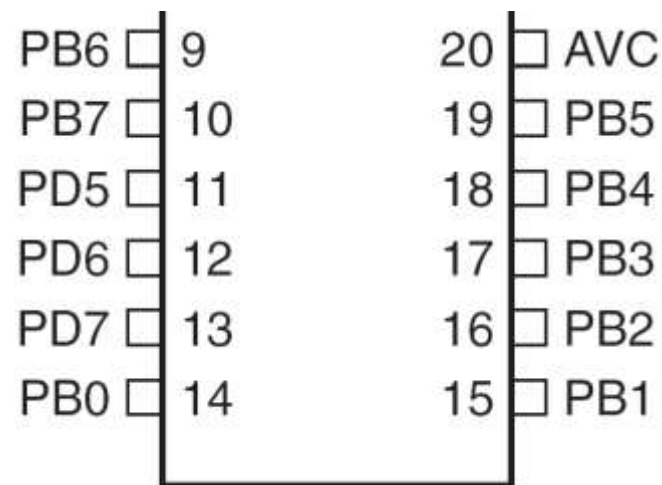


Mikroprocesori u industriji

Izlazni portovi

Microcontrolerski portovi i pinovi

- Priključci kroz koje mikrokontroler opšti sa spoljašnjom sredinom
 - Pr. PORTB
 - Pinovi PB0 – PB7
 - Ne moraju biti susjedni
 - Često bi-direcioni



