**CG2271 Real Time Operating Systems**

**Lab 4 Answer Book**

Question 1 (3 marks)

Digital pin 11 on the Arduino corresponds to \_\_OC2A\_\_\_

Question 2 (5 marks)

If you answer to Question 1 is OC0A, then x=0 and y=A. If it is OC2B, then x=2 and y=B, etc.

Prescaler P=\_256\_\_\_

x=\_2\_\_ y=\_A\_\_\_

TCCR2A

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| PinNum | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| PinName | COM2A1 | COM2A0 | COM2B1 | COM2B0 | - | - | WGM01 | WGM00 |
| Value | 1 | 0 | - | - | - | - | 0 | 1 |

TCCR2B (Initial value)

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| PinNum | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| PinName | FOC2A | FOC2B | - | - | WGM02 | CS02 | CS01 | CS00 |
| Value | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |

TIMSK2

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| PinNum | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| PinName | - | - | - | - | - | OCIE2B | OCIE2A | TOIE2 |
| Value | - | - | -- | - | - | 0 | 1 | 0 |

Question 3 (6 marks)

My code for setupPWM() is shown below:

// Q3

void setupPWM(){

// set initial timer value

TCNT2 = 0;

// place TOP timer value to output compare register

OCR2A = 249;

// Set TCCR2A to set clear mode

// and choose mode 1 Phase correct PWM

TCCR2A = 0b10000001;

// enable compare interrupt

TIMSK2 |= 0b01;

}

Question 4 (6 marks)

The best vector is:

TIMER2\_OVF\_vect, because the overflow is periodic.

My code for the ISR is shown below:

// Q4

ISR(TIMER2\_OVF\_vect){

OCR2A = pwm\_val;

}

Question 5 (3 marks)

My code for startPWM():

// Q5

void startPWM(){

// set prescaler of 0b100

TCCR2B = 0b100;

// set global interrupts

sei();

}

Question 6 (8 marks)

// Q6.1

// set up ADC using interrupt programming

void setupADC(){

// disable power reduction on the ADC

PRR &= 0b11111110;

// prescaler value 64 = 16M/128K; enable ADC, interrupt;

ADCSRA = 0b10001110;

// set the voltage reference

ADMUX &= 0b01111111;

ADMUX |= 0b01000000;

}

// Q6.2

// starts ADC conversion for channel chan

void startADC(int chan){

if(chan == 0){

ADMUX &= 0b11111000;

} else{

ADMUX &= 0b11111001;

ADMUX |= 1;

}

\_chan = chan;

// then start conversion if it's not set

ADCSRA |= 0b01000000;

}

// Q6.3

ISR(ADC\_vect){

int adc\_val = ADCL + 255\*ADCH;

ADCSRA |= 0b01000000;

pwm\_val = adc\_val \* 255.0 / 1024;

if(\_chan == 0){

tune\_adc\_val = adc\_val;

startADC(1);

}else if(\_chan == 1){

pwm\_val = adc\_val \* 255.0 / 1024;

startADC(0);

}

}

Question 7 (7 marks)

int main(){

DDRB |= 0b00101000;

setupPWM();

startPWM();

setupADC();

startADC(0);

int min = 620, max = 890;

while(1){

tone(remap(tune\_adc\_val, min, max));

}

}

Question 8 (7 marks)

1. When setting up the PWM, add in the setting up for TCCR0A. Set the prescaler to be 1024, and OCR0A to be 155 so that it would trigger an interrupt after 156 counts.

Every 10ms means 100Hz, and 16,000,000 Hz / (156\*1024) = 100.16 Hz.

void setupPWM(){

// set initial timer value

TCNT2 = 0;

TCNT0 = 0;

// output compare register

OCR2A = 249;

OCR0A = 155;

// Set TCCR2A to set clear mode

// and choose mode 1 Phase correct PWM

TCCR2A = 0b10000001;

// set prescaler of 0b100

TCCR2B = 0b100;

// enable overflow interrupt

TIMSK2 |= 0b01;

// Set TCCR0A to CTC mode

TCCR0A = 0b00000010;

TCCR0B = 0b00000101; // prescaler to be 1024 (101)

// enable compare interrupt

TIMSK0 |= 0b10;

}

1. Add an interrupt handler to start the two ADC alternatively.

// use timer0 to read the input from potential meter

ISR(TIMER0\_COMPA\_vect){

if(is\_buzzer){

startADC(0);

} else{

startADC(1);

}

}

1. Also change accordingly when reading the ADC

// Q6.3

ISR(ADC\_vect){

int adc\_val = ADCL + 255\*ADCH;

if(is\_buzzer){

tune\_adc\_val = adc\_val;

}else{

pwm\_val = adc\_val \* 255.0 / 1024;

}

is\_buzzer = !is\_buzzer;

}

**Part 1 Demo (For TA)**

|  |  |
| --- | --- |
| **Criteria** | **Marks** |
| LED responds to pot input (3 marks max) |  |
| Buzzer responds to light changes (2 marks max) |  |
| Total: |  |

**Part 2 Demo (For TA)**

|  |  |
| --- | --- |
| **Criteria** | **Marks** |
| LED responds to pot input (3 marks max) |  |
| Buzzer responds to light changes (2 marks max) |  |
| Total: |  |

Report Total: \_\_\_\_\_/45

Demo Total: \_\_\_\_\_/10

Total: \_\_\_\_\_\_\_\_\_\_/55