Motorola Flash Innovation 2003 Contest
Project Number: **F2017**

PROXIMITY SENSOR SYSTEM FOR THE VISUALLY HANDICAPPED USING THE MOTOROLA 68HC908QT

Abstract: A proximity sensor system using ultrasonic technology with the Motorola 68HC908QT as the core processor was developed. The objective of this proximity sensor system is to help the visually handicapped to detect obstacles around them. It is small in size; 9V battery-operated and can detect obstacles more than 2 metres away from the sensor system.

For the purpose of testing the prototype, the Motorola M68DEMOQTY demonstration board was used. Figure 1 shows the block diagram of the implementation.

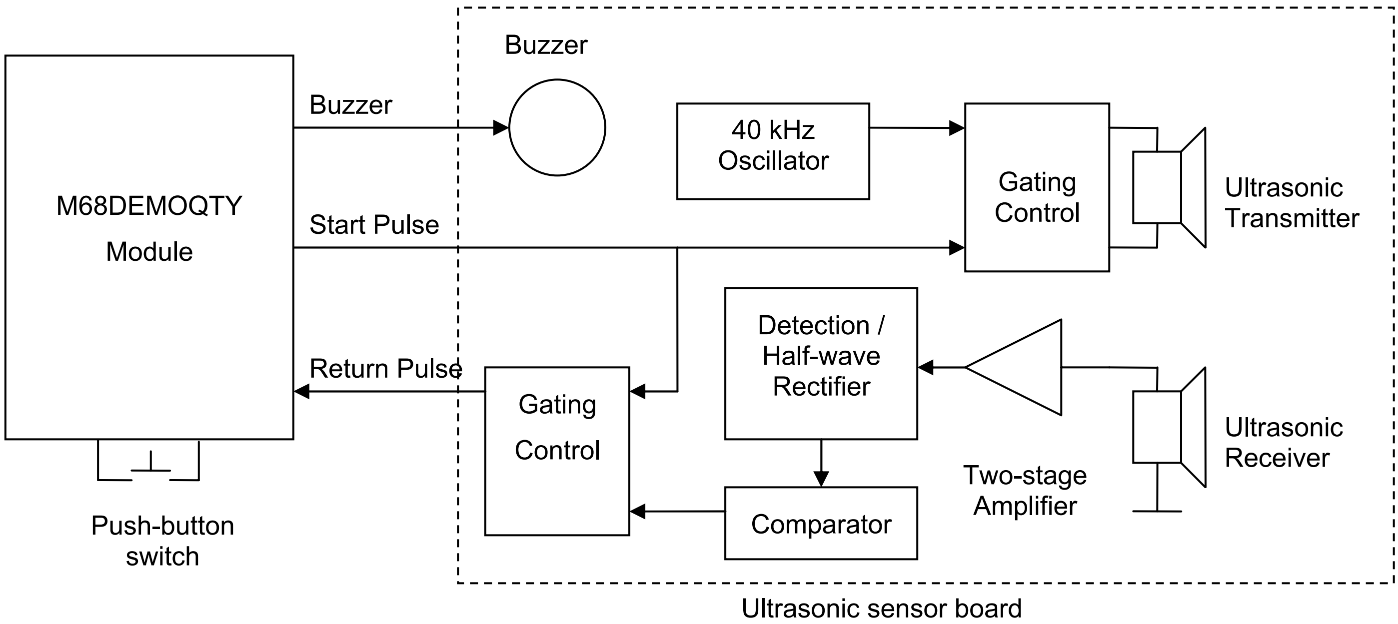

Figure 1: Ultrasonic Proximity Sensor Project using 68HC908QT

Figure 2 shows the implementation of the prototype using a breadboard.