Microkontrolerski portovi i pinovi

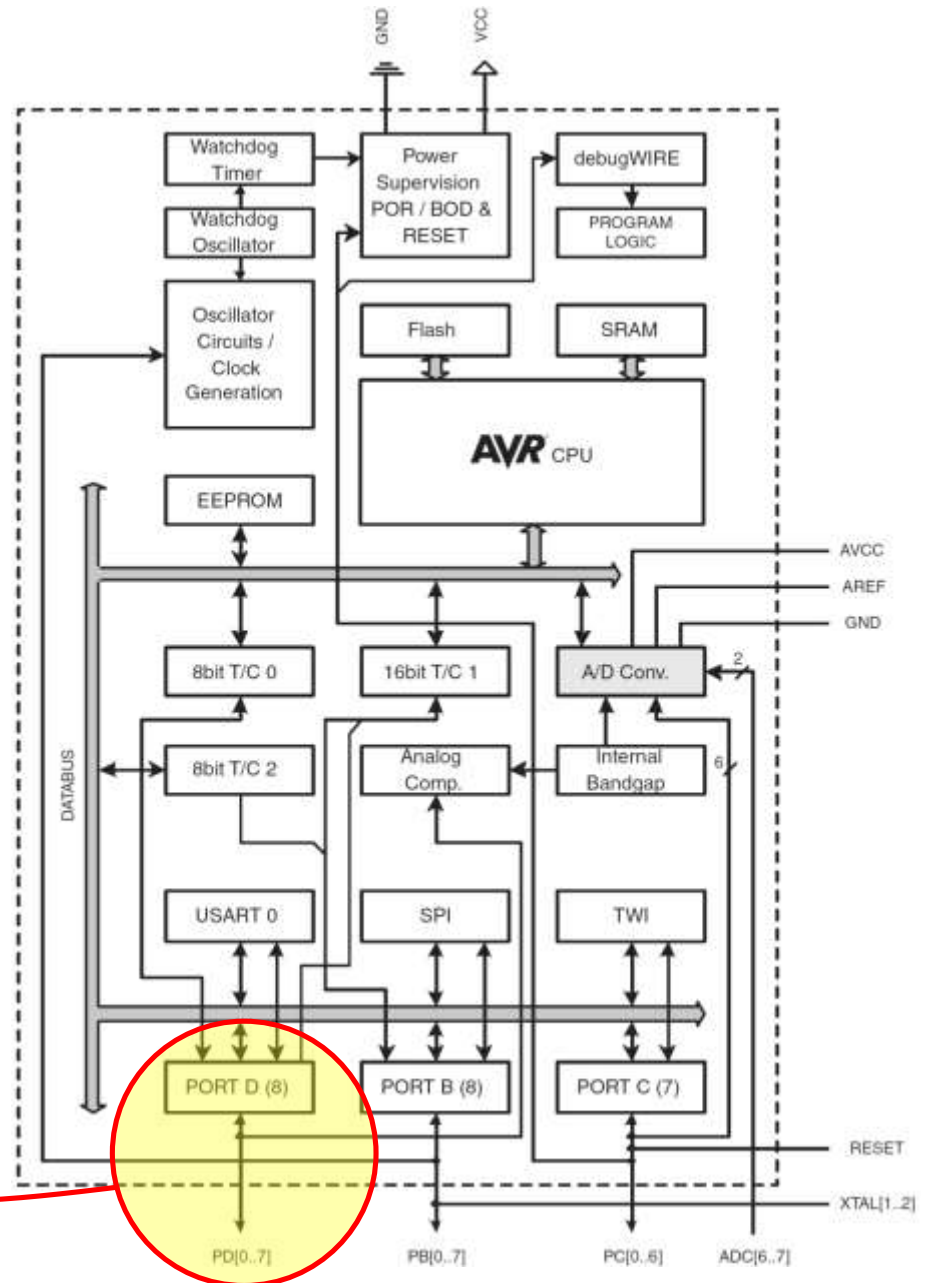
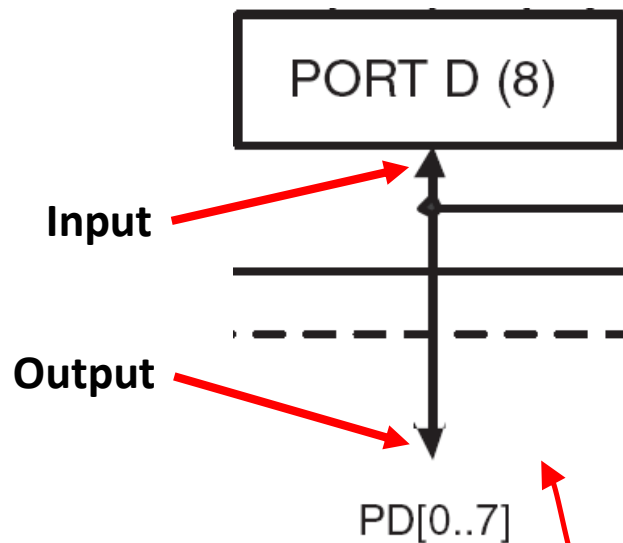
(PCINT14/RESET) PC6	1	28	PC5 (ADC5/SCL/PCINT13)
(PCINT16/RXD) PD0	2	27	PC4 (ADC4/SDA/PCINT12)
(PCINT17/TXD) PD1	3	26	PC3 (ADC3/PCINT11)
(PCINT18/INT0) PD2	4	25	PC2 (ADC2/PCINT10)
(PCINT19/OC2B/INT1) PD3	5	24	PC1 (ADC1/PCINT9)
(PCINT20/XCK/T0) PD4	6	23	PC0 (ADC0/PCINT8)
VCC	7	22	GND
GND	8		

