

โค้ดภาษาซี

```

#include <avr/io.h>
#include <avr/interrupt.h>
#define F_CPU 16000000
#include <util/delay.h>

unsigned char COUNT, TIMER;
unsigned char LOOKUPTB[] = {
    0b00111111, 0b00000110, // 0, 1
    0b01011011, 0b01001111, // 2, 3
    0b01100110, 0b01101101, // 4, 5
    0b01111101, 0b00000111, // 6, 7
    0b01111111, 0b01101111, // 8, 9
    0b01110111, 0b01111100, // A, b
    0b00111001, 0b01011110, // c, d
    0b01111001, 0b01110001 // E, F
};

int main(void)
{
    DDRB = 0xFF;
    DDRD = 0xFB;
    PORTD = 0x04;
    COUNT = 0;
    EICRA = 0x02; // Falling Edge Interrupt
    EIMSK = 0x01; // Enable INT0
    sei();        // Enable Interrupt

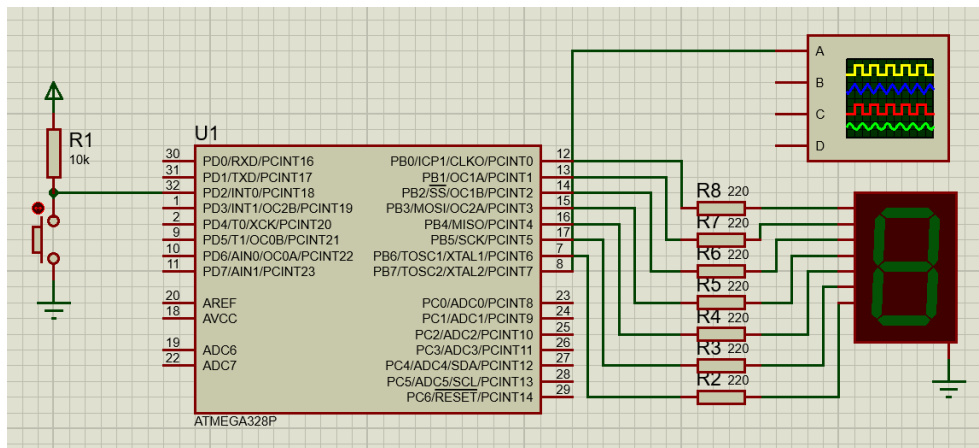
    while(1)
    {
        PORTB = LOOKUPTB[COUNT];
    }
}

ISR(INT0_vect)
{
    TIMER = 0;
    while(!(PIND & 0x04) && TIMER < 187)
    {
        _delay_ms(1);
        PORTB ^= 0x80; // Check 3 Sec
        TIMER++;
    }

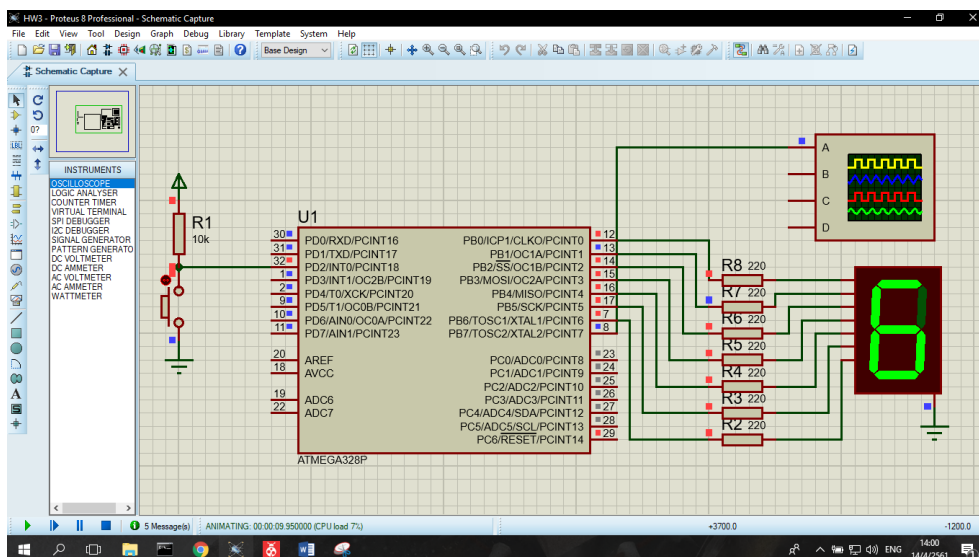
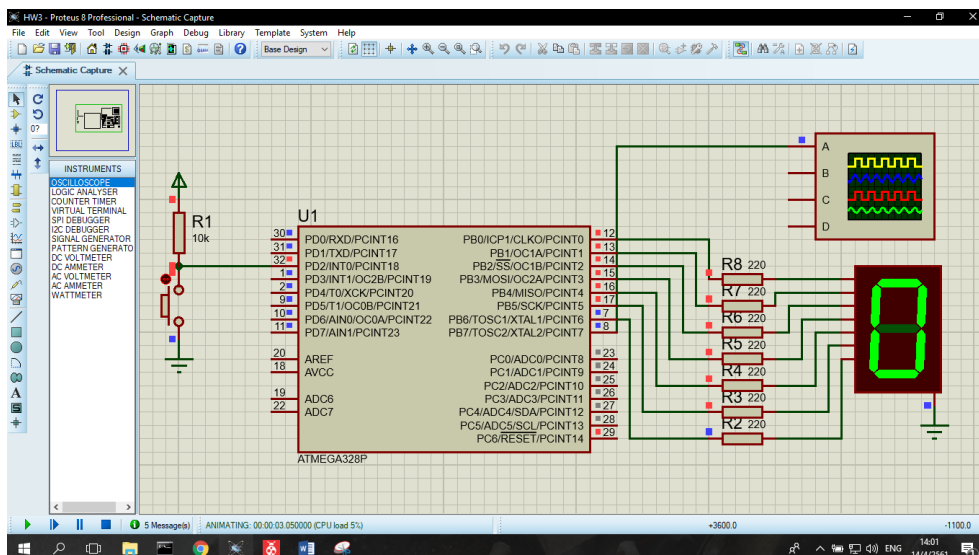
    if(TIMER == 187)
    {
        COUNT = 0;
    }
    else
    {
        COUNT++;
        if(COUNT == 16)
            COUNT = 0;
    }
}

