Department of Electrical and Computer Engineering

Microprocessors 2DP4 Project

Instructor: Dr. Doyle

As a future member of the engineering profession, the student is responsible for performing the required work in an honest manner, without plagiarism and cheating. Submitting this work with my name and student number is a statement and understanding that this work is my own and adheres to the Academic Integrity Policy of McMaster University and the Code of Conduct of the Professional Engineers of Ontario.

Prepared By:
Hanxi Huang
1303296
April 4th,2015

Table of Content

1. Introduction and Background

In this project, our main target is to collecting analog data by using micro-controller. The project are based on ADC conversion and serial communication. ADC is used in embedded system to solve the transformation from electrical signals to digital data. Serial communication is the process to sending data sequentially over a channel or PC.

First I built the circuit, which follows the Fig4 in project manual, and connect the input and ground of analog. I used RealTerm and Excel to determine whether the circuit correct. After that, Matlab output the data collected from analog to plot the frequency.

The ADC is popular in music reproduction technology. People prefer produce much music on computers using analog recording and therefore need ADC converters to create the pulse-code modulation data streams that go onto compact discs and digital music files.[1]

2. Design Methodology

(a)Quantify Signal Properties

Amplitude of a periodic variable is a measure of its change over a single period. The definition is the magnitude of the difference between the variable ‘s extreme values. The range of amplitude is peak to peak, which means from high to trough. Frequency is the number of occurrence of a repeating event per unit time. In the complex circuit, impedance is the measure of the opposition that a circuit presents to a current when a voltage is applied. It apply a limitation function to the circuit.

DC source is 5V which is connect red line+ in my circuit. When I justify the analog, I set amplitude is 2.5 Peak to peak,the sampling frequency is 300Hz based on LSD of my student number. In my experiment, the impedance of microphone vary with my input frequency.

(b) Transducer

I used Realterm to determine whether my circuit was right. If the circuit was right, I supposed to get the same frequency as input frequency. In the fig.2.1, the sine wave was shown.

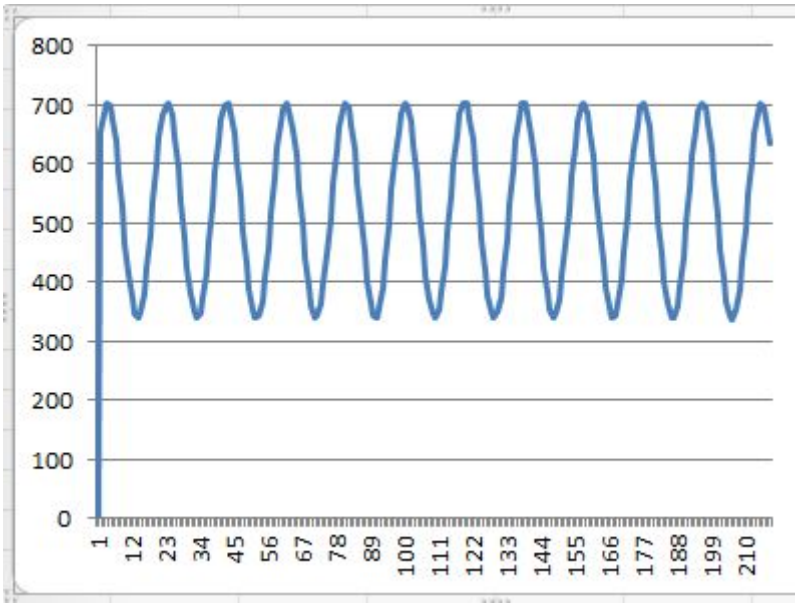

Fig.2.1 Sine Wave by Excel

Also, I took picture from the Oscilloscope when running Codewarrior, the figure was shown in fig.2.2. The fig.2.3 is my circuit board.

