugging to a large extent and accelerate the schedule of the project.

Hardware debugging can be started from every functional modules of the program, until all of the program work correctly. There are 3 main announcements should be paid more attention.

- When installing the 2-ultrasonic transducers, they must be parallel and separated by 3 cm.
- For achieving the suitable sensitivity of receiver and anti-jamming capability, the value of capacitor C4 could be adjusted appropriately.
- When soldering the component, the temperature should not be too higher. Otherwise, it is easily to damage the device more or less and make the total system paralyzed. In addition, this error cannot be found easily.

When the hardware circuit was manufactured and detected completely, the program will be edited and tested in the software. After that, the program should download to the MCU and test run it. According to the specific result, correcting the program repeatedly until the result is identical with presumptive purpose. In a word, both software and hardware debugging cannot be separated, because some researches confirm that the error of hardware was found and corrected in general.

3) Error analysis

Figure 15 [10] shows the change in ultrasound sensor readings due to angular deviation of the object surface but in the same distances. Noise in the measurement is another source of error. If an object has a small surface, there will be an angle between two ultrasound headers and the object. In this saturation, the measured distance contains two parts which is a sum of the distance from the transmit header and receive header to the obstacle rather than the real distance. So, the measured distance is a little difference to the real distance. According to test, the smaller measured distance is, the more accuracy it is.

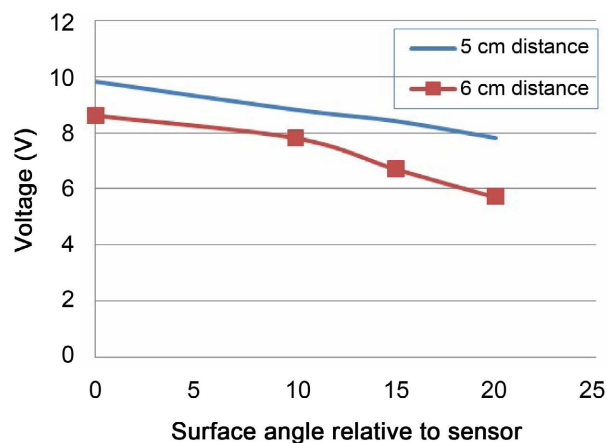


Figure 15. Change in sensor readings due to change in angle.

4) Testing result

The transmitter was enabled and tested on the oscilloscope shown in **Figure 16** to get the desired frequency.

The **Figure 17** shows the receiver oscilloscope graph. The voltage/division used on this oscilloscope is 2 V/div and the time/division used is 5 ms/div.

When the measured distance is the range of 0 - 1.0 meter, the system will generator a alarm signal. As the reduction of the distance, the frequency of alarm signal will be more and more quick. The LED should display the real distance all the time. In addition, when the measured distance is far away from 1.0 meter, that will be a safety region and the speaker will not generator the alarm signal. But the real distance is displayed all the time.

Although some datum is a little different to the actual distance, it is the reasonable error due to the temperature and the angle between the ultrasound header and the object. **Figure 18** shows the testing distance.

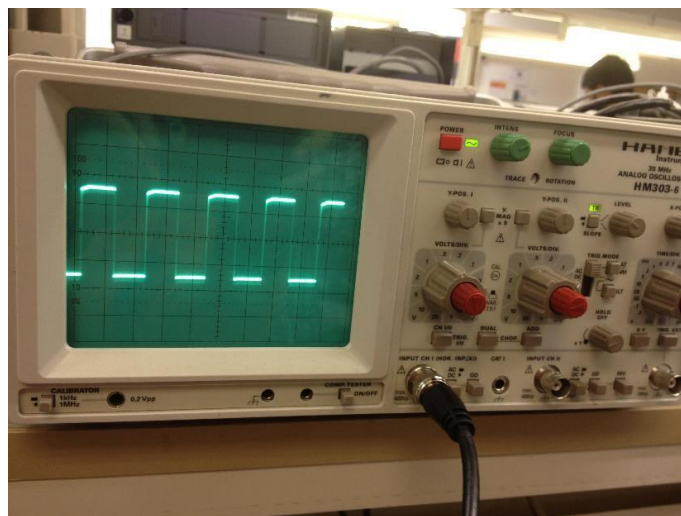


Figure 16. Transmitter signal.