Port Pin – Usmjerenje podataka

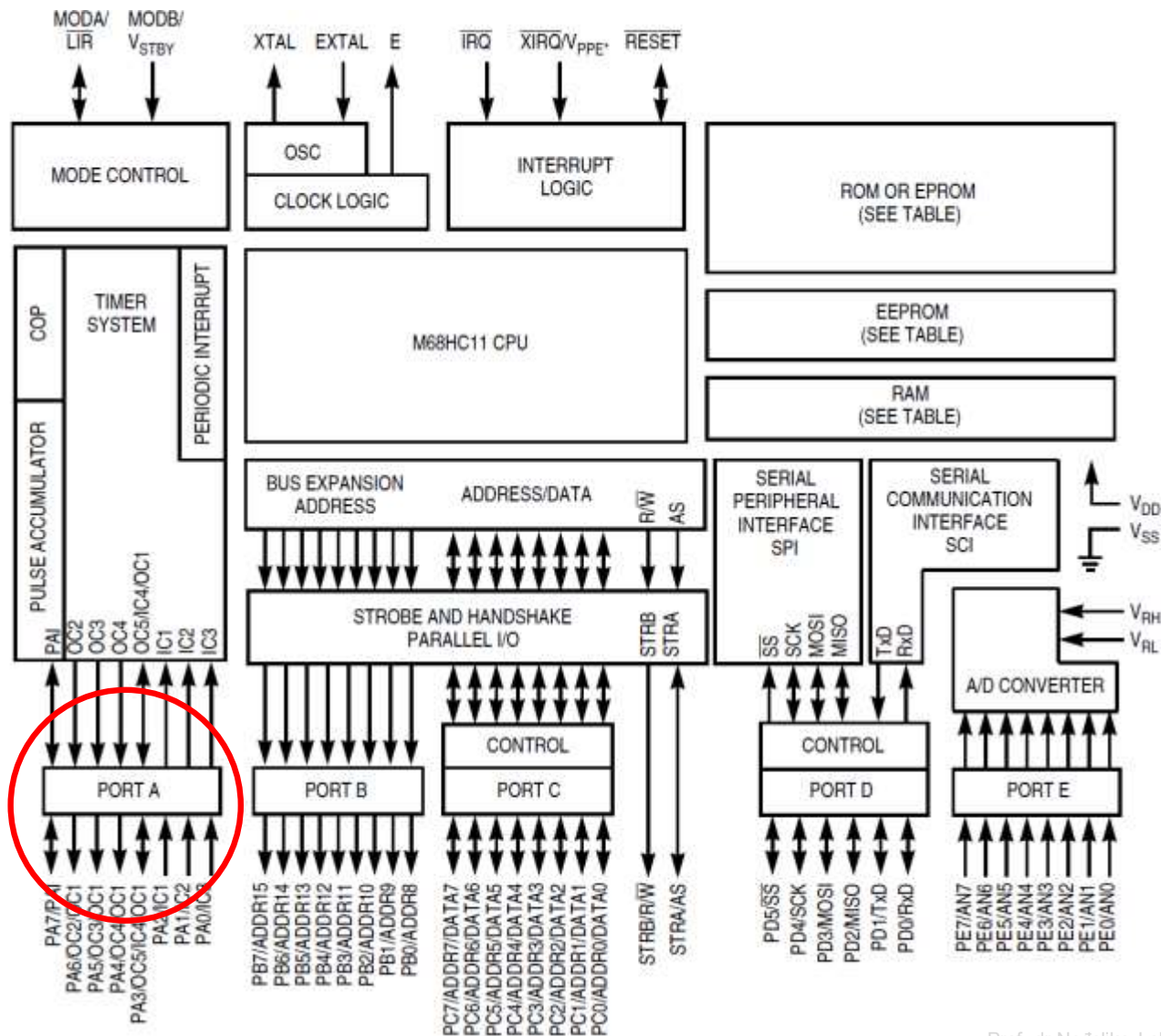
- Ulaz
 - Kada se želi uzeti informacija iz spoljašnjeg svijeta (senzori) u MCU
- Output
 - Kada se želi izmijeniti stanje nečega **izvan** MCU (uključiti ili isključiti motor, itd.) (aktuatori)
- Po uključenju napajanja svi pinovi su ulazni.
- Program može mijenjati usmjerenja podataka za svaki pin u svakom trenutku.

ATmega328

Blok diagram

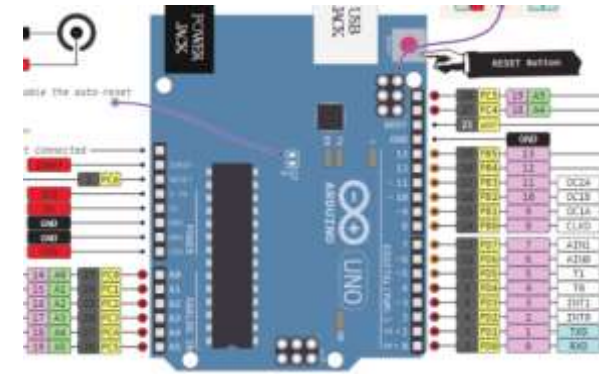


M68HC11 mikrokontroler



Postavljenje smjera toka podatka za pin

- Arduino
 - `pinMode(pin_no., dir)`
 - Pr. postaviti Arduino pin 3 (PD3) kao izlazni
 - `pinMode(3, OUTPUT);`
 - Napomena: jedan pin u jednom trenutku
 - Predpostavimo da se želi postaviti pinove 3, 5, i 7 (PD3, PD5, i PD7) kao izlazne?
 - Postoji li način da se oni postave istovremeno?
 - Da! Kako, slijedi kasnije...