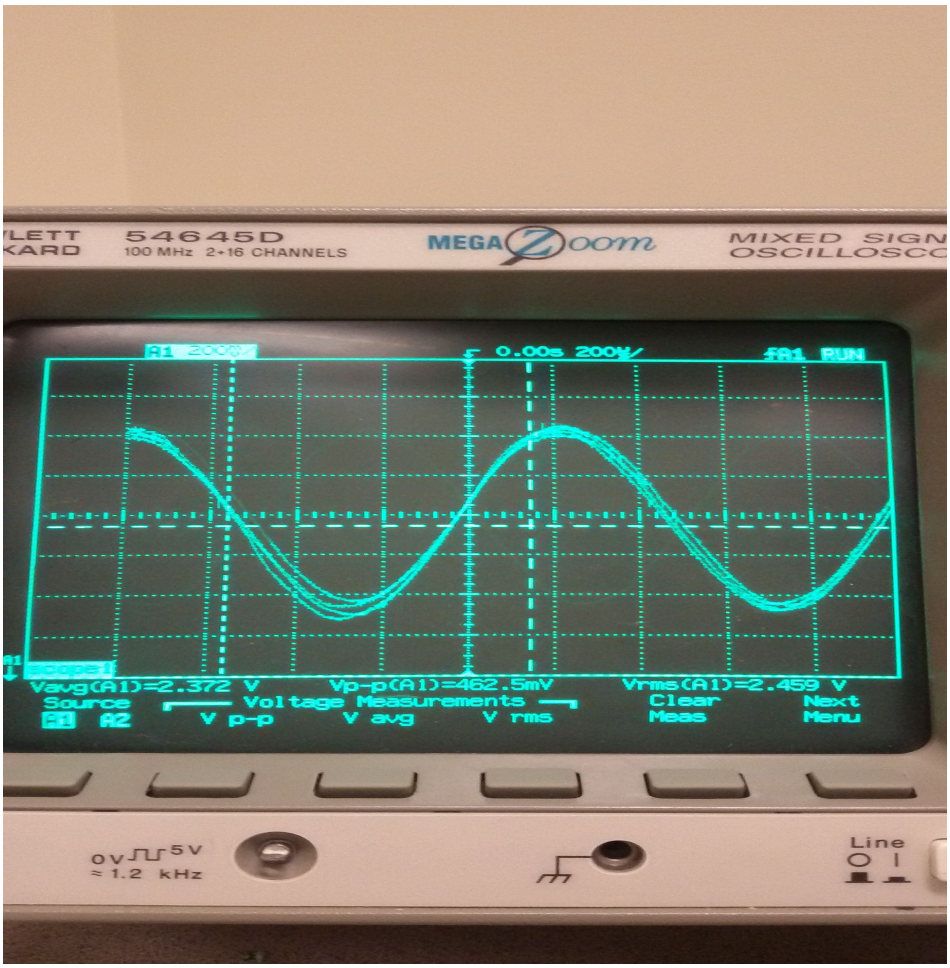

Fig.2.2. Sine Wave from the Oscilloscope

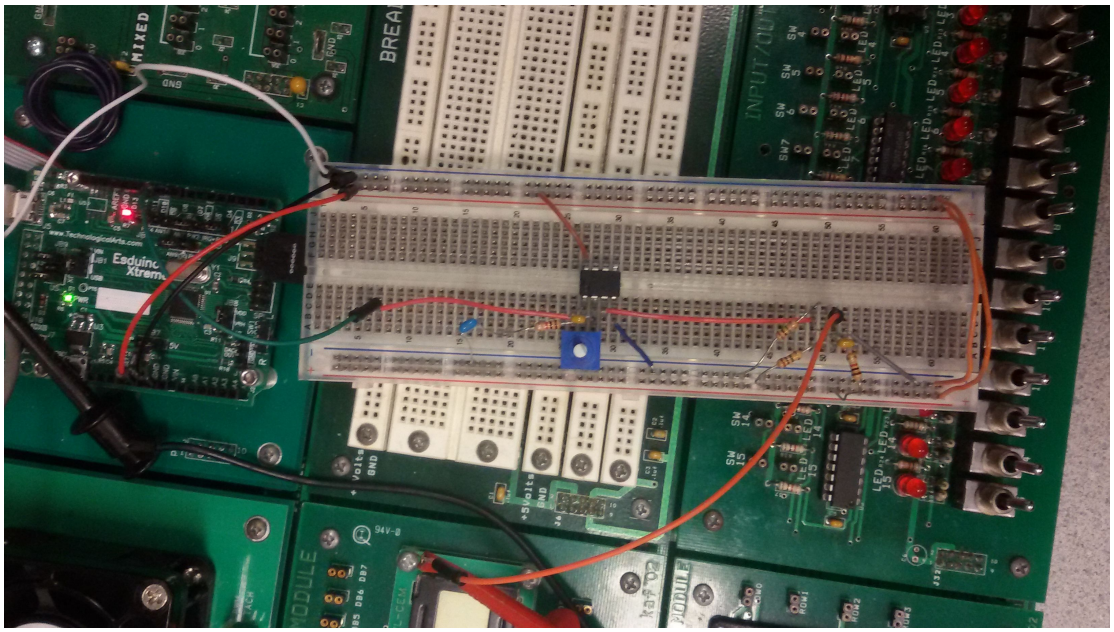

Fig.2.3. channel 9

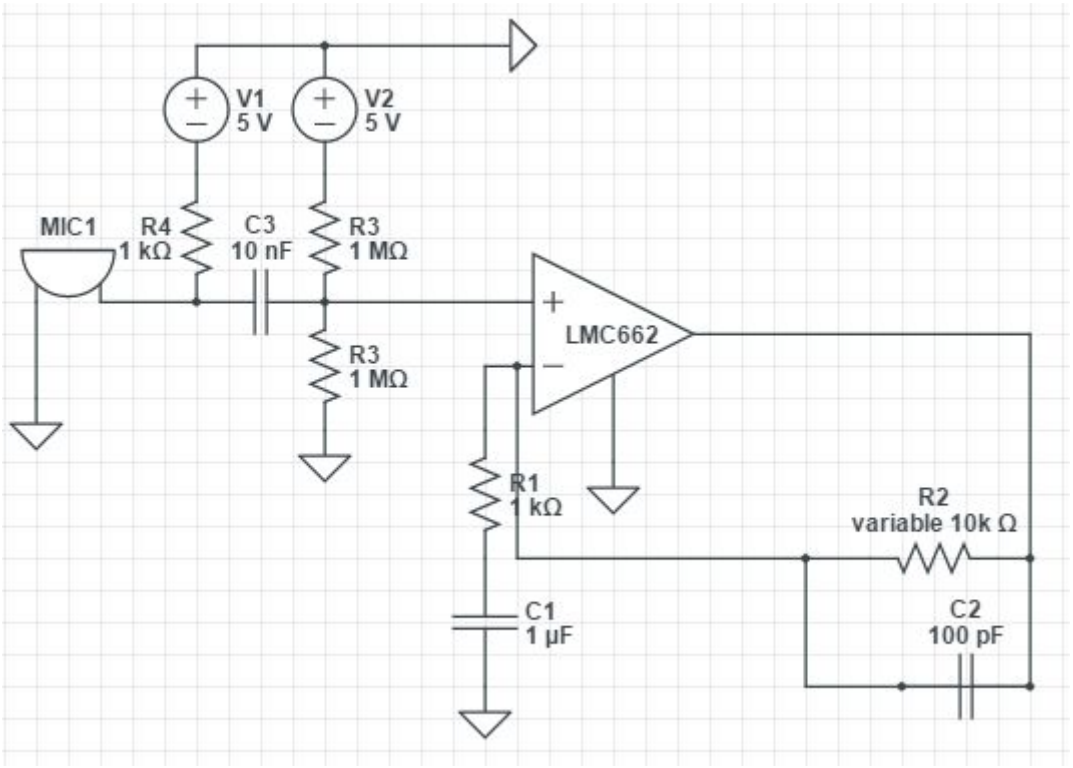

Fig.2.4 Circuit Schematic

Transducer is an electronic device that converts energy from one to another. In our case, the microphone converted voice to electrical signal.

