**CG2271 Real Time Operating Systems**

**Lab 3 – Analog Input**

**Answer Book**

**IMPORTANT:** YOU MUST PRINT OUT THE COMPLETED ANSWER BOOK AND BRING IT TO YOUR NEXT LAB SESSION!

|  |  |
| --- | --- |
| Name: Song Yangyu | Matric No: A0077863N |
| Name: Li Ruize | Matric No: A0078040M |

Question 1 (3 marks)

|  |  |
| --- | --- |
| Minimum Input Voltage: | 3.030303 v |
| Maximum Input Voltage: | 4.347826 v |

Question 2 (4 marks)

|  |  |
| --- | --- |
| Minimum Expected Value: | 620 |
| Maximum Expected Value: | 890 |

Question 3 (4 marks)

My code is pasted below:

int remap(int val, int min\_val, int max\_val){

if(val > max\_val){

return top;

} else if(val < min\_val){

return 0;

} else{

return (int)top \* ((float)(val-min\_val))/(max\_val - min\_val);

}

}

Question 4 (6 marks)

My code is pasted below:

void tone(int input){

int i;

// …- Other necessary code

int freRange = 500 - 100, freMin = 100;

int frequency = freMin + (int)(((float)input)/255 \* freRange);

int delayUnit = 1000/frequency;

int delayTime = delayUnit/2;

for(i=0; i<frequency; i++){

// Write a 1 to digital pin 13

PORTB |= 0b00100000;

// base delay

\_delay\_ms(delayTime);

// Write a 0 to digital pin 13

PORTB &= 0b11011111;

// base delay to generate the low part of the wave form

\_delay\_ms(delayTime);

}

}

Question 5 (10 marks)

My code is pasted below:

int main(){

int i;

int lowal, hival;

int input,tuneValue;

int min = 620, max = 890;

PRR &= 0b11111110;

ADCSRA = 0b10000111;

ADMUX = 0b01000001;

DDRB |= 0b00100000; // set output

while(1){

ADCSRA |= 0b01000000;

while(ADCSRA & 0x01000000);

ADCSRA |= 0b00010000;

lowal = ADCL; hival = ADCH;

input = ADCL + ADCH \* 256;

tuneValue = remap(input, min, max);

tone(tuneValue);

}

}

Question 6 (10 marks)

Changes that I made were:

1. Added an ISR function to handle the ADC interrupt (code below)
2. Added the sei() in main function, to globally switch on the interrupt
3. Move the “Starting conversion” code (ADCSRA |= 0b01000000) outside of the while loop, to start the ADC conversion

My code is pasted below:

ISR(ADC\_vect){

int lowal, hival;

int input,tuneValue;

int min = 620, max = 890;

lowal = ADCL; hival = ADCH;

input = lowal + hival \* 256;

tuneValue = remap(input, min, max);

tone(tuneValue);

ADCSRA |= 0b01000000;

}

int main(){

PRR &= 0b11111110;

ADCSRA = 0b10001111; // enable ADIE

ADMUX = 0b01000001; // use channel 1

sei();

DDRB |= 0b00100000; // set output

ADCSRA |= 0b01000000;

while(1){

}

}