

2. Vaja: Serijski prenos podatkov

Napiši funkcijo, ki naredi serijski prenos 8-bitnega naslova in 8-bitnega podatka preko digitalnih priključkov PD0 in PD1. Delali bomo v načinu direktnega dostopa do priključkov mikrokrmilnika preko registrov DDRD in PORTD. Delovanje posameznih bitov prikazuje tabela:

DDRDn	PORTn	priključek	stanje
0	0	vhod	visoka impedanca
0	1	vhod	pullup
1	0	izhod	logična 0
1	1	izhod	logična 1

Ura SCK je vezana na priključek PD0, podatkovni signal SDA pa na PD1. Na izhodih bomo logično 1 nastavljeni v načinu **pullup**, logično 0 pa kot običajen izhodni signal.

V funkciji setup() nastavimo začetno stanje (pullup na obeh signalih):

```
void setup() {
  DDRD  &= B11111100; // PD1 in PD0 sta vhoda
  PORTD |= B00000011; // z vrednostjo pullup
}
```

Napiši funkcijo, ki sprejme dve 8-bitni vrednosti (adr, dat) in ju pošlje v serijski obliki:

```
void prenos(byte adr, byte dat)
```

Prenos poteka po naslednjem vrstnem redu:

- start: SDA=0, SCK=1, delay()
- 8x prenos naslova v zanki i=7...0: SDI=adr(i), SCK=0, delay(), SCK=1, delay()
- potrditev: SDI=1, SCK=0, delay(), SCK=1, delay()
- 8x prenos podatkov v zanki i=7...0: SDI=dat(i), SCK=0, delay(), SCK=1, delay()
- potrditev: SDI=1, SCK=0, delay(), SCK=1, delay()
- stop: SDA=0, SCK=0, delay(), SDA=0, SCK=1, delay(), SDA=1, SCK=1, delay()

Med preklopi posameznih signalov uporabi funkcijo za zakasnitev: delayMicroseconds(10);