(c) Precondition

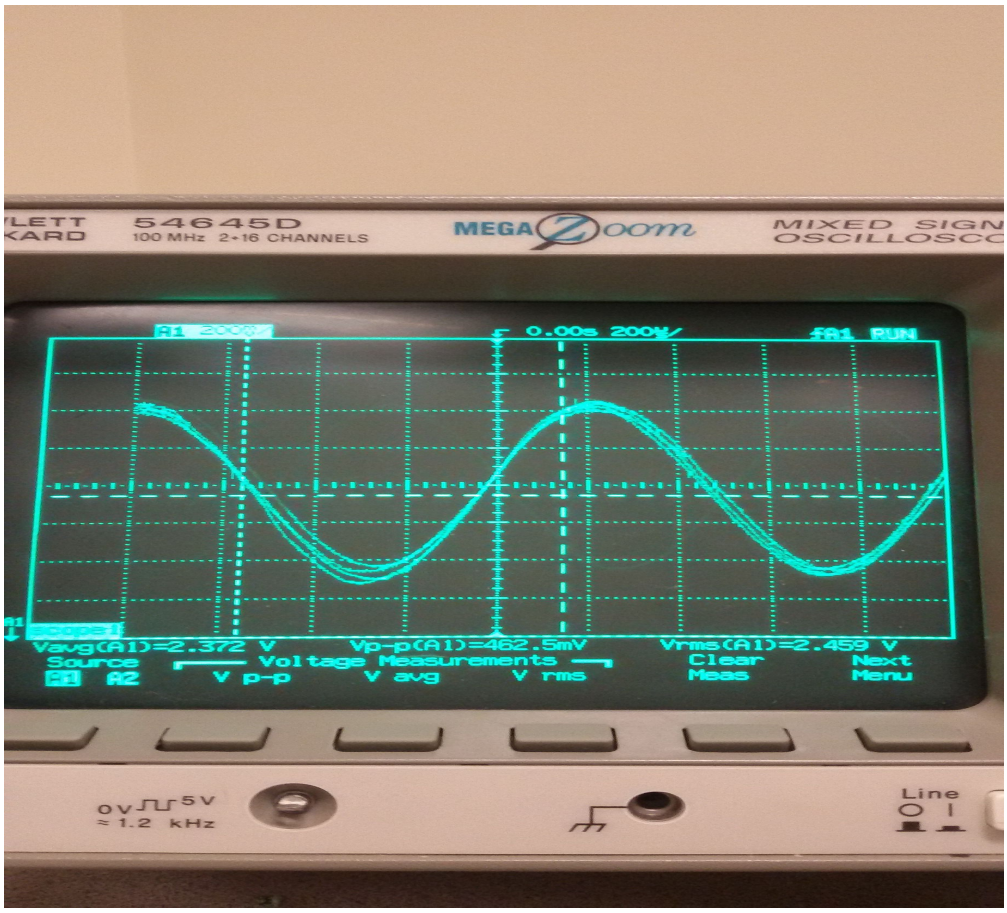

Fig.2.5.Sine Wave from the Oscilloscope

The fig.2.5 illustrate the sine wave, which is average voltage is around 2.5V(shown 2.372V in the graph), V peak to peak is 382.5 mV, Vrms is 2.459 V. When frequency input is 1KHz, the output result is around 1000Hz. When there is no frequency input, the output V peak to peak is 100mV.

(d) ADC **(student number is 1303296)**

In the experiment, the ADC resolution is 8-*bits*, which should get 256 Hz frequency. The ideal voltage range is 0V to 5V, however, I got around 4.5V at the experiment. The sampling frequency should be *300Hz*. My Bus speed is *4MHz* and in *channel 9*.

✓ **Register Configuration**

ATDCTL1

7	6	5	4	3	2	1	0
0	SRES1	SRES0	0	1	1	1	1

SRES 1	SRES0	A/D Resolution
0	0	8-bit Resolution
0	1	10-bit Resolution
1	0	12-bit Resolution
1	1	Reserved

In my case is 8 bits,so my binary number is 0000 1111 that is ATDCTL1=0x0F

ATDCTL3

7	6	5	4	3	2	1	0
1	0	0	0	1	0	0	0

The ATDCTL3 should be 10010000, which bit 7 is 1 because right justified. The bit 4 is 1 because the number of conversion per sequence is 1.

ATDCTL4
We don't need to change

ATDCTL5

7	6	5	4	3	2	1	0
0	0	1	0	1	0	0	1

Bit 5 is one because Continuous Conversion Sequence Mode. Channel 9 is 1001 from bit 3 to bit0.

Table 16-15. Analog Input Channel Select Coding

SC	CD	CC	CB	CA	Analog Input Channel
0	0	0	0	0	AN0
	0	0	0	1	AN1
	0	0	1	0	AN2
	0	0	1	1	AN3
	0	1	0	0	AN4
	0	1	0	1	AN5
	0	1	1	0	AN6
	0	1	1	1	AN7
	1	0	0	0	AN8
	1	0	0	1	AN9
	1	0	1	0	AN10
	1	0	1	1	AN11
	1	1	0	0	AN12
	1	1	0	1	AN13
	1	1	1	0	AN14
	1	1	1	1	AN15

Fig.2.6. Analog input channel coding

✓ Bus Speed Calculation (my bus speed is 4MHz)

✓ Baud Rate Calculation

I set post Div is 0,

```
void setClk(void){


CPMUCLKS = 0x80; //PLLSEL = 1
CPMUOSC = 0x00;  //OSCE = 0
CPMUSYNR = 0x03; //VCOFRQ = 0, SYNDIV = 3
//Now fREF = 1MHz and fVCO = 2*fREF*(3+1) = 8MHz
CPMUFLG = 0x08; //LOCK = 1
CPMUPOSTDIV = 0x00; //
//PLLCLK = fVCO/1 = 8MHz
//Thus the Bus Clock  = 8/2 = 4MHz
}
```

Pllclk/2=BusSpeed=4Mhz
Pllclk=fvco/(PosDIv+1)
fvco=2*fREF*(SYDIV+1)

Bus Speed	CPMUPOSTDIV
4	0x03
8	0x07
16	0x0F

✓ Baud Divisor:
4Mhz/(16*9600)=26

✓ ADC

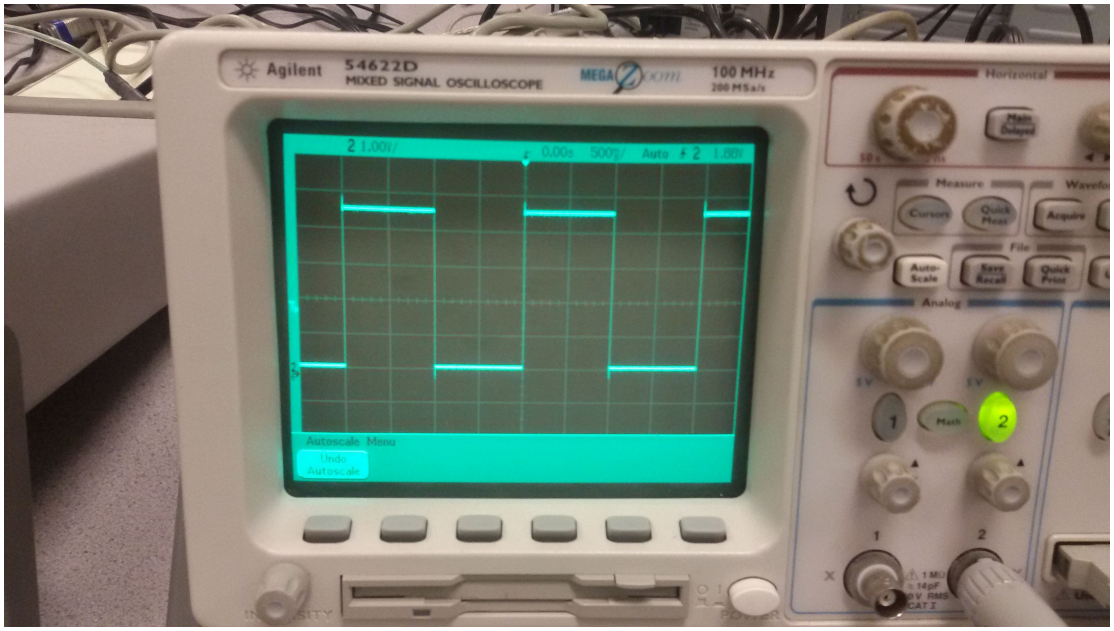

Fig.2.7

From the fig.2.7, the graph shown in oscilloscope. The graph could give us bus speed, delay calculation and sampling frequency. The graph should be in 6ms, which is two delay cycle, thus, a delay is around 3ms. The upper lines refer to LED on,and bottom lines refer to LED off. This is testing if the code in codewarrior right or not.