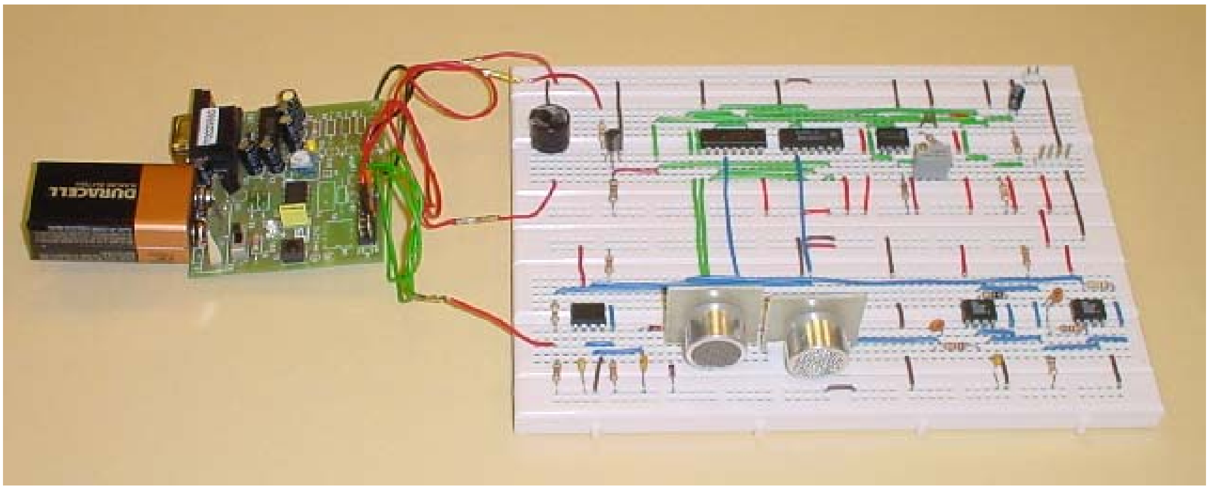

Figure 2: Breadboard Prototype

To detect obstacles, the user presses the push-button switch on the M68DEMOQTY module[1]. This causes the 68HC908QT to generate a short positive-going pulse ("Start Pulse") with a pulse width of approximately 1.2 milliseconds. The pulse acts as a gate for the output of the 40 kHz oscillator to drive the ultrasonic transmitter.

The reflection of ultrasonic waves from an obstacle is picked up by the ultrasonic receiver and converted into a electrical (reflected) signals. After going through the two-stage amplifier, rectifier and comparator, the reflected signal is detected as a positive-going pulse ("Return Pulse") by the 68HC908QT. The 68HC908QT then estimates the time difference between the "Start Pulse" and "Return Pulse".

To aid the user (who is visually handicapped) to estimate the distance between him and the obstacle, the 68HC908QT generates a series of five beeps using the buzzer. The closer is the distance between the user and obstacle, the shorter is the interval between the beeps.

While the project was originally meant to be a proximity sensor, giving rough estimates of the distance between the user and an obstacle, it can be easily adapted as an ultrasonic range meter, in which case, some form of calibration may have to be done for giving accurate measurements, and also some means to display the measured distance e.g. LEDs or LCD.

[1] In final version of the implementation, a larger switch that is more conveniently placed will be used.

1. INTRODUCTION

Currently, a visually handicapped person relies on the use of the cane to help him sense obstacles in his surroundings. The maximum distance of the obstacles that can be sensed is limited by the length of the cane. The visually handicapped has to move sufficiently close to the obstacles to realize their presence.

The ultrasonic proximity sensor system is specifically designed to help the visually handicapped person to increase the range at which potential obstacles can be detected. The system developed here is a very low-cost and easy-to-use system. It makes use of off-the-shelf circuit components and the versatile 68HC908QT micro-controller. For the prototype developed, the Motorola M68DEMOQTY demonstration board was used.

2. USE OF THE 68HC908QT

To implement the project, only four pins of the 68HC908QT are required. Table 1 shows the use of these pins.

Pin Number	Name	Use	Remark
3	PA4	Return Pulse	Input. Designated as bean object "Bit2" in program
4	PA3	Buzzer Control	Output to drive the buzzer. Designated as bean object "Bit3" in program
5	PA2	Push-button switch	Input with internal pull-up resistor active. The M68DEMOQTY module already has the push-button switch installed. Designated as bean object "Bit4" in program
6	PA1	Start Pulse	Output. Designated as bean object "Bit1" in program

Table 1: Usage of port pins of 68HC908QT

3. SENSOR CIRCUIT DESCRIPTION

Please refer to Figure 3.

