

ЛАБОРАТОРНА РОБОТА 3

Взаємодія мікроконтролерів AVR із семисигментними індикаторами. Таймери.

Мета роботи: дослідити роботу семисегментних індикаторів в складі дисплея, навчитися використовувати таймер/лічильник та переривання від нього.

3.1 Короткі теоретичні відомості

Одним із способів виведення інформації є використання семисегментних індикаторів. Для цього можна скористатися мікроконтролером, який керуватиме роботою сегментів. Якщо використовується декілька індикаторів, то використовується мультиплексоване керування індикаторами. Світлодіоди в індикаторах для зменшення кількості зовнішніх контактів з'єднані одним із електродів: катодом чи анодом

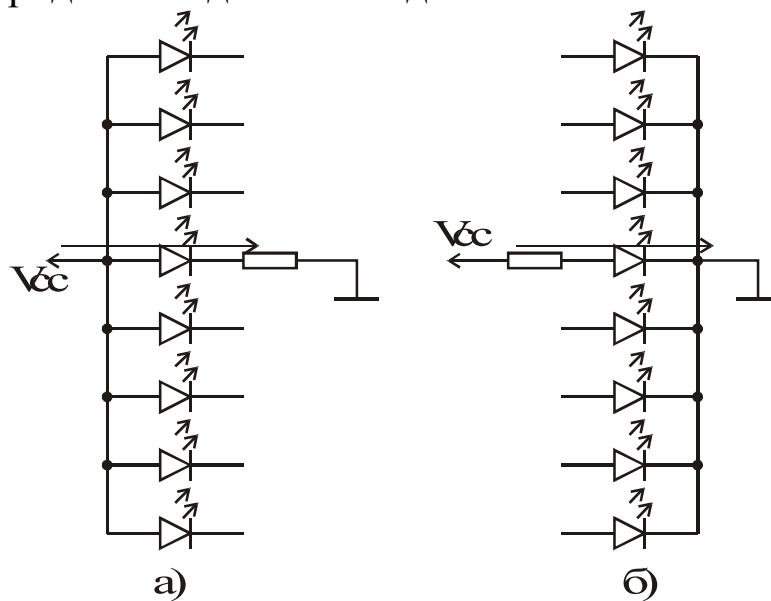


Рисунок Ошибка! Текст указанного стиля в документе отсутствует..б-Формування світлодіодів в семисегментному індикаторі

Вісім біт порта приєднують до катодів чотирьох семисегментних індикаторів через струмообмежуючі резистори (або до анодів і також через резистори). Інший загальний електрод анод (чи катод) приєднують до ключа і відповідного полюса живлення.

Наприклад використаємо індикатори із загальним анодом. Для цих індикаторів ключі приєднуються до Vcc і до відповідних

анодів а сегменти через струмообмежуючий резистор приєднують до контактів порта.

Порт є джерелом струму максимум 20 мА. Для управління катодами індикаторів цього буде достатньо однак треба передбачити струмообмежуючі резистори. Для індикаторів відомі допустимі пікові струми при мультиплексованому включенні. У статичному режимі струм для тієї ж яскравості свічення потрібен менший.

Якщо включаються всі сегменти сумарний струм через анад, буде перевищувати допустимий струм через контакт. Тому треба ставити підсилюючі транзистори в ключовому режимі.

У випадку використання семисегментних індикаторів із малим струмом засвічування, можна обійтися і без транзисторів. Важливо, щоб сумарний струм через контакт не перевищував 20 мА.

Для того, щоб послідовне засвічування індикаторів не були помітні треба, аби зміна проходила швидко.

Для того, щоб звільнити основну програму від постійної зміни цифри на індикаторі такі функції покладають на переривання, яке виникає при переповненні таймера/лічильника.