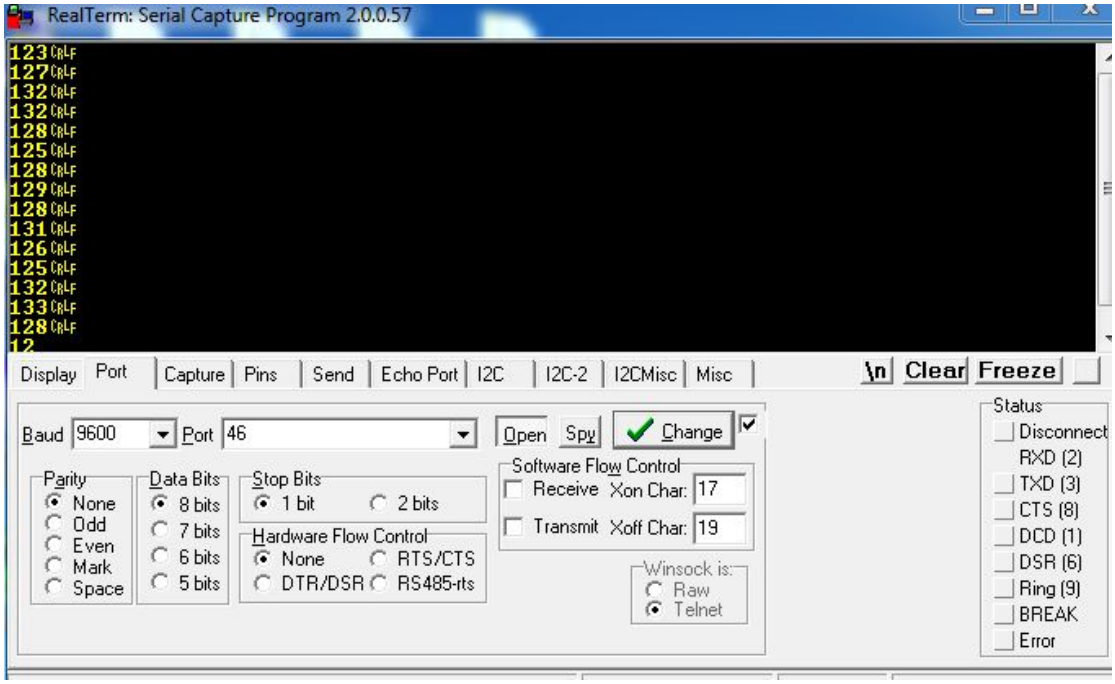

Fig.2.8

In fig.2.8, we can make sure that Vmax is 5V and Vmin is 0V. This picture determines when my average voltage is around 2.5V, the value may wave by 130mV.

(e) Data Processing (Algorithm Flowchart)

- ✓ Esduino

Hardware Introduction:

Freescale designed the 68HC12 as an upgrade to the 8-bit 68HC11 microcontroller. However,Motorola discovered that the performance of the 68HC12 microcontroller was not satisfactoryafter it was introduced to the market. The 68HC12 has the highest bus clock speed of 8 MHz. To be competitive, Freescale revised the design to achieve a bus clock rate of 25 MHz (a few microcontrollers can run at 33 MHz). The revised

68HC12 was referred to as the Star12 family. It was also named the HCS12 family. The HCS12 MCU has the same instruction set and addressing modes as does the 68HC12. However, many of the internal designs have been changed.

Course Application:
We used 6.25MHz in lab 2 due to reset, but actually the idea bus speed is 8MHz. The ESDX has two operating voltage, which are 3V and 5V. The memory resources are 240K Flash, 4K EEPROM and 11K RAM. We spent 90 dollars in ESDX. We used Assembling language, C to coding.

- ✓ CodeWarrior

Code Description:
In CodeWarrior for loop, we create a variable in the above code called val, and then we turn on the LED by using “exclusive”. The ATDDR0 means Wrap around channel select bits. These bits determine the channel for wrap around when doing mulN-channel conversions[2]. After that, we set input from Channel9 by using ATDDR0 and stored it into val. For the next line, I changed val from Binary to Decimal. Finally, using delay function msDelay to delay for 3 ms. It would go back to the top and continue the for loop again.

Process:
1.Setting you own student number specification. Change c code in codewarrior from ATCCTL0-5, and change setClk() function.
2.Start Button: connect to 5V in ESDX, the program will do serial communication.
3.Stop Button: In the high case, the serial communication will continuous; in low case, it will go to the end of main function and stop the whole program.

```
for(;;) {
PTJ ^= 0x01; // toggle LED
//SCI_OutString("huang hanxi 1303296 "); OutCRLF();
val=ATDDR0;
SCI_OutUDec(val);
OutCRLF();
msDelay(3); //sample frequency. 1/300

}
```