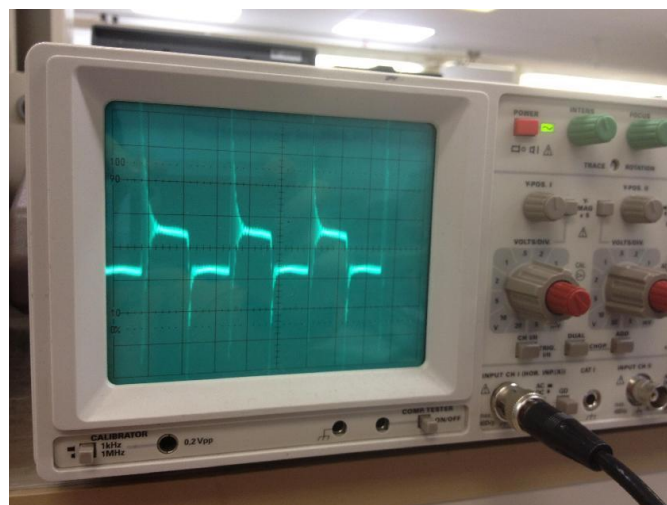


Figure 17. Receiver oscilloscope graph.

Actual distance	Measured distance	Alarm signal?
0.20 m	0.20 m	Yes
0.60 m	0.60 m	Yes
1.00 m	1.00 m	Yes
1.24 m	1.24 m	No
1.84 m	1.85 m	No
2.00 m	2.01 m	No
2.50 m	2.50 m	No

Figure 18. Testing results table.

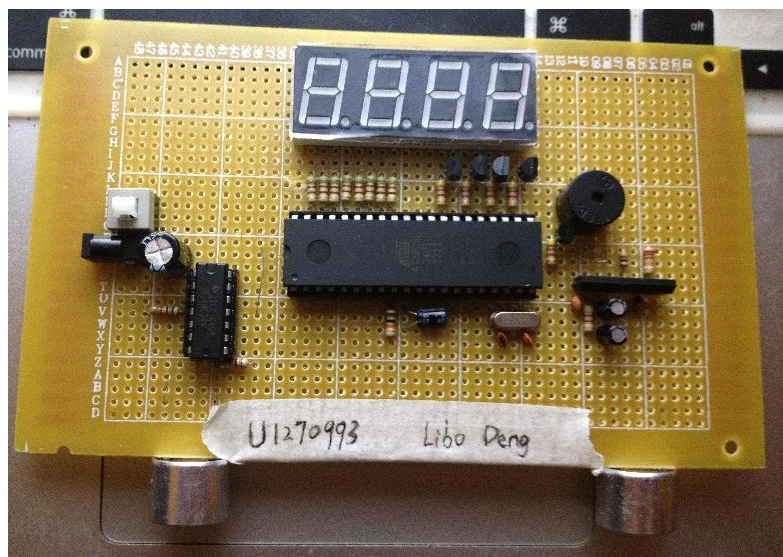


Figure 19. Final board and Printed Circuit Board (PCB).

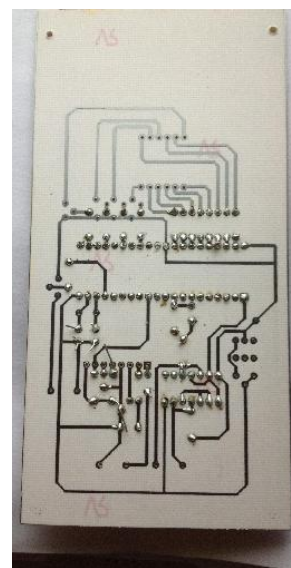


Figure 20. The solder side.

