

Лабораторне заняття 1

Організація введення інформації за допомогою клавіатури

Мета: ознайомитися з принципами побудови клавіатури. Отримати практичні навички по підключенню клавіатури до мікроконтролерів та організації її опитування.

Теоретичні відомості

Для реалізації взаємодії користувача з мікропроцесорною системою (МПС) використовують різні пристрої введення-виведення інформації. У найпростішому випадку в ролі пристрої введення може виступати кнопка, що представляє собою елементарний механізм, який здійснює замикання-розмикання контактів під дією зовнішньої механічної сили.

У даній роботі розглянемо роботу з такими пристроями як перемикачі, кнопки, клавіатура.

Перемикач (рисунок 1.1) – це електромеханічний пристрій для з'єднання і роз'єднання електричних ланцюгів. Існують різні перемикачі, але типовий механічний перемикач складається з клем, які можна замикати між собою механічно.

Кнопочный переключатель	Тумблерный переключатель	Клавишный переключатель	Микро-переключатель	DIL-переключатель
				
				

Рисунок 1.1 – Позначення перемикачів

Кнопочний перемикач (кнопка) – механічний пристрій для передачі сигналу (введення інформації), елемент інтерфейсу людина-машина: елементарний фізичний механізм передачі електричного сигналу різних пристроїв шляхом замикання або розмикання двох або більше контактів. По суті своїй є датчиком зовнішнього фізичного впливу (зусилля натискання), що передає далі факт фізичного впливу до МК.

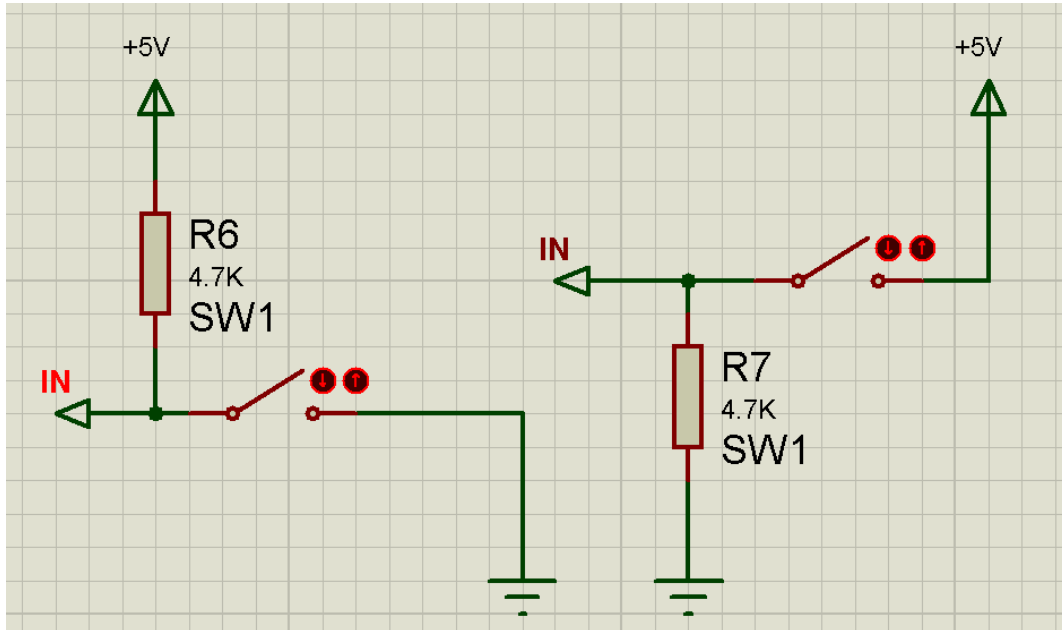
При використанні перемикача в якості датчика у мікроконтролера, один контакт перемикача з'єднується з виводом мікроконтролера, який позначається в програмі як вхід. Коли відбувається з'єднання контакту з землею або електричним потенціалом, змінюється так само значення біта шини відповідне введення мікроконтролера. У зв'язку з електромагнітними перешкодами нікуди не підключений вивід може в будь-який момент отримати довільне значення «0» або «1».