Fig.2.9. For loop

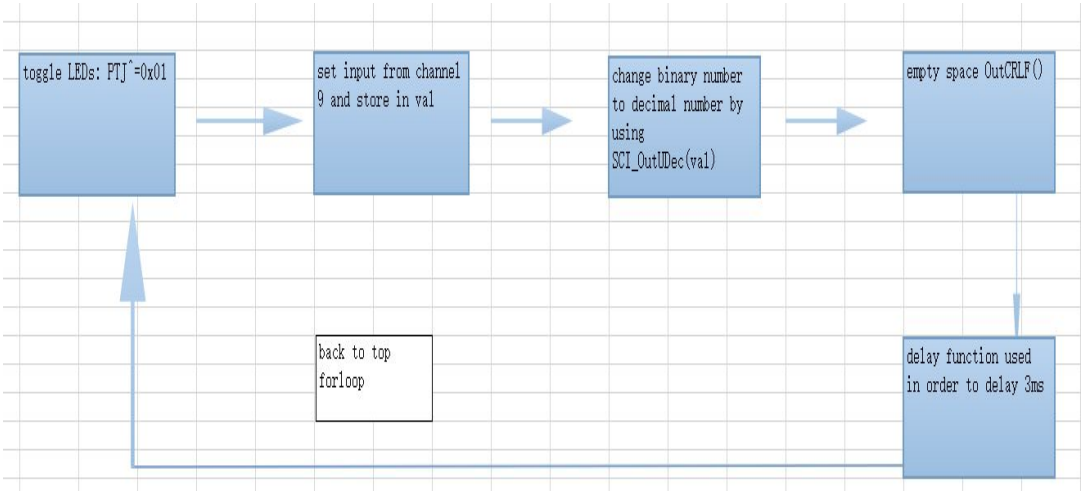

Fig.2.10 code flow chart

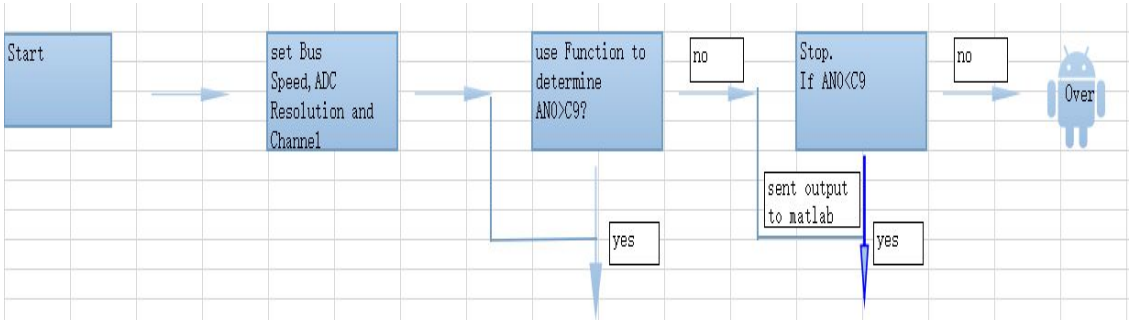

Fig.2.11 process flow chart

(f) Control/Communicate (Algorithm Flowchart)

✓ Introduce the computer and the program you are using

I use Lenovo X1 Carbon in the project. Actually, I used lab computer to do most of stuff. In the project, I used Matlab R2013A, WPS TABLE, WPS WORD, RealTerm, Analog, CodeWarrior IDE. In the following section, I may describe how each component do.

✓ Matlab

In matlab, I used Nyquist law. The Nyquist rate is twice the bandwidth of a bandlimited function or a bandlimited channel. I used Oscilloscope to analysis my data which was from code warrior. Fig.2.12 shows that the square wave, which is actually around 150 Hz frequency. It follows the Nyquist law to get half of sampling frequency.

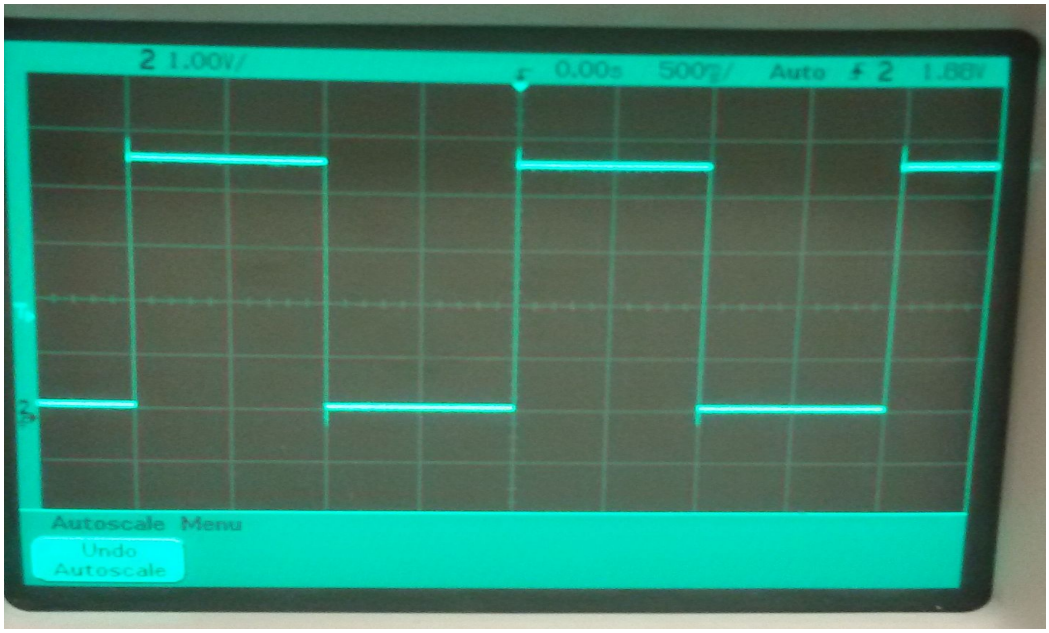

Fig.2.12. Square wave

Please refer to Appendix A matlab code, in while loop, the data was read by fgetl(s) function, which would return values and then plot the graph. The graph present the amount of sample number in x-axis and relative voltage value in y-axis.

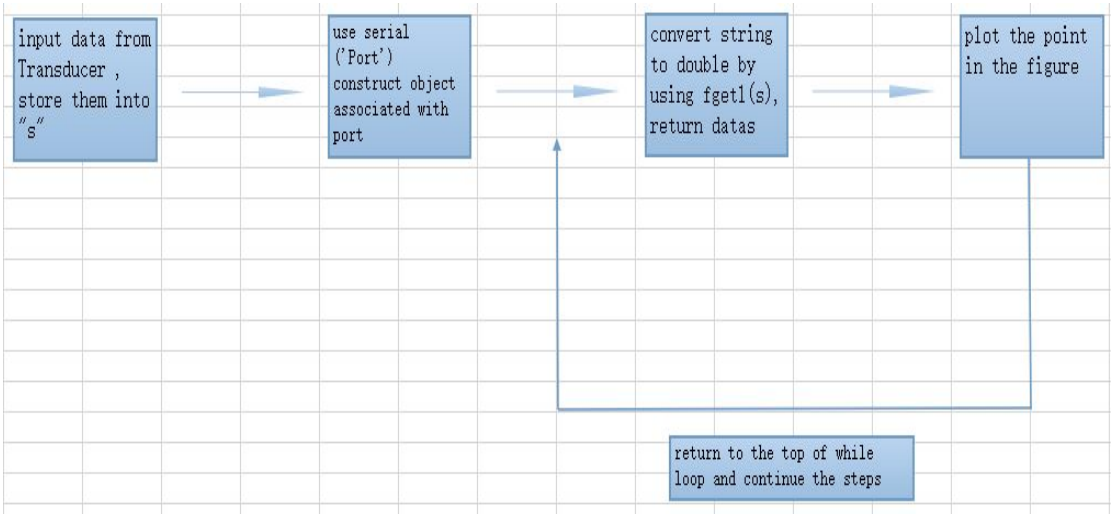

Fig.2.13

✓ CodeWarrior

I used CodeWarrior IDE to coding each registers should do and the delay time confirmation. It control the MicroControllor and connect the data of transducer and the Matlab output.