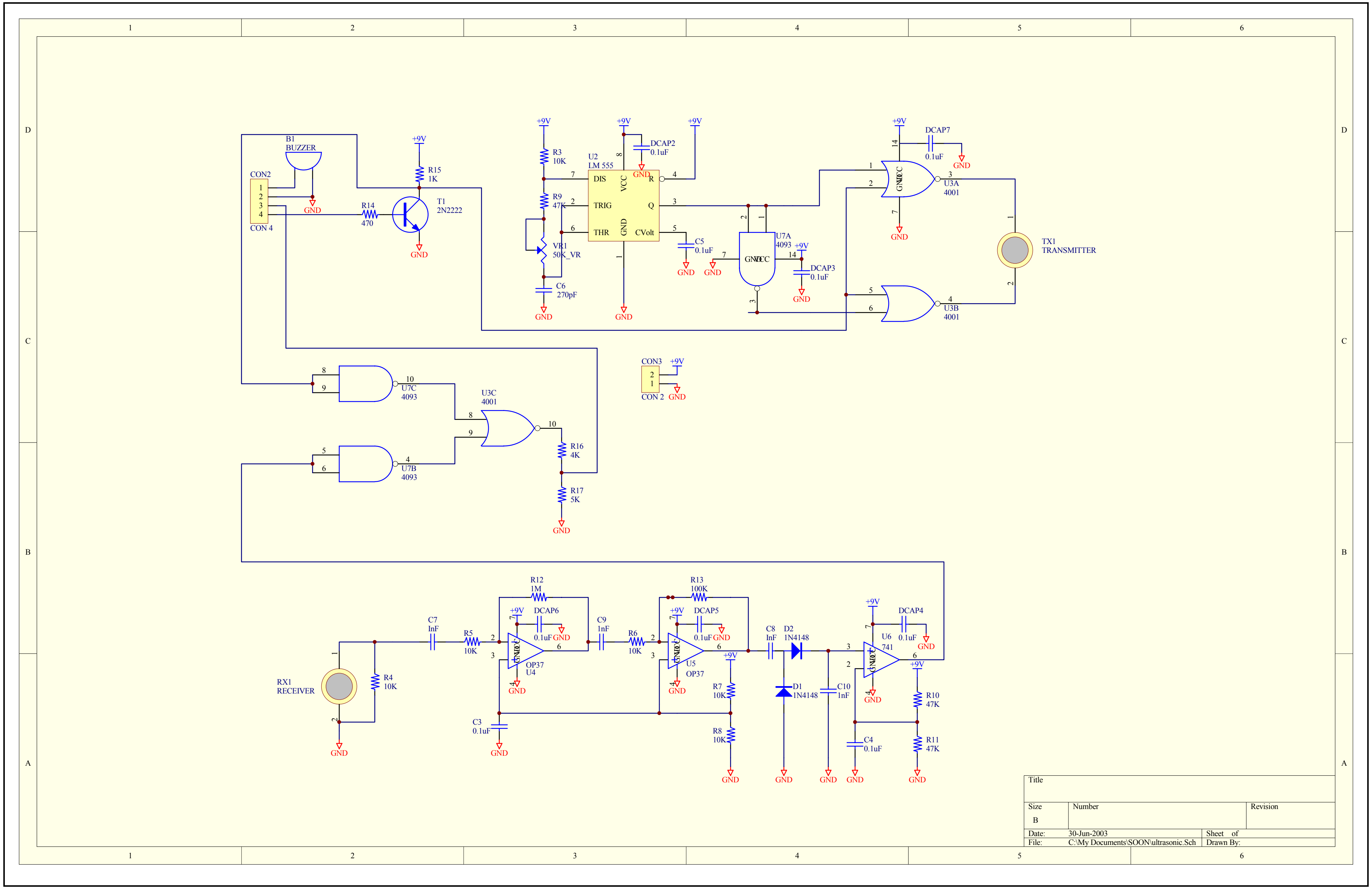

Figure 3: Schematic diagram of sensor module

(a) Oscillator and Transmitter

The oscillator is implemented using a standard 555 Timer (U2). Variable resistor VR1 is used to tune the oscillation frequency to 40 kHz. This is the frequency where maximum response from the ultrasonic transducers is achieved. Mathematically the formula to determine the frequency, f_{osc}, is given by

$$f_{\mathrm{osc}} = \frac{1}{0.693 C_6 \left(R_3 + 2R_9 + 2VR_1 \right)}$$

(b) Start Pulse and Gating control

The "Start Pulse" is generated when the push-button switch on the M68DEMOQTY demonstration board is momentarily pressed. The "Start Pulse" is a positive-going pulse with a pulse width about 1.2 milliseconds and it determines when the 40 kHz signal from the oscillator drives the ultrasonic transmitter. The duration of the "Start Pulse" is software-controlled by the 68HC908QT.

