



ÓBUDAI EGYETEM  
NEUMANN JÁNOS INFORMATIKAI KAR

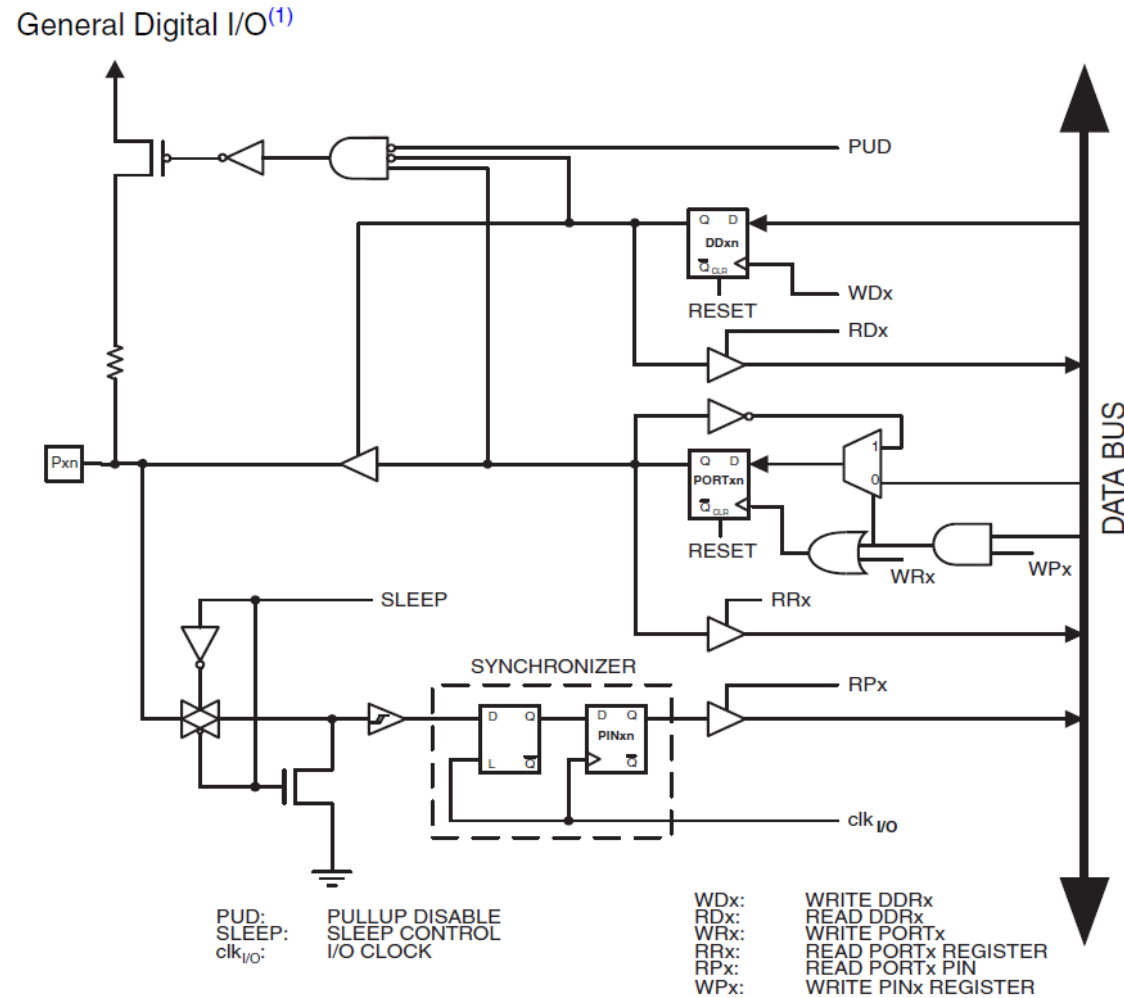
# GPIO

ATMega2560 mikrovezérlő

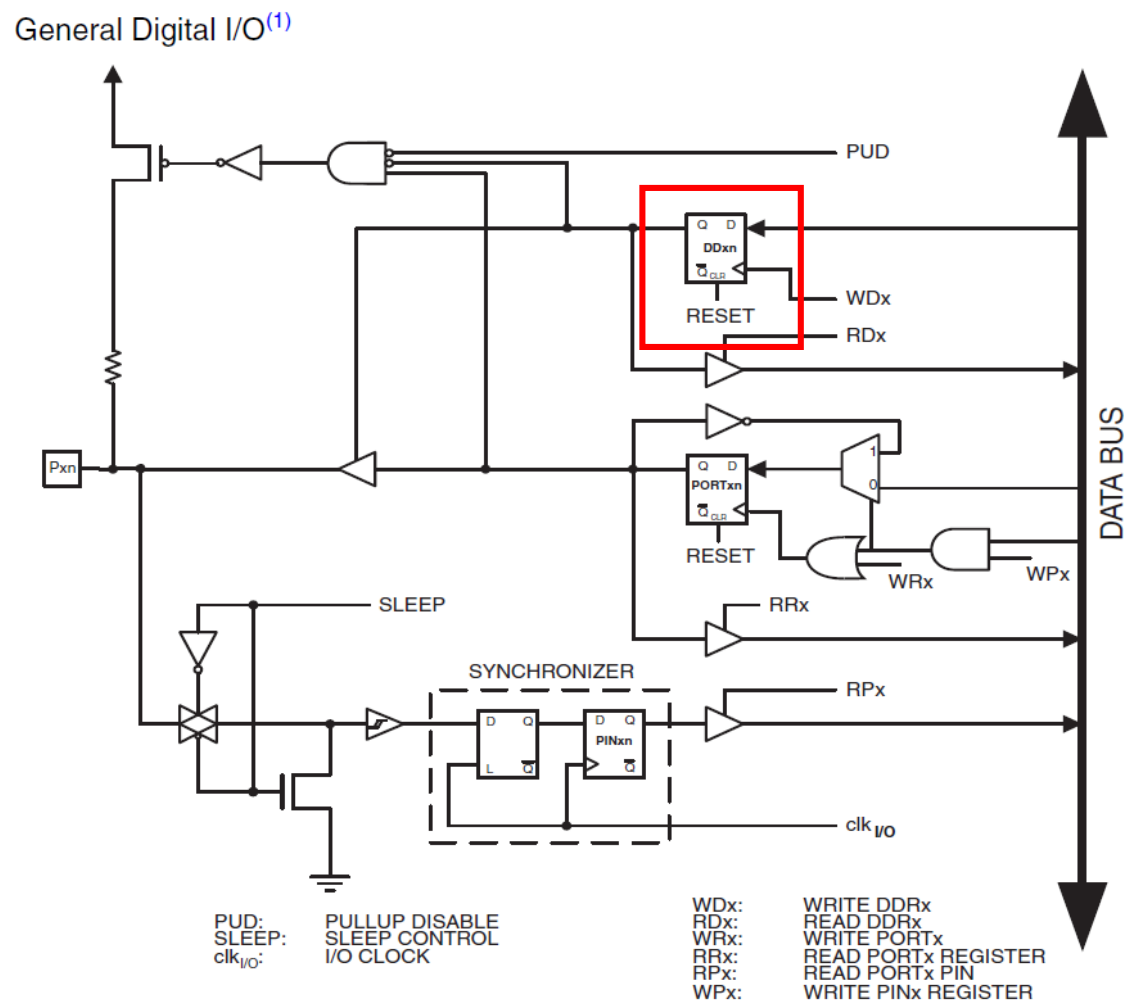
# Bevezetés

- Az ATMEGA 2560 mikrovezérlő 86db GPIO-val rendelkezik, melyek 11db regiszterhez tartoznak:
  - 1. PORTA (PA0..PA7),
  - 2. PORTB (PB0..PB7),
  - 3. PORTC (PC0..PC7),
  - 4. PORTD (PD0..PD7),
  - 5. PORTE (PE0..PE7),
  - 6. PORTF (PF0..PF7),
  - 7. PORTG (PG0..PG5),
  - 8. PORTH (PH0..PH7),
  - 9. PORTJ (PJ0..PJ7),
  - 10. PORTK (PK0..PK7),
  - 11. PORTL (PL0..PL7)