Один з головних методів для уникнення невизначених станів – це підключення виведення мікроконтролера через опір з землею або потенціалом джерела живлення (рисунок 1.2). Коли контакти кнопки SW1 (рисунок 1.2а) розімкнуті через резистор R6 на вхід контролера надходить високий логічний рівень "1", коли ж контакти замкнуті, то вхід виявляється з'єднаним із загальним проводом, що відповідає логічному рівню "0". Якщо паралельний порт мікроконтролера має вбудований генератор струму (підтягаючий резистор), то в схемі можна обійтися без резистора R6.

Опір в цій схемі на англійській мові називають pull-down або pull-up опором. Зазвичай опір pull-up або pull-down резисторів коливається від декількох кіло до одного мегаом. У більшості випадках використовують варіант а).

У зазначеній вище схемі використовувався зовнішній резистор для формування певного рівня сигналу на виводі цифрового порту (варіант а), але у сімейства МК ATMEGA вбудовано внутрішній опір. Для його використання потрібно при налаштуванні вивід, до якого підключили кнопку, перевести у режим INPUT_PULLUP замість простого INPUT:

```
pinMode(PIN_BUTTON, INPUT_PULLUP);
```



а)

б)

Рисунок 1.2 – Підключення перемикача

Якщо необхідно підключити велику кількість кнопок (клавіатуру), то даний варіант не використовується, тому що необхідна велика кількість виводів мікроконтролера. Замість цього використовують матричне включення кнопок (рисунок 1.3 та рисунок 1.4):

