**LAB**

**TIMER 1**

1. จงเขียนโปรแกรมเพื่อส่งพัลซ์ออกทางขา PD2 ความถี่ 20 Hz โดยใช้ Timer1 Overflow interrupt ให้เขียนด้วยภาษาซี

Fclk = 8 MHz, F = 20 Hz

T = 1/20 = 50 ms

IP = 25 ms

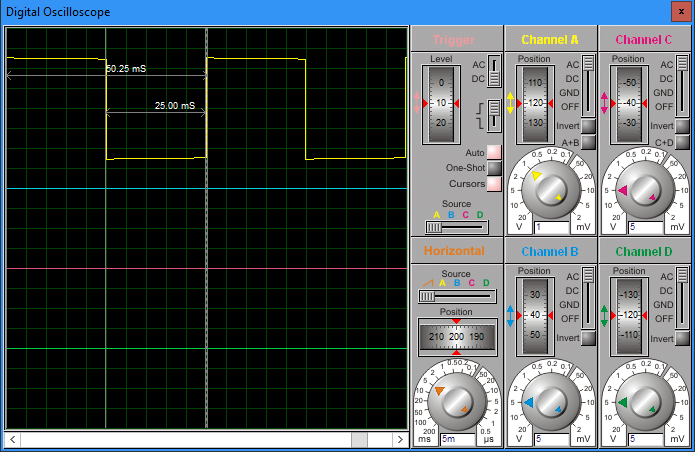
Prescaler = 256

Tcnt1 = 65536 – (CPUclk \* Ip) / N

Tcnt1 = 65536 – (8MHz \* 25ms) / 256

= 65536 – 781.25

= 64755



**Code**

#include <avr/io.h>

#include <avr/interrupt.h>

#define TIME 64755

int main()

{

DDRD = 0xff;

TIMSK1 = 0x01;

TCCR1A = 0x00;

TCCR1B = 0x04;

TCNT1 = TIME;

sei();

while (1);

return 0;

}

ISR(TIMER1\_OVF\_vect)

{

TCNT1 = TIME;

PORTD = ~PORTD;

}

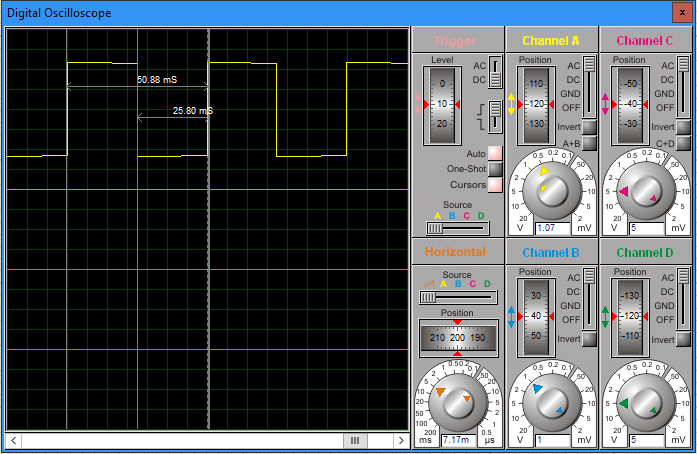
1. จงสร้างสัญญาณดังข้อ 1 โดยเปลี่ยนการทำงานมาใช้โหมด CTC โดยให้สัญญาณออกที่ขา OC1A ให้เขียนด้วยภาษาซี

Fclk = 8 MHz

OCR1A = Fclk / (2 \* n \* F0C0A) - 1

OCR1A = 8MHz / (2 \* 256 \*20) - 1

OCR1A = 780



**Code**

#include <avr/io.h>

int main()

{

DDRB = 0xff;

TIMSK1 = 0x01;

TCCR1A = 0x40; //0100 0000

TCCR1B = 0x0C; //0000 1100

TCNT1 = 0;

OCR1A = 780;

while (1);

return 0;

}

1. **จงใช้ AVR สร้างสัญญาณพัลซ์สี่เหลี่ยมความถี่ 60 Hz ออกทางขา PC0 และ PC1 โดยให้มีเฟสต่างกัน 90 องศา**

**f = 60**

**T = 1/60 = 16.7 ms**

**IP = 0.0167/4 = 4.175 ms**

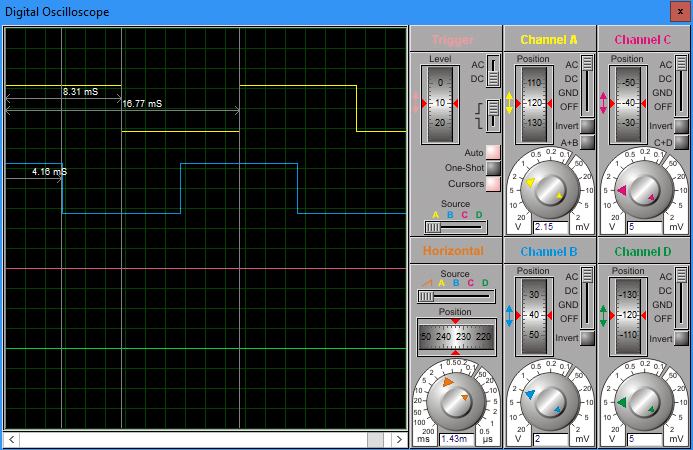
**01 => 11 => 00 => 10**

**กำหนดให้ใช้ Timer0 ใน normal mode**

TCNT0 = 256 – (CPUclk \* IP)/N

= 256 – (8M \* 4.175m) / 1024

TCNT0 = 256 – 32.62 = 223



**Code**

#include <avr/io.h>

#include <avr/interrupt.h>

unsigned char state;