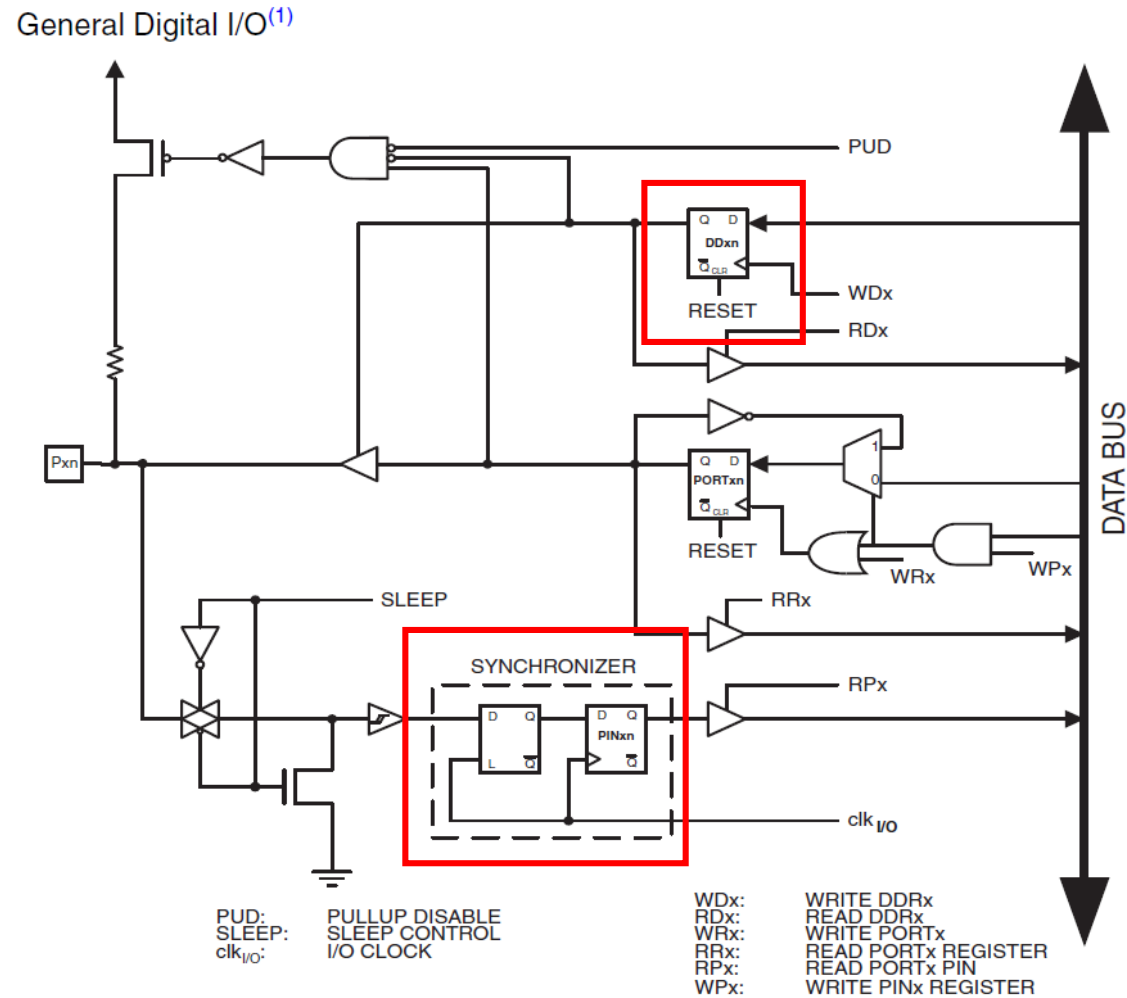
# GPIO logikai felépítés



# GPIO logikai felépítés

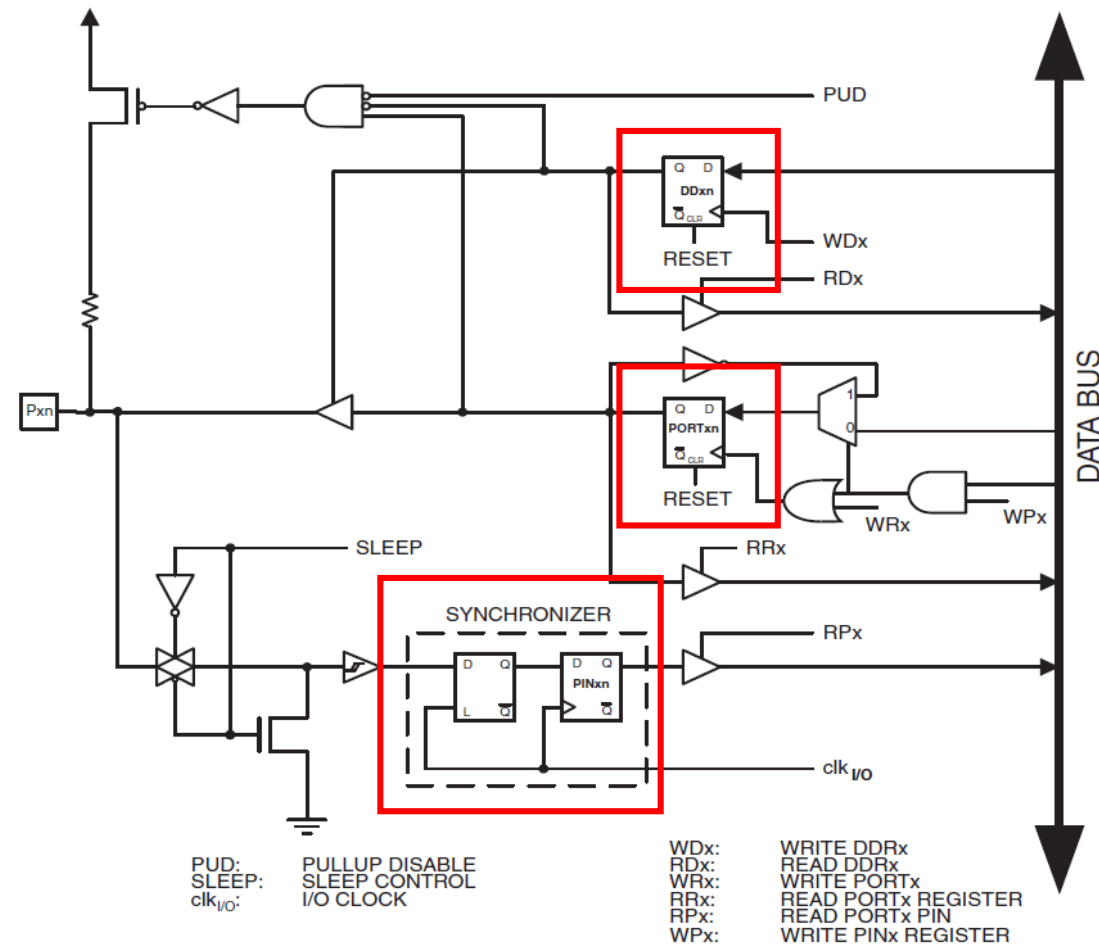


# GPIO logikai felépítés



# GPIO logikai felépítés

General Digital I/O<sup>(1)</sup>



# DDRx regiszter

- DDRx (Data Direction Register) :
  - 8 bites regiszter, 8db PORT tartozik hozzá
  - Az „x” ebben az esetben A, B, C, ..., L érték lehet.
  - A DDRx regiszterben lehet beállítani a PORT irányát (ki vagy pedig bement legyen).
  - Alap állapotban minden port bemenet.