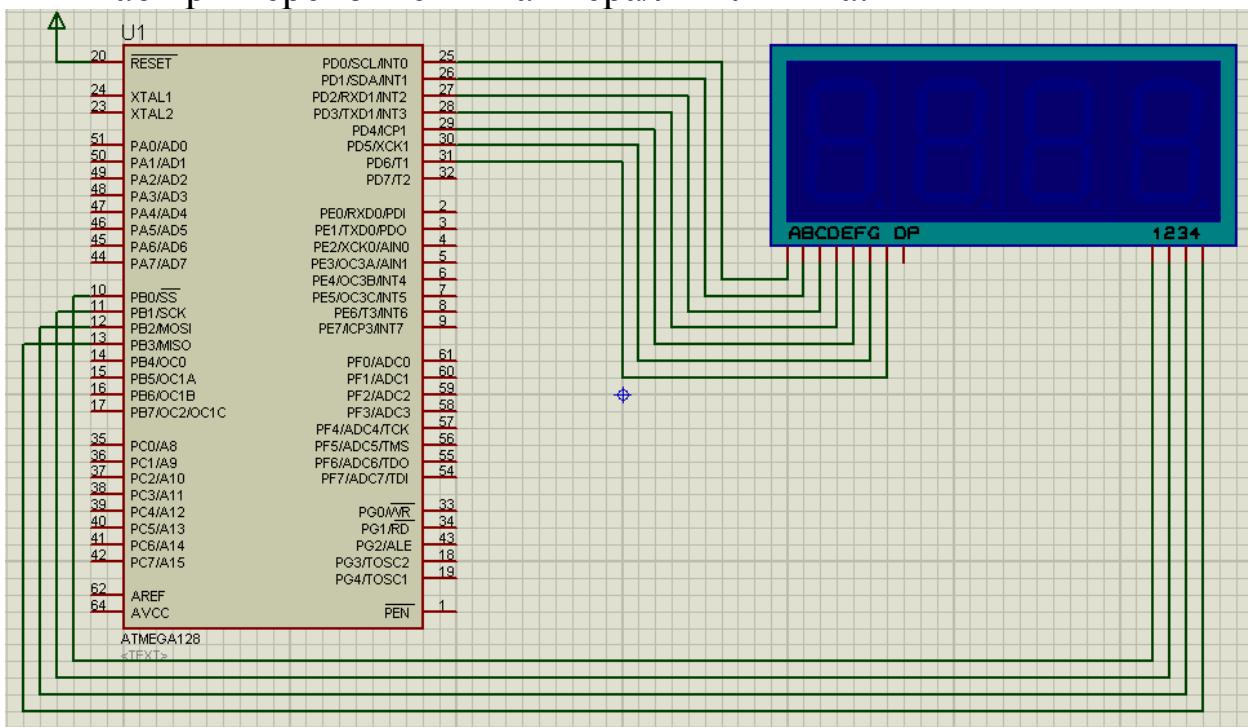


Рисунок Ошибка! Текст указанного стиля в документе отсутствует..7-Схема пристрою із чотирьохпозиційним семисегментним індикатором

Дана схема враховує особливості функціонування індикатора, який в симуляторі керується не струмом а логічними рівнями. Тому всі контакти індикатора приєднано до контактів портів безпосередньо. Індикатор– 7SEG-MPX4-CA-BLUE.

Для переключення режимів використовується таймер/лічильник 0 (TC0) із прескалером на 256. TC0 перегружається кожні 80 тактів і при частоті 4.096МГц це дає можливість отримувати період в 5 мс.

$$4096000/256=16000(\text{Гц})$$

щоб переривання виникало кожні 5мс, тобто з частотою 200 Гц, треба щоб лічильник рахував $16000/200=80$ тактів.

При перегрузці TC0 виникає переривання в якому через порт виводиться наступна цифра на індикатор. Основна програма формує масив цифр, які при перериванні просто береться із масиву цифр для дисплея.

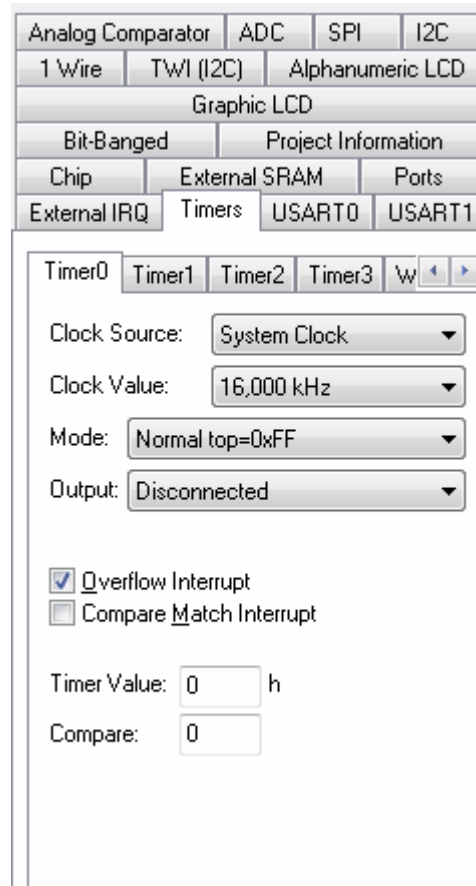
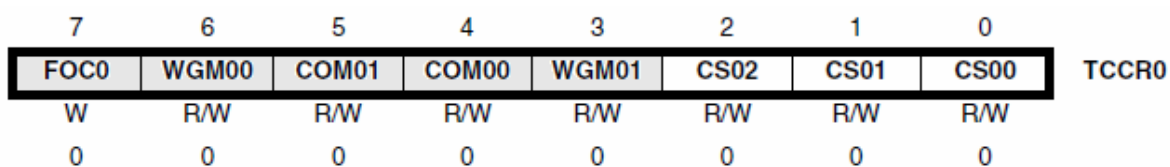


Рисунок 3.3 - Приклад налаштування у WizardAVR

Приклад програми яка при ініціалізації на екран виводиться число 0123, а через секунду 2480.