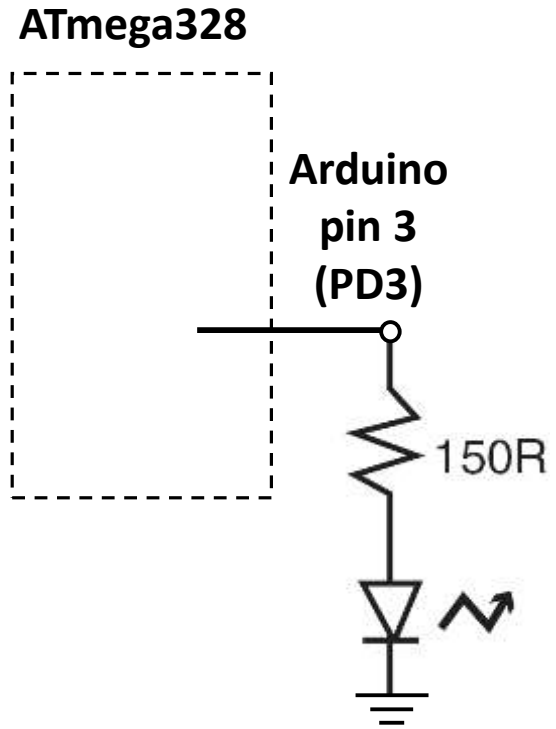


Napon na pinu

- Mikrokontroleri su u osnovi ***digitalni*** uređaji.
Za digitalne ulazno/izlazne (IO) pinove:
 - Informacija je ‘kodirana’ u dva diskretna stanja:
 - HIGH or LOW (logic: 1 or 0)
 - Naponi
 - TTL
 - » 5 V (za HIGH)
 - » 0 V (za LOW)
 - 3.3 V CMOS
 - » 3.3 V (za HIGH)
 - » 0 V (za LOW)

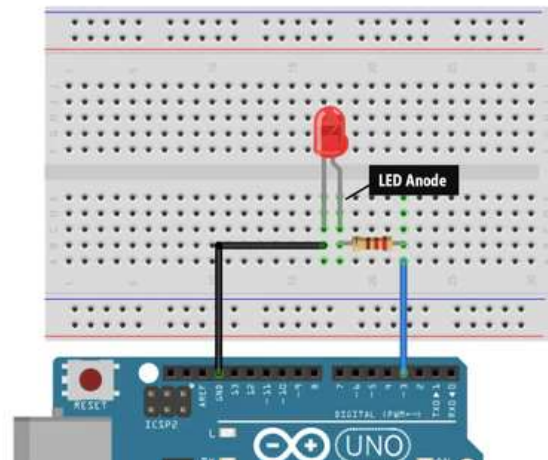
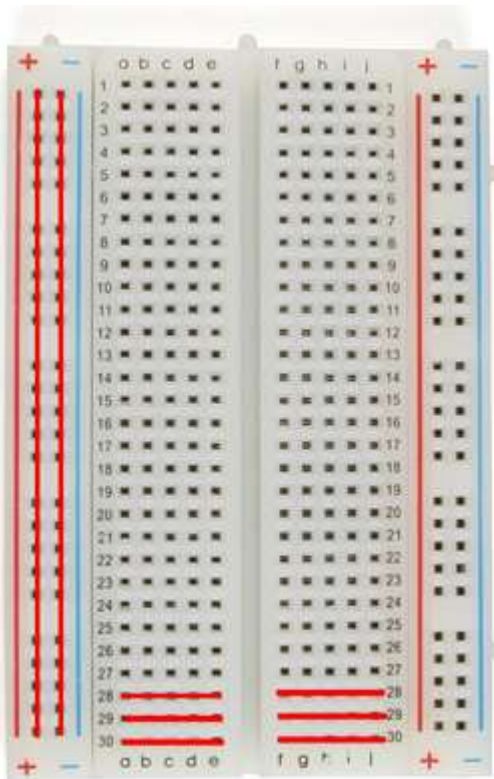
Pin upotrijebljen kao izlazni