//state 01->11->01->00

int main()

{

// Write your code here

DDRC = 0xff;

TIMSK0 = 0x01;

TCCR0A = 0x00; //0000 0000

TCCR0B = 0x05; //0000 0101

TCNT0 = 223;

PORTC = 0x02;

state = 1;

sei();

while (1);

return 0;

}

ISR(TIMER0\_OVF\_vect)

{

if(state == 1)

{

TCNT0 = 223;

PORTC = 0x02;

state = 2;

}

else if(state == 2)

{

TCNT0 = 223;

PORTC = 0x03;

state = 3;

}

else if(state == 3)

{

TCNT0 = 223;

PORTC = 0x01;

state = 4;

}

else

{

TCNT0 = 223;

PORTC = 0x00;

state = 1;

}

}

1. **ทำข้อ 1 ปรับ duty cycle = 10 + 2\*group**

Group = 8

Duty cycle = 10 + 2\*8 = 10 + 16 = 26 %

จาก ข้อ 1 f = 20 Hz , Prescaler = 256

T = 1/20 = 50 ms

Thigh = 50 \* 0.26

= 13 ms

Tlow = 50ms – 13 ms

= 37 ms

High -> TCNT0 = 65536 – (8M\*13m)/256

= 65536 – 406.25

= 65129.75

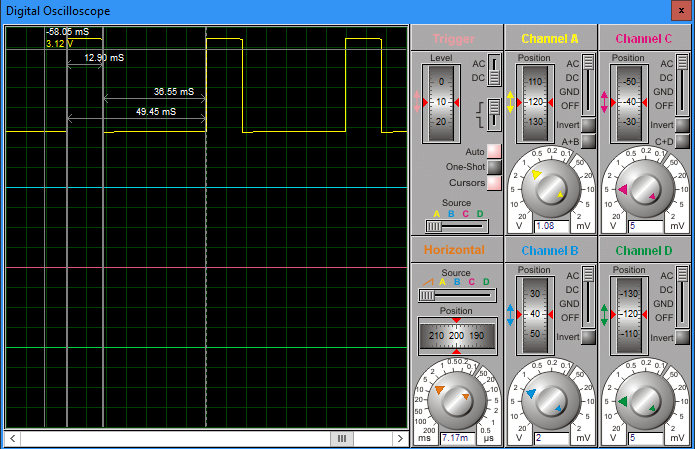
= 65130

Low -> TCNT0 = 65536 – (8M\*37m)/256

= 65536 – 1156.25

= 64379.75

= 64380



**Code**

#include <avr/io.h>

#include <avr/interrupt.h>

#define TIME\_HIGH 65130

#define TIME\_LOW 64380

unsigned char state;

int main()

{

DDRD = 0xff;

TIMSK1 = 0x01;

TCCR1A = 0x00;

TCCR1B = 0x04;

TCNT1 = TIME\_LOW;

state = 1;

sei();

while (1);

return 0;

}

ISR(TIMER1\_OVF\_vect)

{

if(state == 1)

{

PORTD = ~PORTD;

TCNT1 = TIME\_HIGH;

state = 2;

}

else

{

PORTD = ~PORTD;

TCNT1 = TIME\_LOW;

state = 1;

}

}

1. **จาก ข้อ 2 ให้ f = 50 + (group\*5) Hz**

**ใช้โหมด CTC โดยให้สัญญาณออกที่ขา OC1A ให้เขียนด้วยภาษาซี Fclk = 8 MHz**

Group = 8

F = 50 + (8\*5) = 50 + 40 = 90 hz

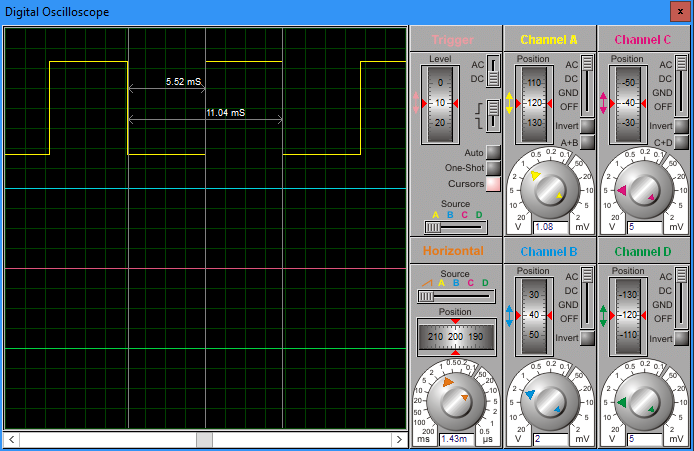
T = 1/90 = 11.11 ms

OCR1A = Fclk / (2 \* n \* F0C0A) - 1

OCR1A = 8MHz / (2 \* 256 \* 90) - 1

OCR1A = 172.61

= 173



**code**

#include <avr/io.h>

int main()

{

DDRB = 0xff;

TIMSK1 = 0x01;

TCCR1A = 0x40; //0100 0000

TCCR1B = 0x0C; //0000 1100

TCNT1 = 0;

OCR1A = 173;

while (1);

return 0;

}