The "Start Pulse", when activated, turns on transistor T1 (2N2222) and drives it to saturation. This causes the collector voltage to drop to approximately 0 V. In fact, the signal observed at the collector is the inverted version of the start pulse.

U7A acts as an inverter to invert the signal generated by U2. During the active period of the start pulse, the output of the U3A and U3B are inverted with respect to one another. This exerts a push-pull effect on the ultrasonic transmitter.

Figure 4 shows the signals at the collector of T1 (top trace) and at the output of U3A (bottom trace) respectively.

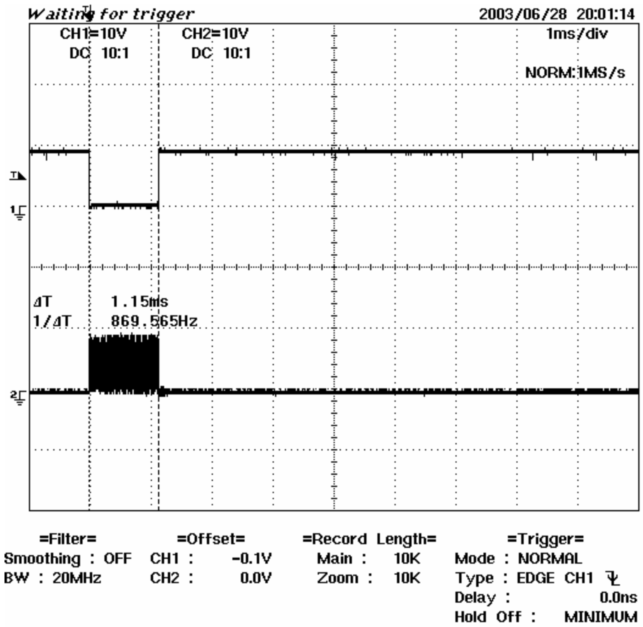

Figure 4

(c) Return Signal Processing

When the ultrasonic waves hit an obstacle, they are reflected and picked up by the ultrasonic receiver, which then converts the waves to an electrical signal ("reflected signal").

(i) Amplifier

The reflected signal is fed through a 60-dB two-stage amplifier system comprising U4 and U5. U4 and U5 are low-noise operational amplifiers. As both op-amps are operated in single-supply mode, R7 and R8 forms a voltage divider to provide a d.c. shift of $0.5V_{CC}$ where $V_{CC} = 9V$.

Figure 5 (lower trace) shows the reflected signal (with respect to the inverted start pulse) after passing through the amplifier stage.

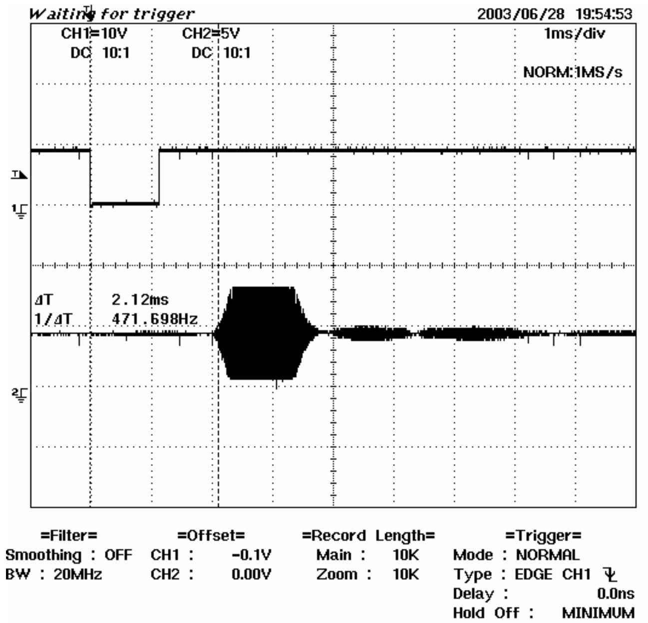

Figure 5

(ii) Rectifier/Detector

The circuit comprising D1, D2, C8 and C10 acts as a half-wave rectifier circuit-cum-envelope detector of the output of U5.