- Uključiti LED, koja je povezana na Arduino pin 3 (PD3) (otpornik!)
 - Koji tok podataka treba biti za pin 3 (PD3)?
 - **pinMode(____, ____);**
 - Uključenje LED
 - **digitalWrite(3,HIGH);**
 - Isključenje LED
 - **digitalWrite(3,LOW);**



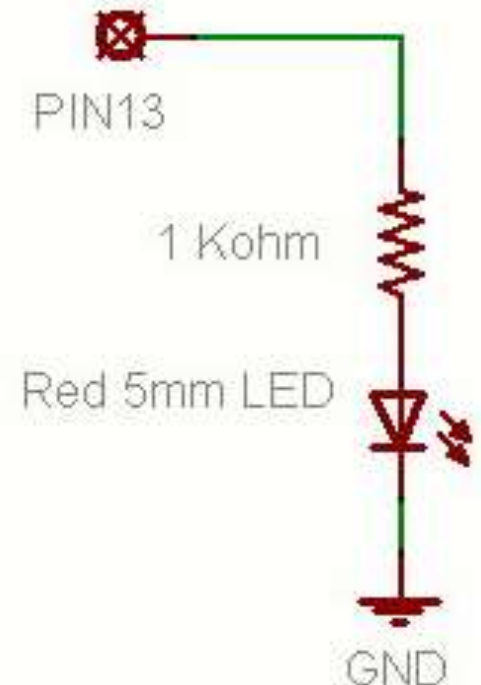
Pin upotrijebljen kao izlazni

- Uključenje LED
 - **digitalWrite(3,HIGH);**
- Isključenje LED
 - **digitalWrite(3,LOW);**



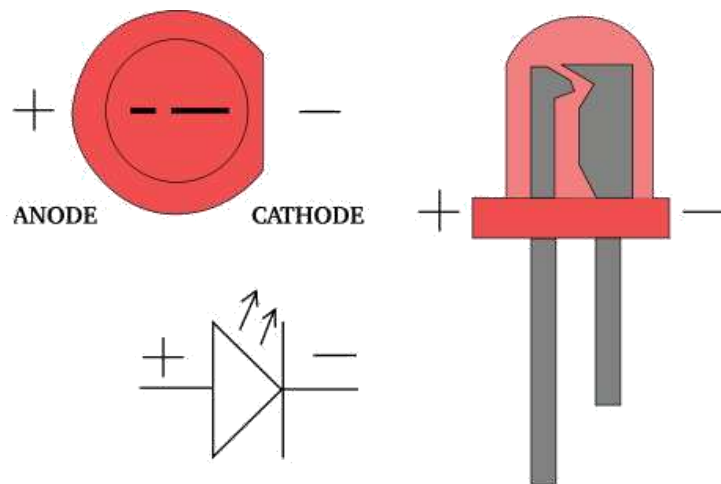
Osnovno LED kolo

- Povežite pin 13 mikrokontrolera na jedan kraj otpornika.
- Drugu nožicu otpornika spojite na dužu nožicu LED.
 - Veća otpornost znači slabije svjetlo.
 - Manja otpornost znači jače svjetlo.
 - Bez otpornosti znači pregorijevanje LED ili preopterećenje porta.
- Kraću nožicu LED spojite na negativni priključak napajanja (masu).