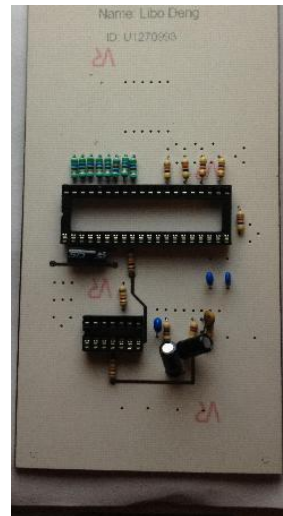


Figure 21. The topside.

6. The Final Board and Printed Circuit Board (PCB) (Figure 19)

This PCB is manufactured by the staff in the lab, by using the design that the author provided. Unfortunately, some components were not soldered in time. Without doubt, it should work directly if all the components were soldered correctly (as shown in **Figure 20** and **Figure 21**).

7. Conclusion and Further Work

In this report, an ultrasound distance measurement system is described, which is useful for reversing a vehicle. This system contains hardware and software to support it. Two ultrasound headers are used to transmit and receive the signal and a speaker is for generating an audio signal if the measured distance falls in the safety zone. An LED displays the real distance and all the process is controlled by a microcontroller. Although this project is simple and low-cost, the sensor allows resolutions of better than 1 mm to be obtained in quiet conditions. The obtained measurements are accurate enough for giving important information to the driver.

Some details of this system should be improved in further work. For example, with the change of temperature, the speed of ultrasound is changed to a small extent. If the surface of object is not smooth, this affects the accuracy of this system and leads to a little error on the measurement. By the way, further work could be done on the fusion of the information supplied by the ultrasound that can be utilized for mobile robot navigation in unknown environment rather than reversing a car.

Conflicts of Interest

The author declares no conflicts of interest regarding the publication of this paper.

References

- [1] Swickard, J.E., *et al.* (2004) 2003 Accident Review. Business & Commercial Aviation.
- [2] Parvis, A. (2002) An Ultrasonic Sensor for Distance Measurement in Automotive Applications. *IEEE Sensors Journal*, **1**, 200.
<https://doi.org/10.1109/JSEN.2001.936931>
- [3] Parrilla, M., Anaya, J.J. and Fritsch, C. (1991) Digital Signal Processing Techniques for High Accuracy Ultrasonic Range Measurements. *IEEE Transactions on Instrumentation and Measurement*, **40**, 759-763. <https://doi.org/10.1109/19.85348>
- [4] Eik-Nes, S.H. (1995) A Review of Epidemiological Studies of Human Exposure to Ultrasound. *Ultrasound in Obstetrics & Gynecology*, **6**, 293-298.
<https://doi.org/10.1046/j.1469-0705.1995.06040293.x>
- [5] Zhang, S.Y. (2000) Research of Ultrasonic Distance Measurement Device. Master's Thesis, Hohai University, Nanjing.
- [6] The Datasheet of CX20106A.
<https://download.csdn.net/download/xhr131452007/30844728>
- [7] Elmer, H., Schweinzer, H. and Magerl, G. (2009) High Resolution Ultrasound Distance Measurement for Long Distances. *Annals of Surgery*, **250**, 943-949.
- [8] Li, X.M. and Qi, R.H. (2003) The Design of Vehicle Navigation Alarm System Based on SB5227 Microprocessor, *Technisches Messen*, **70**, 173-179.
- [9] Datasheet of DS18B20. <https://www.renrendoc.com/paper/87231442.html>
- [10] Weir, R.F. and Childress, D.S. (1997) A New Method of Characterising Gait Using a Portable, Real-Time, and Ultrasound Ranging Device. *Proceedings of the 19th Annual International Conference of the IEEE Engineering in Medicine and Biology Society. 'Magnificent Milestones and Emerging Opportunities in Medical Engineering' (Cat. No.97CH36136)*, Chicago, 30 October 1997 - 2 November 1997, 1810-1812.
<https://doi.org/10.1109/IEMBS.1997.757080>