(iii) Comparator and Gating Control

Op-amp U6 acts as a comparator that compares the envelope of the detected signal with a reference voltage of $0.5V_{CC}$, such that when the envelope voltage exceeds $0.5V_{CC}$, the output of the comparator is at the positive saturation voltage of U6.

The output of U6 is gated with collector signal of T1, so as to eliminate the effect of spurious signals picked up during the time when the transmitter is on.

4. Software description

The software implementation was done using the special edition of the Code Warrior for HC08, making full use of the **beans** feature. Listing 1 shows the main program code to implement the project. Listing for the other codes automatically generated by the Code Warrior IDE (e.g. Events.c etc) have been omitted.

```
#include "Cpu.h"
#include "Events.h"
#include "AD1.h"
#include "Bit1.h"
#include "Bit2.h"
#include "Bit3.h"
#include "Bit4.h"
/*Include shared modules, which are used for whole project*/
#include "PE_Types.h"
#include "PE_Error.h"
#include "PE_Const.h"
#include "IO_Map.h"

/* static byte myValues[1]; */ /* Number of channels */

void main(void)
{

	int i;
	int j;
	int k;
	bool s;

	for(;;) {
		/* Check if switch is pressed */
		s = Bit4_GetVal();
		if (s!=0) { /* switch is not pressed, do nothing */
		}
		else {
			Bit1_PutVal(1);     /* start pulse high */
			for (i=0; i<100; i++); /* wait for 1.2 ms */
			Bit1_PutVal(0);     /* start pulse low */
			i = 0;
			for(;;) {
				/* Check the return pulse */
				s = Bit2_GetVal();
				if (s != 0) {
					break; /* return pulse detected */
				}
				else {
					i = i+1;
					if (i > 2000) {
						break;  /* do not wait forever */
					}
				}
			}
			/* the value of i indicates the distance */
			i = i/50;
			if (i<1) {
				i = 1;
			}
			/* The interval between beeps is proportional to i */

			for (j=0; j<5; j++) {
				/* Turn on the buzzer */
				Bit3_PutVal(1);
				for (k=0; k<5000; k++);
				/* Turn off the buzzer */
				Bit3_PutVal(0);
				for (k=0; k<1000*i; k++);
			}
		}
	}
}
```

Listing 1: Program Code

In the program, the 68HC908QT operates in an infinite loop. In the loop it does the following:

(a) Check if the push-button switch is pressed i.e. `Bit4_GetVal()` returns a zero.

(b) If the push-button switched is pressed, generate the "Start Pulse". This is done by the statements:

```
Bit1_PutVal(1);
for (i=0; i<100; i++);
Bit1_PutVal(0);
```

(c) Once the start pulse has been removed (in which case the ultrasonic sensor has stopped transmitting, the program times the detection of the return pulse by reading the status of bean Bit2 using `Bit2_GetVal()`. To avoid a situation where no return pulse is detected (due to the absence of nearby obstacles) and the ending up in an infinite loop, a cap of 2000 is placed on the distance counter variable `i`.

(d) The five beeps are generated by turning on (`Bit3_PutVal(1)`) and off (`Bit3_PutVal(0)`) the buzzer. The on-time is fixed, whereas the off-time is made proportional to the value of the distance counter. Hence the farther the distance of the obstacle, the longer is the interval between beeps.

5. Conclusion

The system prototype was implemented and tested. It was found that it was able to detect obstacles at least two meters away. Despite its simplicity, this project and the process involved are useful because of the following reasons:

(a) it serves its purpose as a proximity sensor system.

(b) it integrates the knowledge of analogue circuit implementation and troubleshooting, prototyping, programming and micro-controller application.

(c) it makes use of the Code Warrior as a powerful development tool to facilitate the human interface with the micro-controller.