Для запуску таймера треба програмувати регістр управління таймером TCCR0.

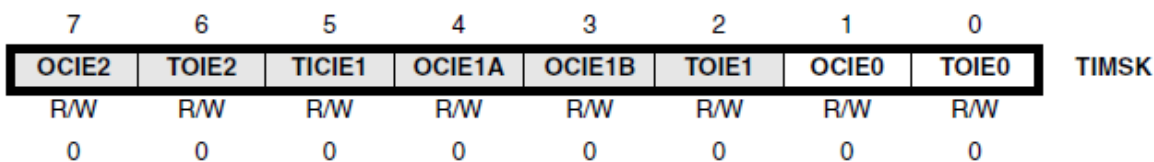


Біти CS02-CS00 мають наступні функції (табл.3.1)

Таблиця Ошибка! Текст указанного стиля в документе отсутствует..1- Функції бітів CS02-CS00

CS02	CS01	CS00	Опис
0	0	0	TC0 зупинено
0	0	1	TC0 працює з частотою тактового генератора (СК)
0	1	0	СК/8
0	1	1	СК/32
1	0	0	СК/64
1	0	1	СК/128
1	1	0	СК/256
1	1	1	СК/1024

Для виникнення переривання при переповненні таймера TC0 треба програмувати регістр-маску управління перериваннями TIMSK



TOIE0 – дозвіл переривання при переповненні TC0

OCIE0 – дозвіл переривання при співпаданні вмісту TCNT0 та OCR0

```

#include <mega128.h>
#include <delay.h>
//коди, які утворюють семисегментний код
char
decode[10]={0x3F,0x06,0x5B,0x4F,0x66,0x6D,0x7D,0x07,0x7F,0x6F};
char cdig;//номер відображуваної цифри на дисплеї
char dig[4]={0,1,2,3};//цифри на дисплеї

interrupt [TIM0_OVF] void timer0_ovf_isr(void){
    TCNT0=0xB0; //поновлення коду для дорахування
    if (cdig++==4) cdig=0;//вибір наступної цифри
    PORTD=~decode[dig[cdig]];//виведення коду цифри
    PORTB=PINB & 0xF0 | (1 << cdig); //виведення біта номера
цифри
}

```

```

void main(void) {
    PORTB=0x00; DDRB=0x0F; //порт номера цифри
    PORTC=0x00; DDRC=0x00;
    PORTD=0x00; DDRD=0xFF; //порт коду цифри
    //в контрольн.реєстрі TCCR0 є три біти прескалера. Тут
    прескалер на 256
    TCCR0=0x04; TCNT0=0xB0; //код для дорахування

    TIMSK=0x02;
    #asm("sei")
    while (1){
        delay_ms(1000);
        dig[0]=2; dig[1]=4; dig[2]=8; dig[3]=0; };
    }

```

Основну частину програму було згенеровано помічником CodeVision.

3.1 Програма роботи

Отримання у викладача допуску до виконання роботи, який передбачає перевірку наявності письмової домашньої підготовки, а також перевірку засвоєння основних теоретичних положень, необхідних для виконання роботи.

Отримання вказівок викладача стосовно виконання конкретної роботи, а також вхідних даних до її проведення.

Виконання лабораторної роботи згідно порядку, поданого в п. 3.3 вказівок до цієї роботи.

Захист лабораторної роботи, який передбачає подання письмового звіту про її виконання з поясненнями студента до отриманих результатів згідно запитань викладача а також демонстрацію роботи системи.

3.2 Порядок виконання роботи

Побудувати схему контролера. Вивести спочатку свій порядковий номер у списку із необхідною кількістю нулів спереду. Через секунду вивести число і місяць народження виконавця лабораторної роботи.

При написанні програми передбачити процедуру формування масиву цифр дисплея.

3.3 Форма звітності по роботі

Звіт по роботі повинен містити:

- тему роботи,
- завдання,

- принципову електричну схему із використанням реальних семисегментних індикаторів,
- алгоритм роботи програми,
- текст програми,
- скріншот у програмі стимулятора.

3.4 Контрольні запитання до роботи

У яких режимах можуть працювати таймери/лічильники AVR?

У чому суть режиму захоплення таймера/лічильника?

Який принцип формування цифри на семисегментному індикаторі?

Який принцип формування зображення на 7-ми сегментному дисплеї ?

Що може бути джерелом переривання в МК АТМega128?

Які регістри програмують роботу таймера/лічильника?

Тривалість заняття: 4 год.