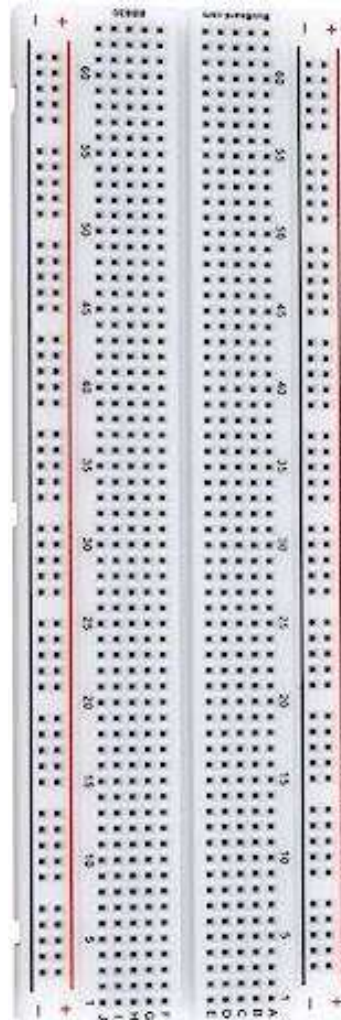
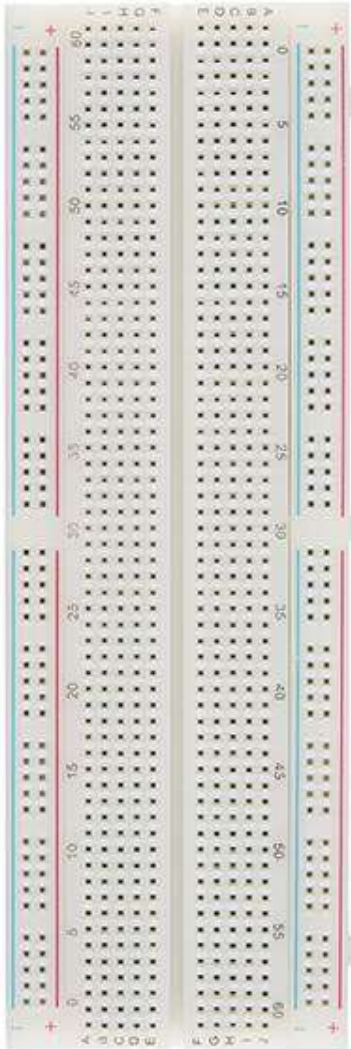


Blink Skeč (Treperenje)

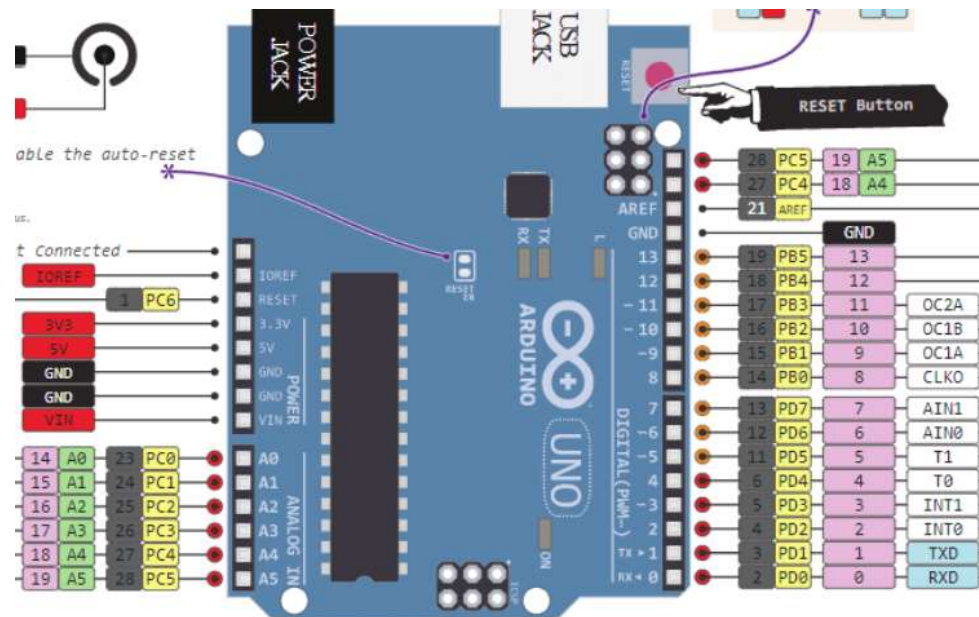
- **File > Examples > Digital > Blink**
- LED ima polaritet
 - Negativni je indikovani zasječenim obodom tijela diode i kraćom nožicom.