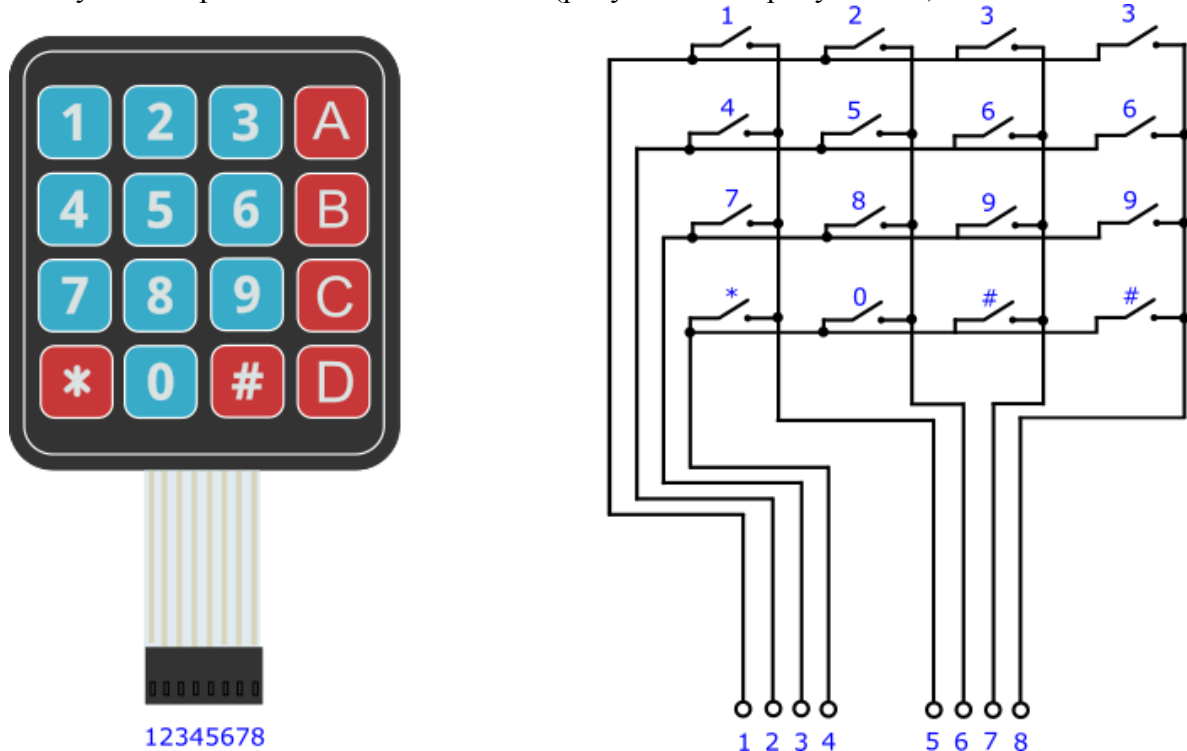


Рисунок 1.3 – Організація матричного підключення кнопок

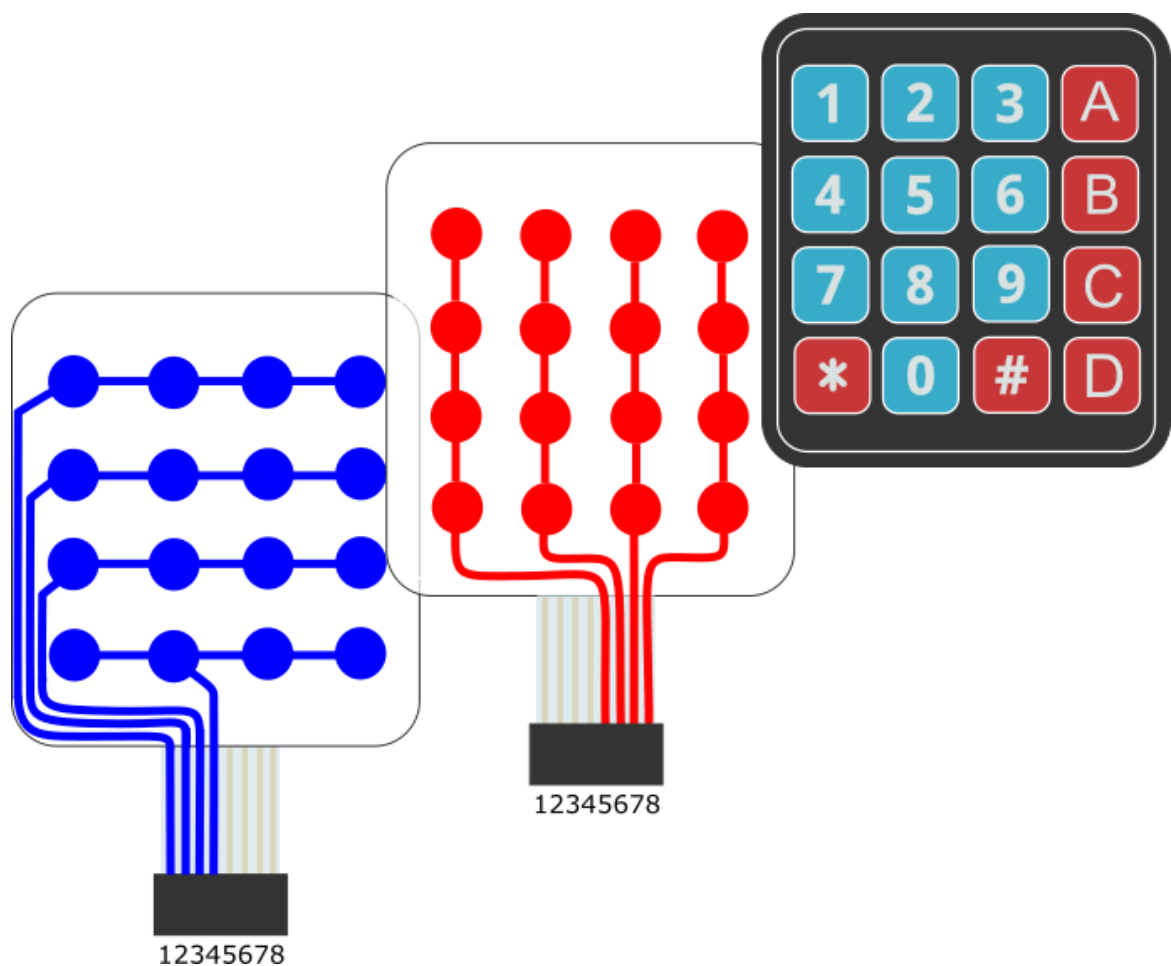


Рисунок 1.4 – Конструкційне виконання плівкової клавіатури

Принцип роботи такої клавіатури досить простий. МК по черзі подає логічний «0» на кожен з виводів 4х рядків, а з виводів стовпців навпаки – зчитує значення. Якщо натиснути будь-яку з кнопок, то вона замкне вивід рядка і вивід стовпчика, з якими пов'язана, в результаті чого на відповідному виводі стовпчика