```

ภาพวงจรบน Proteus



ภาพการทำงานของวงจรบน Proteus



SUBJECT:

NO:

DATE: / /

การทำงานของโปรแกรม

1. ตัวไมโครคอนโทรลเลอร์ B เป็นเอาต์พุต
2. ตัวไมโครคอนโทรลเลอร์ D บิตที่ 2 เป็นอินพุต และบิตอื่นๆ เป็นเอาต์พุต
3. สร้างตัวแปร COUNT ให้ค่าเริ่มต้นเป็น 0 เพื่อคำนวณการกด Push button
4. ตัวไมโครคอนโทรลเลอร์ INTO โดยตอบรับเมื่อเกิด Falling edge
5. ตัวไมโครคอนโทรลเลอร์ AVR
6. การทำงานของฟังก์ชันหลัก ให้นำค่าจาก LOOKUPB ตำแหน่งที่ COUNT ไม่เกิน 000 ที่ 7-segment ผ่านเอาต์พุต B
7. การทำงานของฟังก์ชันรองรับการเกิดอินเทอร์รัพต์
 - 7.1 ตั้งตัวแปร TIMER ให้นับเวลาในการกดปุ่ม
 - 7.2 มี while loop รอทำงาน เมื่อตรวจพบว่าปุ่มกดปุ่มอยู่ และ TIMER ยังไม่ครบเวลา ในลูป ทำการเพิ่มค่า TIMER และ delay 1 ms ทุกๆรอบ
 - 7.3 เมื่อเสร็จแล้วค่า TIMER ถึงค่าที่กำหนดหรือไม่ว่าง
 - 7.3.1 หากถึง ให้ค่า COUNT เป็น 0
 - 7.3.2 หากไม่ถึง ให้บวกค่า COUNT ขึ้น 1 หากมีค่าถึง 15 ก็จะทำให้การรีเซ็ตให้ COUNT เป็น 0

สรุป

การทำงานของโปรแกรมจากปุ่ม หากกดปกติ (short press) ให้เพิ่มค่าขึ้น 1 และ แสดง 000 บน 7-Segment คือ 0-F วนไปเรื่อยๆ และหากกดค้าง 3 วินาที (long press) ก็ให้ reset ค่า COUNT และ แสดง 000 เป็น 0 และมีการจัดการรองรับการกดปุ่ม โดยทำเมื่อเกิดการกดปุ่ม (Interrupt)

วิเคราะห์

ส่วนของตรวจสอบการกดปุ่ม 3 วินาที ใน while loop มี delay 1 ms ทุกๆ ครั้ง ที่ทำการเพิ่มค่า COUNT ค่าที่ COUNT จะนับได้ว่าเป็น 3 วินาที คือ COUNT เท่ากับ 3000 แต่ในการจำลองด้วย PROTEUS ค่า COUNT ไม่ถึง 187 ก็ให้ค่าการรอ 3 วินาที (ใช้ Oscilloscope รับสัญญาณได้ 3 วินาที)