**Lab assignment**

1. <https://github.com/mikolasstradej/Digital-Electronics-2>
2. Logické operátory :

* | - logický součet (OR)
* & - logický součin (AND)
* ^ - exkluzivní součet (EX-OR)
* ~ - jednotkový doplněk (negace)
* << - bitový posun doleva

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| A | B | | | & | ^ |
| 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 1 | 0 | 1 |
| 1 | 0 | 1 | 0 | 1 |
| 1 | 1 | 1 | 1 | 0 |

|  |  |
| --- | --- |
| A | ~ |
| 0 | 1 |
| 1 | 0 |

Bitový posun doleva – posune všechny bity o x pozic doleva

A = 11100110

A << 2 = 10011000

1. C code

|  |
| --- |
|  |
|  | #define LED\_GREEN PB5 |
|  | #define SHORT\_DELAY 350 |
|  | #ifndef F\_CPU |
|  | #define F\_CPU 16000000 |
|  | #endif |
|  |  |
|  | #include <util/delay.h> |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  | int main(void) |
|  | { |
|  |
|  |
|  | DDRB = DDRB | (1<<LED\_GREEN); |
|  |
|  |
|  | PORTB = PORTB & ~(1<<PB5); |
|  | while (1) |
|  | { |
|  |
|  | \_delay\_ms(SHORT\_DELAY); |
|  |  |
|  |
|  |
|  | PORTB = PORTB ^ (1<<LED\_GREEN); // (^ je fce XOR) |
|  | \_delay\_ms(SHORT\_DELAY\*3); |
|  | PORTB = PORTB ^ (1<<LED\_GREEN); |
|  | \_delay\_ms(SHORT\_DELAY); |
|  | PORTB = PORTB ^ (1<<LED\_GREEN); |
|  | \_delay\_ms(SHORT\_DELAY); |
|  | PORTB = PORTB ^ (1<<LED\_GREEN); |
|  | \_delay\_ms(SHORT\_DELAY); |
|  | PORTB = PORTB ^ (1<<LED\_GREEN); |
|  | \_delay\_ms(SHORT\_DELAY); |
|  | PORTB = PORTB ^ (1<<LED\_GREEN); |
|  | \_delay\_ms(SHORT\_DELAY\*5); // D |
|  |  |
|  | PORTB = PORTB ^ (1<<LED\_GREEN); |
|  | \_delay\_ms(SHORT\_DELAY); |
|  | PORTB = PORTB ^ (1<<LED\_GREEN); |
|  | \_delay\_ms(SHORT\_DELAY\*5); // E |
|  |  |
|  | PORTB = PORTB ^ (1<<LED\_GREEN); |
|  | \_delay\_ms(SHORT\_DELAY); |
|  | PORTB = PORTB ^ (1<<LED\_GREEN); |
|  | \_delay\_ms(SHORT\_DELAY); |
|  | PORTB = PORTB ^ (1<<LED\_GREEN); |
|  | \_delay\_ms(SHORT\_DELAY); |
|  | PORTB = PORTB ^ (1<<LED\_GREEN); |
|  | \_delay\_ms(SHORT\_DELAY); |
|  | PORTB = PORTB ^ (1<<LED\_GREEN); |
|  | \_delay\_ms(SHORT\_DELAY\*3); |
|  | PORTB = PORTB ^ (1<<LED\_GREEN); |
|  | \_delay\_ms(SHORT\_DELAY); |
|  | PORTB = PORTB ^ (1<<LED\_GREEN); |
|  | \_delay\_ms(SHORT\_DELAY\*3); |
|  | PORTB = PORTB ^ (1<<LED\_GREEN); |
|  | \_delay\_ms(SHORT\_DELAY); |
|  | PORTB = PORTB ^ (1<<LED\_GREEN); |
|  | \_delay\_ms(SHORT\_DELAY\*3); |
|  | PORTB = PORTB ^ (1<<LED\_GREEN); |
|  | \_delay\_ms(SHORT\_DELAY\*5); |
|  | } |
|  |  |
|  | return 0; |
|  | } |
|  |  |