з'явиться логічний «0». Наприклад, якщо натиснути кнопку S7 (рисунок 1.5), то вона замкне виводи "Рядок 2" і "Стовпець 3". Неважко здогадатися, що в такому випадку, при подачі на "Рядок 2" логічного «0», через кнопку S7 логічний «0» встановиться і на виводі "Стовпець 3".

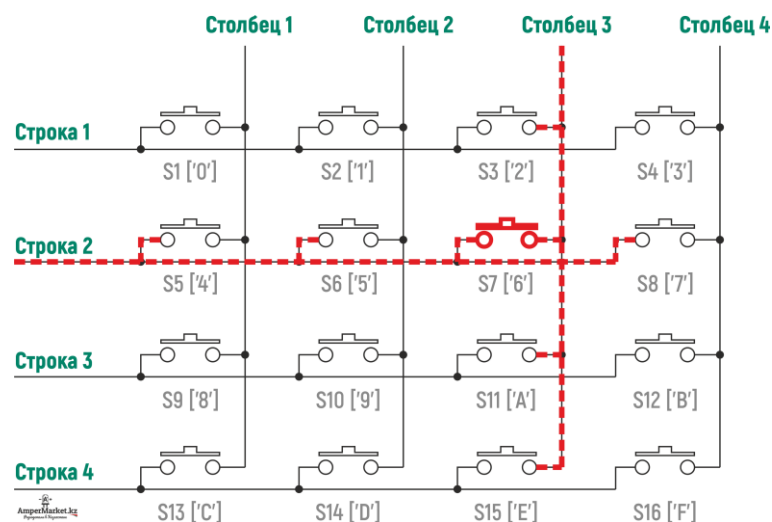


Рисунок 1.5 – Принцип роботи матричної клавіатури

Електрична схема підключення матричної клавіатури приведена на рисунку 1.6, а зовнішній вигляд підключення до Arduino представлено на рисунку 1.7.