Neke varijante experimentalnih ploča



Pitanje od prije?



- Pitanje od prije:
 - Postoji li način da se tok podataka postavi za više pinova istovremeno?
- Sav rad na MCU de

Data Direction Register

- Ako je bit nula -> pin će biti ulazni
 - Postavljenje bita na nulu == ‘**čišćenje bita**’ (‘clearing the bit’)
- Ako je bit jedan -> pin će biti izlazni
 - Postavljenje bita na jedinicu == ‘**postavljanje bita**’ (‘setting the bit’)
- Za istovremenu promjenu toka podataka za više pinova koji pripadaju portu PORTx:
 1. Određivanje koje bitove treba postaviti a koje očistiti u registru DDRx.
 2. Upisati binarni (hex) broj u DDRx.

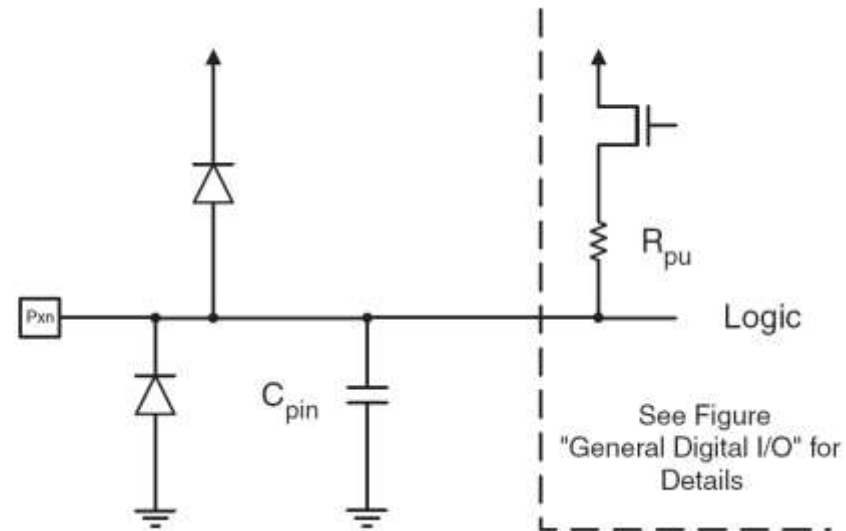
ATmega328 registri za rad sa portovima

- Vidijeti ATmega328 data sheet, pp. 76-94
- Za digitalne IO, važni registri su:
 - DDRx
 - Data Direction bit u DDRx registru (read/write)
 - PORTx
 - PORTx data registar (read/write)
 - PINx
 - PINx registar (read only)

PORT Pin i registar - detalji

ATmega328 datasheet, pp. 76-94

Figure 13-1. I/O Pin Equivalent Schematic



Primjer 1

- Postaviti Arduino pinove 3, 5, i 7 (PD3, PD5, i PD7) kao izlazne

- Arduino pristup

```
pinMode(3, OUTPUT);  
pinMode(5, OUTPUT);  
pinMode(7, OUTPUT);
```

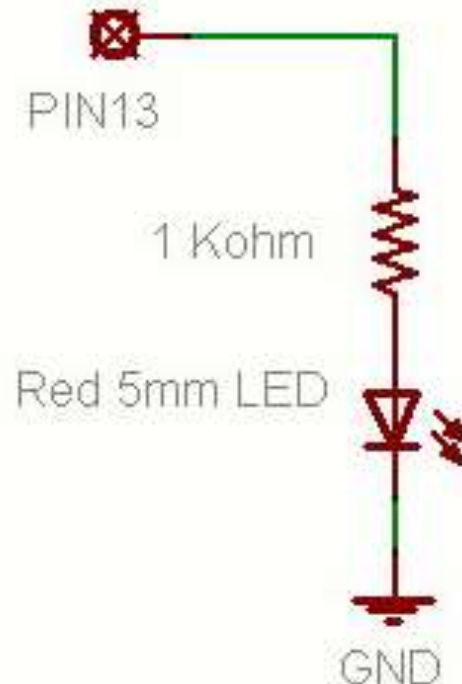
Ili ako je upotrijebljena me106.h:

```
pinMode(PIN_D3, OUTPUT);  
pinMode(PIN_D5, OUTPUT);  
pinMode(PIN_D7, OUTPUT);
```