# PINx regiszter

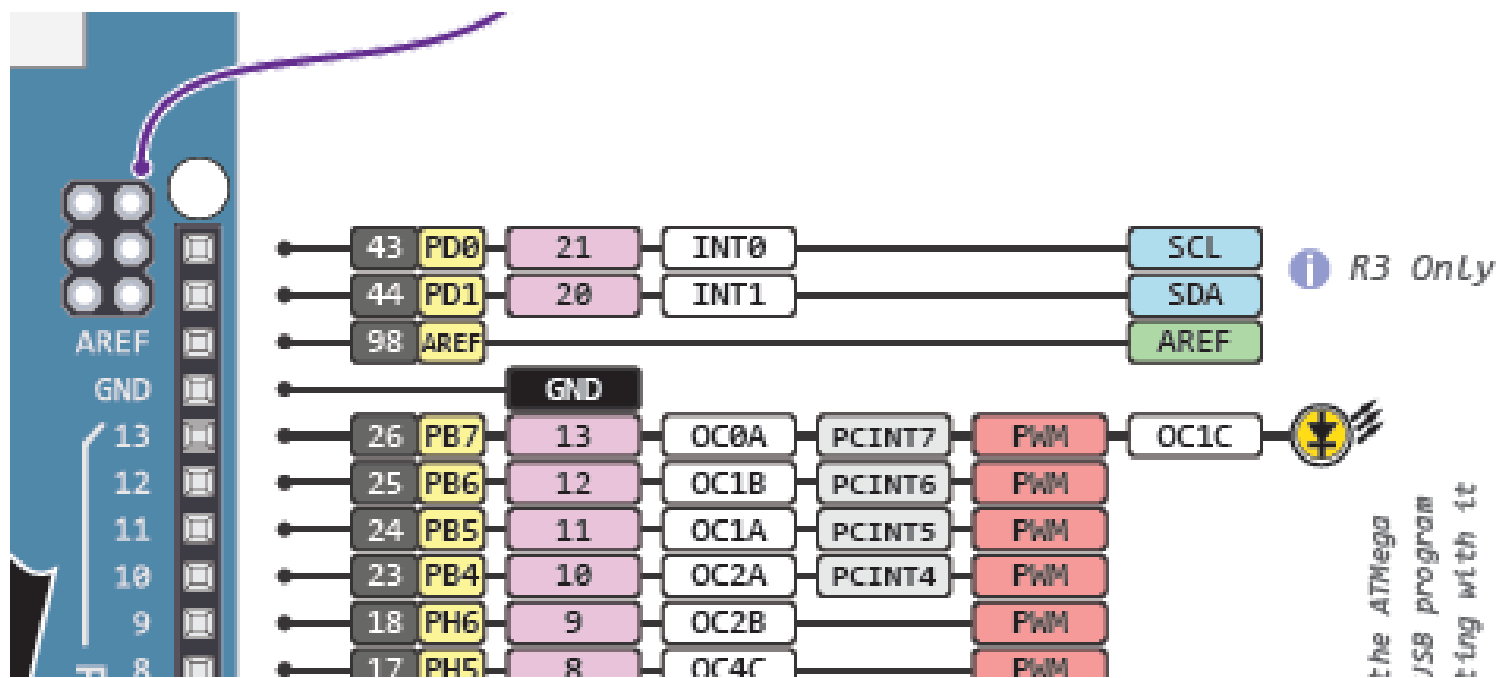
- PINx (Port Input Register):
  - 8 bites regiszter
  - 8 db port tartozik hozzá
  - A PINx regiszterből lehet kiolvasni az adott port logikai állapotát (pl. bemenet esetén LOW vagy HIGH értéken van).