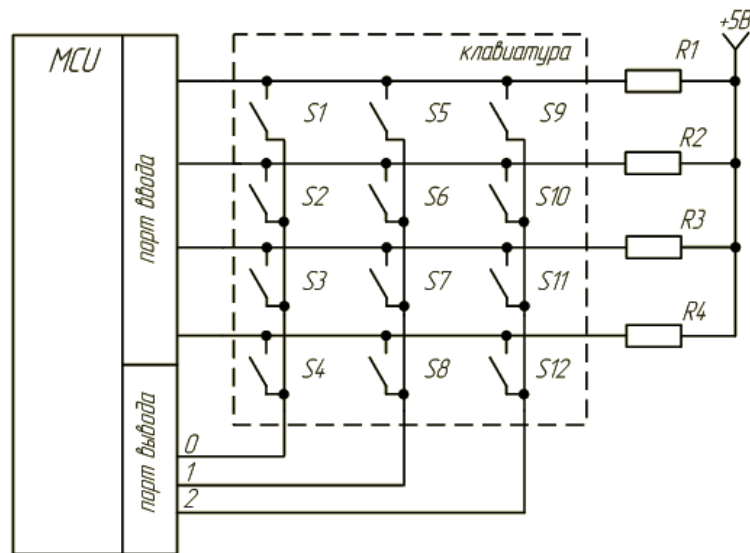


Рисунок 1.6 – Схема підключення матричної клавіатури до МК

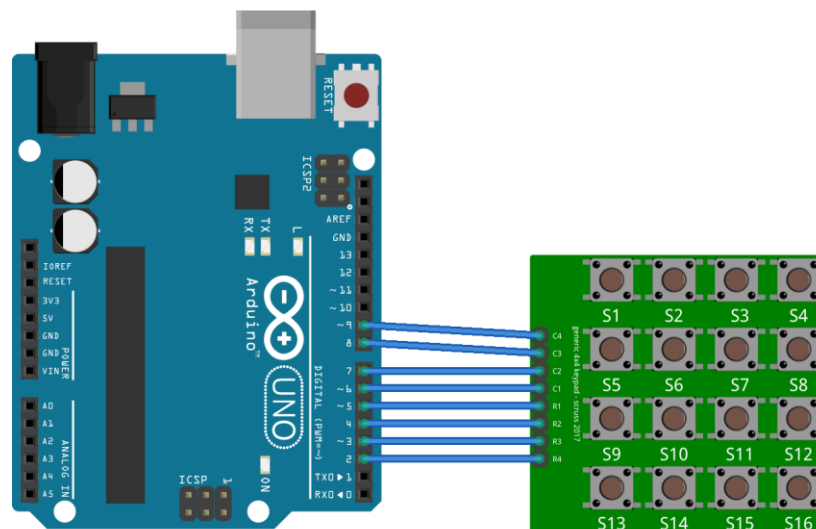


Рисунок 1.7 – Схема підключення матричної клавіатури до Arduino

Розглянемо організацію введення інформації про стан однієї кнопки (рисунок 1.2a) в середовищі Arduino IDE. Як і висновок, введення також ділиться на три етапи:

- 1) Оголошення змінних, відповідних виводам Arduino;
- 2) Ініціалізація виводів як вхід;
- 3) Введення інформації.

Для наочності наведемо приклад програми (рисунок 1.8), яка управляє включенням світлодіода при натисканні на кнопку.

```

1  int ledPin = 13; /* объявление переменной для вывода.
2                      13 пин - светодиод на плате Arduino */
3  int inPin = 7;    /* оголошення змінної для вводу */
4  boolean val = 0;  /* обязательный блок инициализации портов */
5
6  void setup()
7  {
8      pinMode(ledPin, OUTPUT); /* инициализация ledPin как выхода */
9      pinMode(inPin, INPUT);   /* инициализация inPin как входа */
10 }
11
12 void loop()
13 {
14     val = digitalRead(inPin); /* считать значение с 7 пина */
15     digitalWrite(ledPin, val); /* установит значение из val в ledPin */
16 }

```

Рисунок 1.8 - Тестовий приклад

Кожен вивід мікроконтролера може виконувати одну з декількох функцій, тому порти необхідно налаштувати. Налаштування портів можна виконати як за допомогою функцій Arduino, так і за допомогою налаштування регістрів МК (рисунок 1.9).