- Alternativni pristup

Prosto elektronsko kolo

- Najjednostavniji sklop.
- Uključi/isključi svjetlo.
- Struja teče iz pina (izvora napajanja), kroz potrošač (LED).



Struktura Arduino programa

- Arduino program == 'sketch'
 - Mora imati:
 - `setup()`
 - `loop()`
 - `setup()`
 - Konfiguriše pinove i registre
 - `loop()`
 - Pokreće glavno tijelo programa neprestano
 - Kao `while(1) {...}`
 - Gdje je `main()` ?
 - Arduino uprošćava stvari
 - Odrađuje za Vas

```
/* Blink - turns on an LED for DELAY_ON msec,
then off for DELAY_OFF msec, and repeats
BJ Furman rev. 1.1   Last rev: 22JAN2011
*/
#define LED_PIN 13    // LED on digital pin 13
#define DEL
```

Funkcije: millis (micros)

Izbjegavanje upotrebe dužeg čekanja u skeču!

Vraća broj milisekundi (mikrosekundi) koji je prošao od kada je Arduino ploča počela da izvršava određeni program.

```
unsigned long myTime;
```

```
myTime = millis();
```

Funkcije: millis (micros)

Primjer upotrebe funkcije millis():

```
unsigned long startMillis; //globalne promjenljive
unsigned long currentMillis;
const unsigned long period = 1000; //vrijednost je u milisekundama
const byte ledPin = 13; //korištenje ugrađene diode LED

void setup()
{
    pinMode(ledPin, OUTPUT);
    startMillis = millis(); //inicijalno početno vrijeme
}

void loop()
{
    currentMillis = millis(); //dodjela trenutnog „vremena“ promjenljivoj
    if (currentMillis - startMillis >= period) //ispitivanje da li je period istekao
    {
        digitalWrite(ledPin, !digitalRead(ledPin)); //ako jeste, promjena stanja LED.
        startMillis = currentMillis; //VAŽNO je ubilježiti startno vrijeme tren. stanja LED.
    }
}
```

Funkcije: millis (micros)

Izbjegavanje problema preliivanja (overflow) kada se koristi millis() i micros()

Funkcije millis() i micros() prelijevaju nakon otprilike 50 dana i 70 minuti, respektivno. Potencijalni problem može se lako izbjeći malim prilagođenjem koda.

Umjesto ovako:

```
int period = 1000;
unsigned long time_now = 0;

void setup() {
    Serial.begin(115200);
}

void loop() {
    if(millis() > time_now + period){
        time_now = millis();
        Serial.println("Hello");
    }

   
```

Funkcije: millis (micros)

Može izgledati da predhodno nema puno smisla. Može izgledati da se problem samo pomjera, a da se ne rješava.

Gledajući matematički, nema puno smisla, jer će lijeva strana postati negativna kada dođe do prelijevanje millis() funkcije (rezultat oduzimanja od veoma malog cijelog broja veoma velikog cij

TRČEĆE SVJETLO - 4 LED

SEDMO-SEGMENTNI DISPLAY

Napisati program koji broji od 0 do 9 s pon

Rješenje:

```
#define PERIOD 1000
```

```
int i = 0;
```

```
long start;
```

ČETVOROCIFARSKI SEDMO-SEGMENTNI DISPLAY

Rješenje:

```
#define PERIOD 2
```

```
int i = 0;
```

```
long
```

ZADACI ZA VJEŽBU 1

1. „Trčeća tama“ upotrebom 4 LED. Uv

4. Napisati program koji na četvorocifarskom sedmosegmentnom LED displeju ciklič