Appendix I: The Program

```

#include<reg51.h>
#include<math.h>
#define uint unsigned int
#define uchar unsigned char
/*define the function*/
long int time;      /*define "time"*/
bit CLflag;        /*define "measuring"*/
char cshu;
#define T12us (256-12)
sbit VOLCK=P1^0;

sbit S1=P2^1;
sbit S2=P2^3;
sbit S3=P2^5;
sbit S4=P2^7;

sbit alam=P3^7;

unsigned char number[5]; /*display data*/
unsigned char temp_number[5];

unsigned char fr_alam;
unsigned char icon; //count the distance

char code table[] = {0x5F,0x44,0x9D,0xD5,0xC6,0xD3,0xDB,0x45,0xDF,0xD7};
/* 0,1,2,3,4,5,6,7,8,9 */

/* delay */
void delay( int j)
{
    int i;
    for(i=0;i<j;i++);
}

void display(void) /* display the data */
{
    P0=~table[number[4]];
    S1=0;
    delay(200);
    S1=1;
    P0=~(table[number[3]]|0x20);
    S2=0;
    delay(200);
    S2=1;
    P0=~table[number[2]];
}

```

```
S3=0;
delay(200);
S3=1;
P0=~table[number[1]];
S4=0;
delay(200);
S4=1;
}

/* initialize the timer T0、 T1 */
void init_CTC(void )
{
    TMOD = 0x21;
    ET0 = 0;
}

void init_INT( )
{
    IP=0x01;
    TCON=0;
}

//=====The convert of distance and frequency=====
void Distance_Frequency(void)
{
    if(number[3]>0) // the distance is more than 1meter, no alarm signal
        icont=0;
    else
    {
        if(number[2]>0)
        {
            switch(number[2])
            {
                case 9: icont=9; break;
                case 8: icont=8; break;
                case 7: icont=7; break;
                case 6: icont=6; break;
                case 5: icont=5; break;
                case 4: icont=4; break;
                case 3: icont=3; break;
                case 2: icont=2; break;
                case 1: icont=1; break;
                default: break;
            }
        }
    }
}
```

```

        else
        {
            if(number[1]>0)
                icont=1;
            else
                icont=0;
        }
    }
}

void Timetojuli(void) /* calculate the distance */
{
    long i;
    i=(long)time*170;
    temp_number[4]=i/1000000; /* 10meters */
    i=i-temp_number[4]*1000000;
    temp_number[3]=i/100000; /* meters */
    i=i-temp_number[3]*100000;
    temp_number[2]=i/10000; /* decimeter */
    i=i-temp_number[2]*10000;
    temp_number[1]=i/1000; /* cm */
    i=i-(long)temp_number[1]*1000;
    temp_number[0]=i/100; /* mm */

    if((temp_number[4]!=0)||((temp_number[3]!=0)||((temp_number[2]!=0)||((temp_number[1]!=0)))
    {
        number[4]=temp_number[4];
        number[3]=temp_number[3];
        number[2]=temp_number[2];
        number[1]=temp_number[1];
        number[0]=temp_number[0];

        Distance_Frequency();
    }
}

void serve_INT0( ) interrupt 0
{

    TR0=0; /* switch off the timer 0 */
    EX0=0; /* switch off the interrupt 0 */
    time=(long)TL0;
    time+=(long)TH0*256;
    Timetojuli();
    CLflag=1;
    TH0=0;
}

```

```
    TL0=0;
}

void CTC0_INT ( )  interrupt  1
{
    TR0=0;
    ET0=0;
}

void CTC1_INT ( )  interrupt  3
{
    VOLCK=~VOLCK;
    cshu++;
}

/*main*/
void main()
{
    init_CTC( );           /* initialize the timer */
    init_INT( );          /* initialize the external interrupt*/
    CLflag=1;             /* begin to measure   */
    cshu=0;
    IE=0x80;
    ET1=1;      ET0=1;
    TR1=0;      TR0=0;
    TL1=T12us;  TH1=T12us;
    while(1)
    {
        CLflag=0;
        cshu=0;
        EX0=0;
        TH1=T12us;
        TL1=T12us;
        TL0 = 0;      TH0 = 0;
        VOLCK=0;
        TR1=1;
        while(cshu<20) ;
        TR1=0;
        TR0=1;
        EX0=1;
        display();
        Distance_Frequency();
        fr_alam++;
    }
}
```