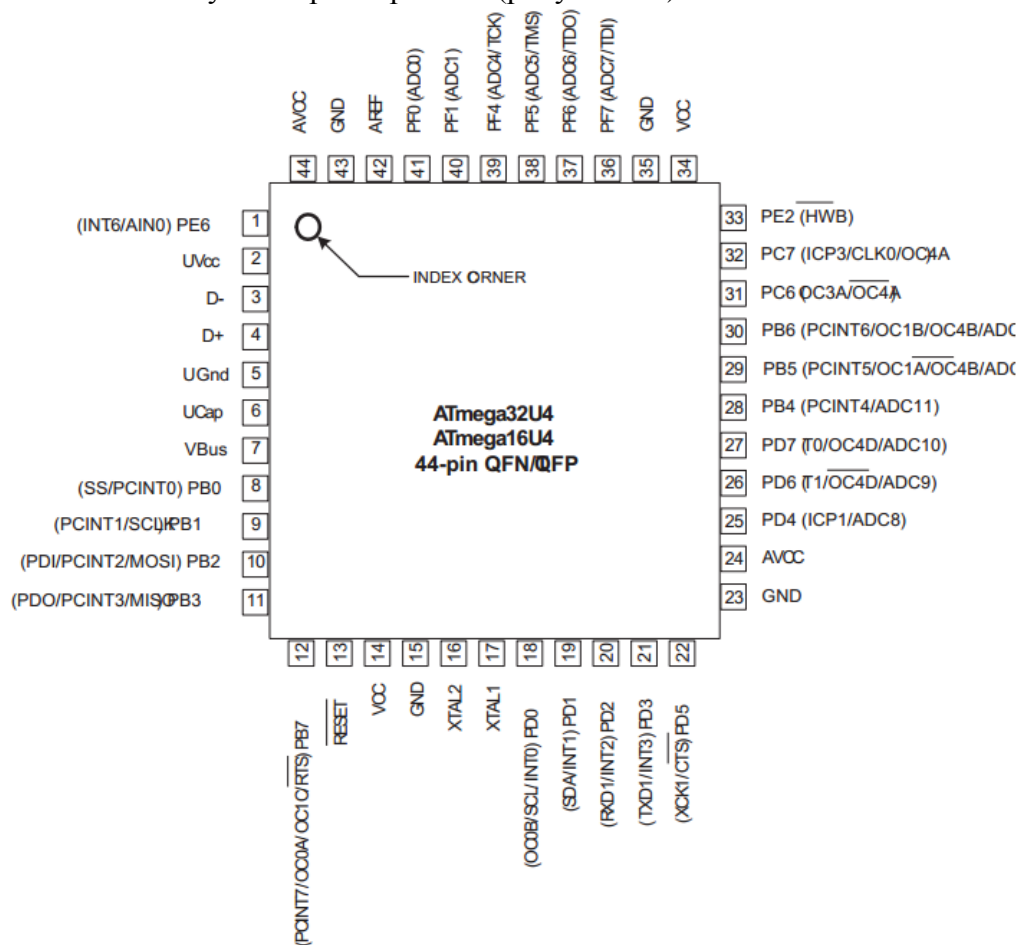


Рисунок 1.9 – Призначення виводів МК

В адресному просторі вводу/виводу для кожного порту відведено по 3 регістри:

- DDRx - напрямку роботи (ввод/вивід);
- PORTx – вихідних даних;
- PINx – вхідних даних.

Наприклад, для порту A - DDRA, PORTA, PINA (рисунок 1.10).

Розряди цих регістрів мають назви:

- DDx7... DDx0 – для регістрів DDRx;
- Px7... Px0 – для регістрів PORTx;
- PINx7... PINx0 – для регістрів PINx.

Регістри PINx дозволяють здійснити доступ до фізичних значень сигналів на виводах порту. Відповідно вони доступні лише для читання. Регістри PORTx та DDRx доступні як для читання, так і для запису. Після рестарту мікроконтролера в регістри PORTx та DDRx записуються початкові нульові значення. Це відповідає режиму приймача.

Запис у порт означає запис необхідного стану для кожного виводу порту у відповідний регістр даних порту PORTx. А читання стану порту виконується читанням або регістра даних порту PORTx, або регістра виводів порту PINx. При читанні регістра виводів порту PINx відбувається зчитування логічних рівнів сигналів, що присутні на виводах порту. А при читанні регістра даних порту відбувається зчитування даних, що знаходяться у регістрі PORTx.

Регістри портів розташовані в сегменті регістрів вводу/виводу, для якого можливі бітові маніпуляції. Це суттєво спрощує роботу з окремими лініями портів.