✓ RealTerm

Realterm was used in the beginning of the project in order to determine the circuit was right and the frequency and delay time were correct. It may give me a direct result to check the code and circuit.

✓ WPS

WPS was used to write the report and plot the flow chart.

(g) Full System Block Diagram

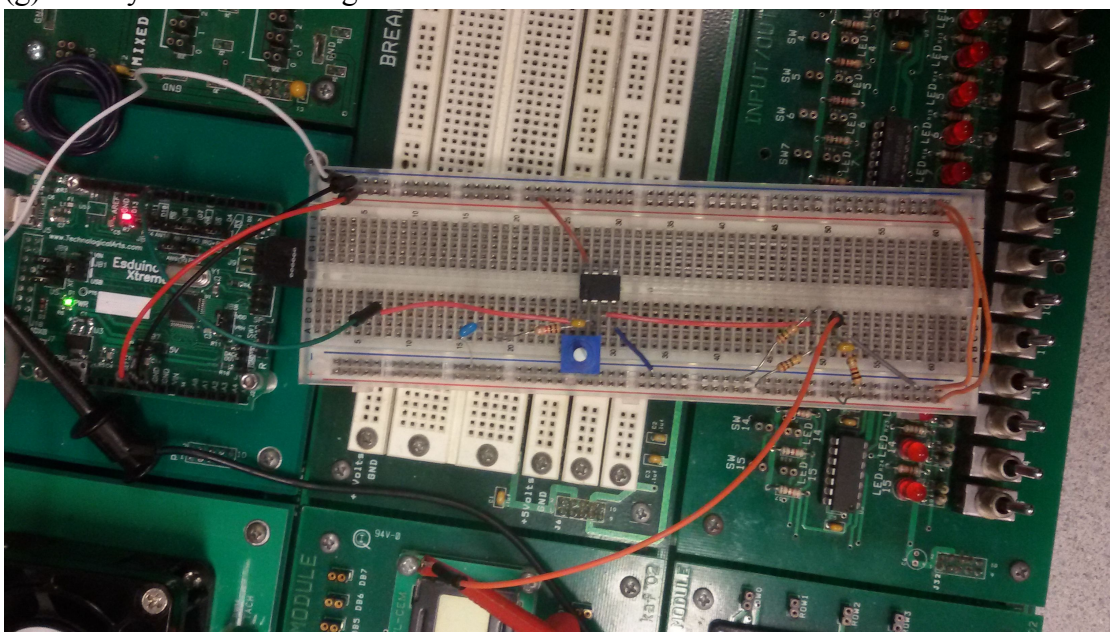

Fig.2.14(PC doesn't shown)

✓ What does each part doing?

The circuit is connect to the function generator. The green line connect to channel 9, the right red line connects to 5V in function generator. The left red line connects to ESDX 5V. The top white line and top black line both connectt to ground.

✓ How does it work?

The signal input from right red line. I didn't connect electric microphone here because I used function generator at interview. Then, the signal transfer to op-amp, which apply amplification and level shifting. After that, the signal may convert to digital data and go to computer.

(h) Full System Circuit Schematic

I talked it in the precondition and Transducer.

3. Result

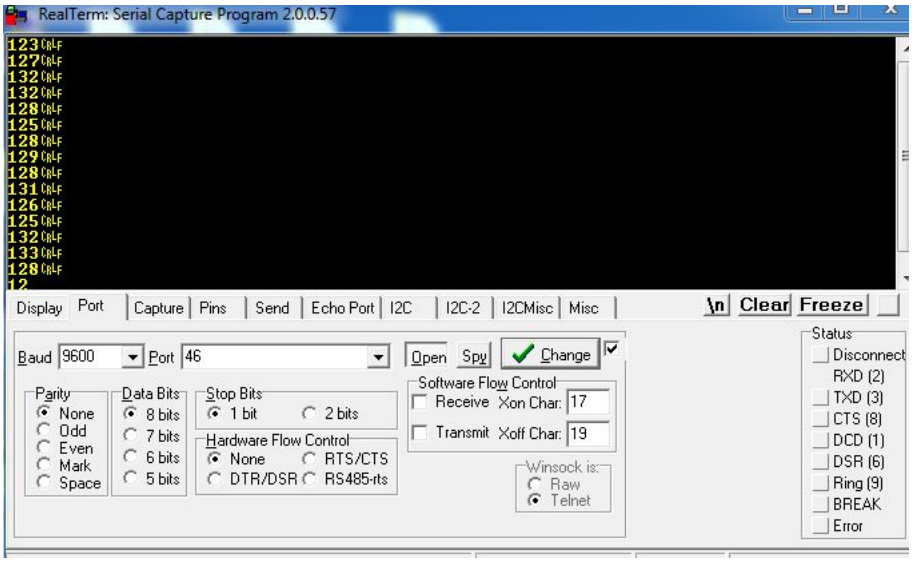

Fig.3.1 Testing the Circuit by Using Realterm

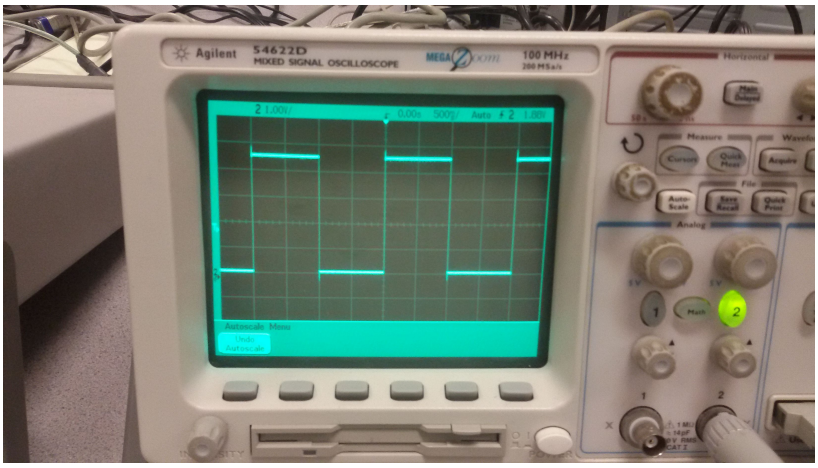

Fig.3.2 Square wave in Oscilloscope

✓ The previous graphs I explain it in ADC section.

