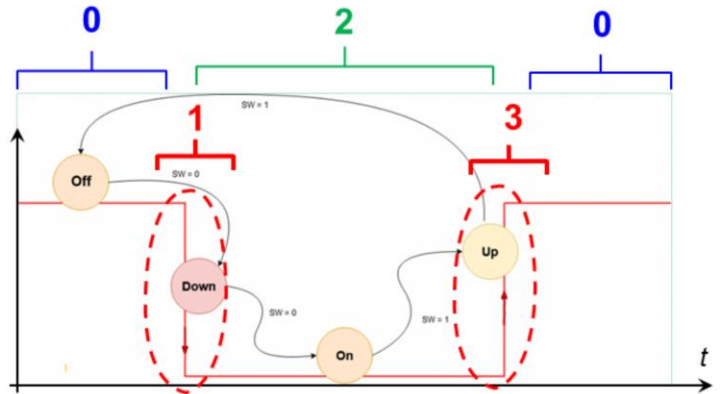
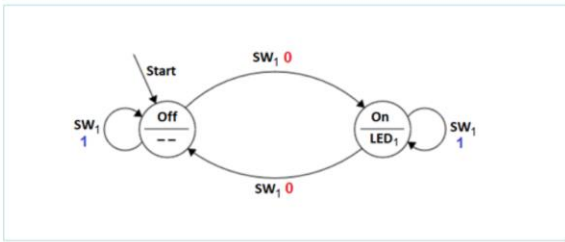


Analýza cv.03.

Cieľ: Naprogramovať zapínanie a vypínanie led jedným tlačítkom.



Stav ledky sa zmení pri zatlačení (1), niekde v strede (2) alebo pri pustení tlačítka (3)? Ak sa zhodneme, že zmena stavu sa udeje v (1), resp. (3) potom je namieste kde a akým spôsobom treba pozmeniť vzorový program aby sa prepínanie stavu led zmenilo: (1)->(3), resp. (3)->(1).

Odozvané programy použili funkciu `_delay_ms(10)`:

- Nula krát,
- Dva krát, resp.
- Štyri krát.

Namieste je otázka: koľko trvá stav (1), resp. (3) pri zatlačení, resp. pustení tlačítka?

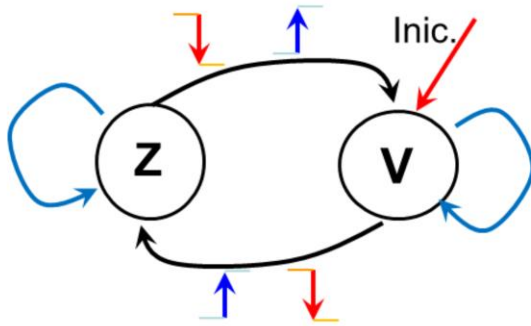
Odpoveď:

1. Niečo menej ako 2us.
2. O niečo viac ako 10ms.

Skúste zdôvodniť.

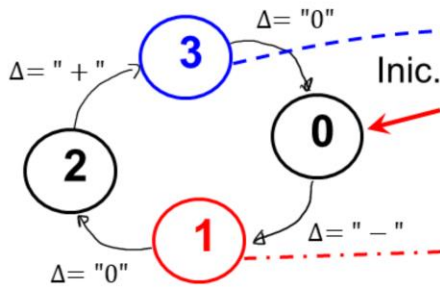
Na prednáške sme na ošetrenie zákmitov použili „diferenciu“. Môžeme povedať, že príklad z prednášky je stavový automat a naopak príklad z cvičenia vyhodnocuje diferenciu, teda smer? Down, resp. Up?

LEDKA: `if ("TRUE") toggle_bit(PORTB,LED1);`



```
#define F_CPU 16000000UL
#include <avr/io.h>
#include <util/delay.h>
#define set_bit(ADDRESS,BIT) (ADDRESS |= (1<<BIT))
#define toggle_bit(ADDRESS,BIT) (ADDRESS ^= (1<<BIT))
#define LED1 PB5 // zabudovana dioda
#define SW1 PD6 // tlacitko
enum states { Low, High };
int main(void){
enum states last_st = High;
enum states new_st = High;
set_bit(DDRB,LED1); // set pin LED1 as output
set_bit(PORTD,SW1); // pull-up resistor ON
while(1){
delay_ms(10); // chvilu pockajme
{ last_st = new_st;
if (bit_is_clear(PIND,SW1)) new_st = Low; else new_st = High;
// if (new_st && !last_st) toggle_bit(PORTB,LED1);
if (!new_st && last_st) toggle_bit(PORTB,LED1);
} /* end of while loop */
return(0); // sem nikdy neprideme
}
```

TLAČÍTKO: $\Delta = y(t_{n+1}) - y(t_n)$



Zdefinujeme dva stavy Ledky:

Z – zapnutá

V – vypnutá.

A prechod medzi týmito stavmi realizujeme NÁBEŽNOU, resp. DOBEŽNOU hranou.