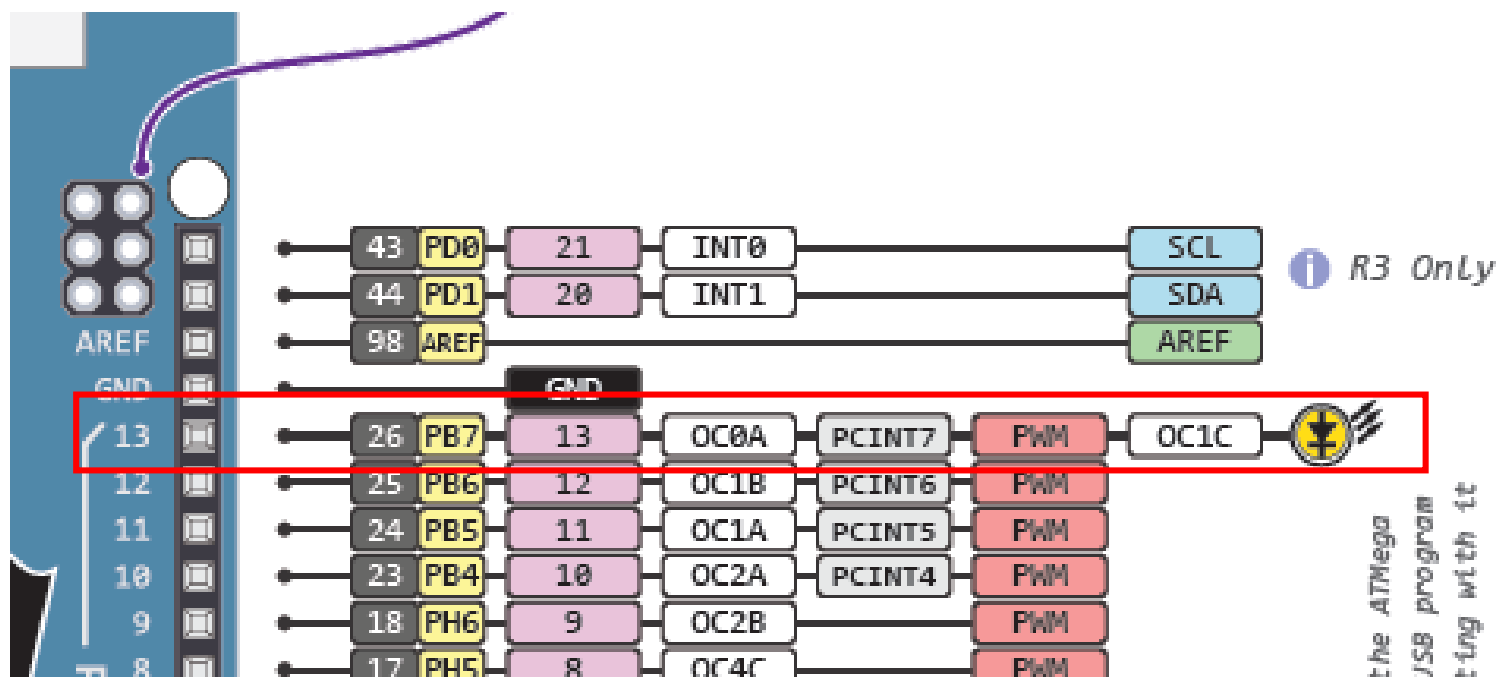
# PORTx regiszter

- -PORTx (Port Output Register):
  - 8 bites regiszter
  - 8 db port tartozik hozzá
  - A PORTx regiszter segítségével lehet beállítani az adott PORT logikai állapotát (LOW vagy HIGH).