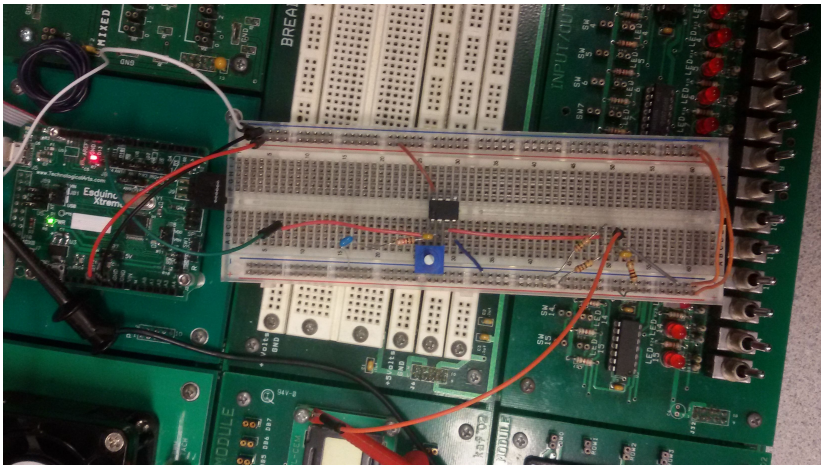

Fig.3.3 Circuit

✓ Serial communication on Matlab:

This three Matlab figures indicate Aliasing, which is an effect that causes different signals to become indistinguishable from each other during sampling. This is testing ADC and my circuit.For example, in fig.6.4, we can count for 40 peak to peak, which should multiply by 30 to get 1200Hz.
The 1200Hz is 200 Hz larger than the theoretical result. It may due to the loss of current and another loss in the circuit.The limitation of design is the signal may be not accurate because my wires are too long. It may cause the loss as well as inaccurate values. If I can improve my circuit, I will shorten the wire to avoid loss.