Address	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Page
0x17 (0x37)	Reserved	-	-	-	-	-	-	-	-	
0x16 (0x36)	TIFR1	-	-	ICF1	-	OCF1C	OCF1B	OCF1A	TOV1	
0x15 (0x35)	TIFR0	-	-	-	-	-	OCF0B	OCF0A	TOV0	
0x14 (0x34)	Reserved	-	-	-	-	-	-	-	-	
0x13 (0x33)	Reserved	-	-	-	-	-	-	-	-	
0x12 (0x32)	Reserved	-	-	-	-	-	-	-	-	
0x11 (0x31)	PORTF	PORTF7	PORTF6	PORTF5	PORTF4	-	-	PORTF1	PORTF0	
0x10 (0x30)	DDRF	DDF7	DDF6	DDF5	DDF4	-	-	DDF1	DDF0	
0x0F (0x2F)	PINF	PINF7	PINF6	PINF5	PINF4	-	-	PINF1	PINF0	
0x0E (0x2E)	PORTE	-	PORTE6	-	-	-	PORTE2	-	-	
0x0D (0x2D)	DDRE	-	DDE6	-	-	-	DDE2	-	-	
0x0C (0x2C)	PINE	-	PINE6	-	-	-	PINE2	-	-	
0x0B (0x2B)	PORTD	PORTD7	PORTD6	PORTD5	PORTD4	PORTD3	PORTD2	PORTD1	PORTD0	
0x0A (0x2A)	DDRD	DDD7	DDD6	DDD5	DDD4	DDD3	DDD2	DDD1	DDD0	
0x09 (0x29)	PIND	PIND7	PIND6	PIND5	PIND4	PIND3	PIND2	PIND1	PIND0	
0x08 (0x28)	PORTC	PORTC7	PORTC6	-	-	-	-	-	-	
0x07 (0x27)	DDRC	DDC7	DDC6	-	-	-	-	-	-	
0x06 (0x26)	PINC	PINC7	PINC6	-	-	-	-	-	-	
0x05 (0x25)	PORTB	PORTB7	PORTB6	PORTB5	PORTB4	PORTB3	PORTB2	PORTB1	PORTB0	
0x04 (0x24)	DDRB	ddb7	ddb6	ddb5	ddb4	ddb3	ddb2	ddb1	ddb0	
0x03 (0x23)	PINB	PINB7	PINB6	PINB5	PINB4	PINB3	PINB2	PINB1	PINB0	
0x02 (0x22)	Reserved	-	-	-	-	-	-	-	-	
0x01 (0x21)	Reserved	-	-	-	-	-	-	-	-	
0x00 (0x20)	Reserved	-	-	-	-	-	-	-	-	

Рисунок 1.10 – Розподілення адресів регістрів (частина)

Налаштування регістрів портів здійснюється відповідно до рисунку 1.11.

DDxn	PORTxn	PUD (in MCUCR)	I/O	Pull-up	Comment
0	0	X	Input	No	Tri-state (Hi-Z)
0	1	0	Input	Yes	Pxn will source current if ext. pulled low
0	1	1	Input	No	Tri-state (Hi-Z)
1	0	X	Output	No	Output Low (Sink)
1	1	X	Output	No	Output High (Source)

Рисунок 1.11 – Налаштування регістрів портів

Зміст роботи

Завдання 1: Реалізувати опитування однієї кнопки клавіатури (у відповідності варіанту див. таблицю 1.1). При натисканні виконати зсув інформації на семисегментному індикаторі та виведення коду натиснутої кнопки у шістнадцятковій системі в останньому розряді індикатору.

Таблиця 1.1

Варіант	Кнопка	Код кнопки
1, 17	S1	0
2, 18	S2	1
3, 19	S3	2
4, 20	S4	3
5, 21	S5	4
6, 22	S6	5
7, 23	S7	6
8, 24	S8	7
9, 25	S9	8
10, 26	S10	9
11, 27	S11	A
12, 28	S12	B
13, 29	S13	C
14, 30	S14	D
15, 31	S15	E
16, 32	S16	F

Завдання 2: Реалізувати опитування усіх кнопок клавіатури. При натисканні виконати зсув інформації на семисегментному індикаторі та виведення коду натиснутої кнопки у шістнадцятковій системі в останньому розряді індикатору (див. таблицю 1.1).

Завдання 3 (додаткове на оцінку відмінно): Виконати завдання 2, виконуючі роботу з портами за допомогою регістрів МК.