# GPIO kezelés – LED villogtatás



# GPIO kezelés – LED villogtatás



# DDRB, PORTB Regiszterek

DDRB7	DDRB6	DDRB5	DDRB4	DDRB3	DDRB2	DDRB1	DDRB0
0	0	0	0	0	0	0	0

```
DDRB |= (1 << PB7); //0b10000000
```

DDRB7	DDRB6	DDRB5	DDRB4	DDRB3	DDRB2	DDRB1	DDRB0
1	0	0	0	0	0	0	0

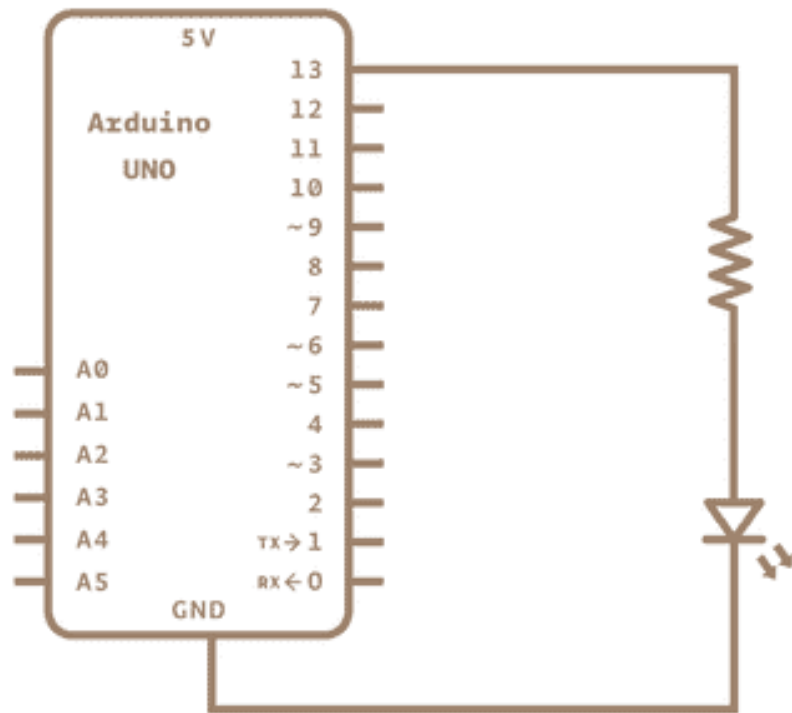
```
PORTB &= ~(1 << PB7); //0b01111111
```

PORTB7	PORTB6	PORTB5	PORTB4	PORTB3	PORTB2	PORTB1	PORTB0
0	0	0	0	0	0	0	0

```
PORTB |= (1 << PB7); //0b10000000
```

PORTB7	PORTB6	PORTB5	PORTB4	PORTB3	PORTB2	PORTB1	PORTB0
1	0	0	0	0	0	0	0

# Kapcsolás összeállítása



$$R = \frac{V_{in} - 0.7V}{I} = \frac{5V - 0.7V}{10 \text{ mA}} = 430 \, \Omega$$

# Alaplapi LED villogtatása

```
1 void setup() {  
2     //PB7 láb beállítása kimenetnek  
3     DDRB |= (1 << PB7);  
4 }  
5  
6 void loop() {  
7     // PORTB7 bit értékének beállítása 1-re  
8     PORTB &= ~(1 << PB7); // ~(1 << PB7) = 0b01111111  
9     delay(1000);  
10  
11     // PORTB7 bit értékének beállítása 0-ra  
12     PORTB |= (1 << PB7); // (1 << PB7) = 0b10000000  
13     delay(1000);  
14 }
```

# Alaplapi LED villogtatása

```
1 void setup() {
2     //PB7 láb beállítása kimenetnek
3     DDRB |= (1 << PB7);
4 }
5
6 void loop() {
7     // PORTB7 bit értékének beállítása 1-re
8     PORTB &= ~(1 << PB7); // ~(1 << PB7) = 0b01111111
9     delay(1000);
10
11     // PORTB7 bit értékének beállítása 0-ra
12     PORTB |= (1 << PB7); // (1 << PB7) = 0b10000000
13     delay(1000);
14 }
```

# Alaplapi LED villogtatása

```
1 void setup() {  
2     //PB7 láb beállítása kimenetnek  
3     DDRB |= (1 << PB7);  
4 }  
5  
6 void loop() {  
7     // PORTB7 bit billegtetése XOR művelettel  
8     PORTB ^= (1 << PB7); // (1 << PB7) = 0b10000000  
9     delay(1000);  
10 }
```





ÓBUDAI EGYETEM

NEUMANN JÁNOS INFORMATIKAI KAR

# Köszönöm a figyelmet!