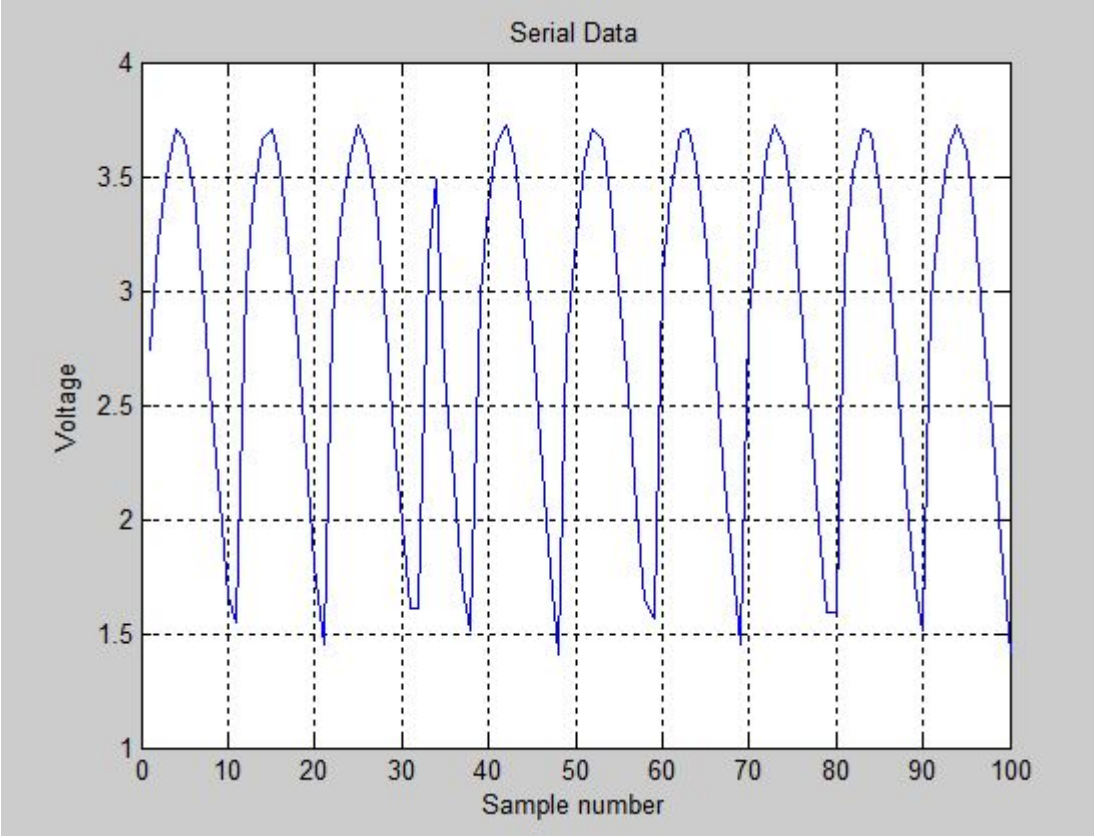

Fig.3.4 Matlab Graph in 150Hz

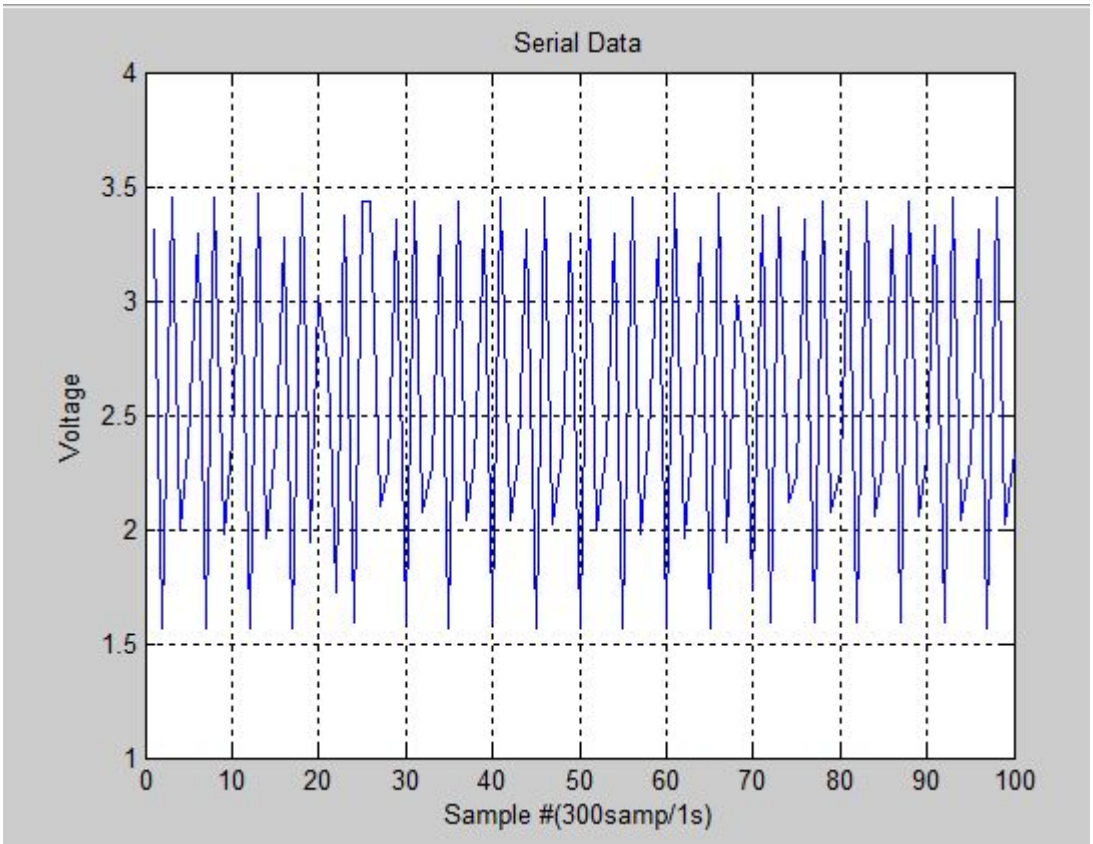

Fig.3.5

4. Discussion

1. Calculate your maximum quantization error.
The Theoretical value in my case is 2^8=256. The max quantization error is 5/256=0.0195V.

2. Based upon your assigned bus speed, what is the maximum standard serial communication rate you can implement.
My assigned Bus Speed is 4Mhz, and I chose baud rate to be 9600.

3. Reviewing the entire system, which element primarily limitation on speed? How did you test this?
I think it's baud rate. Baud Rate indicates bit per seconds, therefore, the larger the baud rate, the program will run more bit per seconds.

4. Based upon the Nyquist Rate, what is the maximum frequency of the analog signal you can effectively reproduce? What happens when your input signal exceeds this frequency?
My sampling frequency is 300Hz, so the maximum frequency is 150Hz.If my input

signal exceeds 150Hz, it will have aliasing.Aliasing is, the signal now takes on a different “persona," or a false presentation, due to being sampled at an insufficiently high frequency

5. User Manual

Check:

1.Please check your computer port number. You can open the Device manager in the control pannel, we need to remember the number and change code in matlab in sentence’COM’.

2.Please check the USB connection in order to avoid hardware problems.For example, the white wire USB which connects ESDX and computer.

Start:

Firstly, the circuit is connect to the function generator. The green line connects to channel 9, the right red line connects to 5V in function generator. The left red line connects to ESDX 5V. The top white line and top black line both connect to ground.

Secondly, run the code in codewarrior. You will see a window, please find a green arrow to start the code.

Then, please set the function generator. We can set frequency is 300Hz, peak to peak and offset to a very small number like 1mV.

After that, you can open matlab file to run the code.

Finally, in matlab, you will get a graph which is a sine wave that the result of the project.

After you finish all the above steps, please Log off the computer to clean up the memory to avoid occupying port.

6. Conclusion

The project indicate how the signal transfer and the relationship between digital data and computer. We used Analog to Digital method to imply the result. We apply the knowledge about codewarrior and realterm, it give us a visual result about the frequency. The result showed a sine wave about the frequency which was transferred from the microphone.When I was doing the measurement, I used analog and oscilliscope, which are both learn from another course. I found I learned some useful stuff in the last term.

Appendix

CodeWarrior Code

```
//file name Main.c//
//Student: HanXi Huang, Student Number: 1303296      Student ID: huangh32//
#include <hidef.h> /* common defines and macros */
#include "derivative.h" /* derivative information */
#include "SCI.h"
char string[20];
unsigned int val;

void delayby1ms(int);
void setClk(void);
//---------------------OutCRLF---------------------
// Output a CR,LF to SCI to move cursor to a new line
// Input: none
// Output: none
// Toggle LED each time through the loop
void OutCRLF(void){
SCI_OutChar(CR);
SCI_OutChar(LF);
PTJ ^= 0x20; // toggle LED D2
}

void main(void) {
// Setup and enable ADC channel 0
// Refer to Chapter 14 in S12G Reference Manual for ADC subsystem details

ATDCTL1 = 0x0F; // set for 8-bit resolution
ATDCTL3 = 0x88; // right justified, one sample per sequence
ATDCTL4 = 0x02; // prescaler = 2; ATD clock = 6.25MHz / (2 * (2 + 1)) == 1.04MHz
ATDCTL5 = 0x29;      // 00101001
setClk();

// Setup LED and SCI
DDRJ |= 0x01; // PortJ bit 0 is output to LED D2 on DIG13
SCI_Init(9600); //baud rate


for(;;) {
PTJ ^= 0x01; // toggle LED
//SCI_OutString("huang hanxi 1303296 ");
OutCRLF();
```

```
val=ATDDR0;
SCI_OutUDec(val);
OutCRLF();
delayby1ms(3); //sample frequency. 1/300

}
}

void delayby1ms(int k)
{
        int ix;
        TSCR1 = 0x90;   /* enable timer and fast timer flag clear */
        TSCR2 = 0x00;   /* disable timer interrupt, set prescaler to 1*/
        TIOS |= 0x01; /* enable OC0 */              //(not necessary)
        TC0 = TCNT + 4000;
        for(ix = 0; ix < k; ix++) {
                while(!(TFLG1_C0F));
                TC0 += 4000;
        }
        TIOS    &= ~0x01;     /* disable OC0 */    //(not necessary)
}

void setClk(void){
 //COMUCLKS=0x00;
 //CPMUSYNR=initSYNR+VCOPRQ;
 //CPMUREFDIV=initREFDV+REFFRQ;
 CPMUCLKS = 0x80;   //PLLSEL = 1
 CPMUOSC    = 0x00;   //OSCE = 0
 CPMUPOSTDIV=0x00;        //POST DIV is 0
 CPMUSYNR = 0x03;   //VCOFRQ = 0, SYNDIV = 3
 CPMUFLG    = 0x08; //LOCK=1

}
```

Matlab Code

```
clc
clear all
s=serial('COM48','BaudRate',9600,'Terminator','CR');
fopen(s);
t=100;
```

```
while 1
for n=1:t
val(n)=str2double(fgetl(s))*5/(2^8-1); %fgetl(s) returns a value in type
'str' as t increase by 1
plot(val);
title('Serial Data');%title
xlabel('Sample number'); % x-axis label
ylabel('Voltage value'); % y-axis label
axis([0, t, 1.5, 5]); % Set the limits for the x-axis which is x to x+50
and set the minimum y-axis limit which is 0 to 200.

grid
drawnow;%function
end

end

fclose(s);

delete(s);

clear s;
```

Reference

[1]:http://en.wikipedia.org/wiki/Analog-to-digital_converter,wikipedia,2010
[2]: lecture notes week9-3, ATD control registers