



WYDZIAŁ ELEKTROTECHNIKI, AUTOMATYKI I INFORMATYKI

Technika mikroprocesorowa - laboratorium

Sprawozdanie 2

Wacławczyk Michał

Kaczmarczyk Marek

Woźnica Dariusz

kierunek: Automatyka i Robotyka

studia stacjonarne I stopnia

Opole, 2020

1.Opis programu

Edycja naszego pierwszego projektu

Program wykonuje się w pętli, domyślnie miga dioda czerwona, po naciśnięciu dioda czerwona gaśnie i zaczyna migać dioda zielona. Po kolejnym naciśnięciu zaczyna migać dioda czerwona.

Dodaliśmy przerwanie które zmienia flagę odpowiadającą za kolor diody która miga. Gdy funkcja obsłuży przerwanie, następuje powrót do programu i wznowienie jego działania od miejsca w którym został przerwany.

2. Kod

```
#include <msp430.h>

#define SW      BIT3           // przycisk -> P1.3
#define RED     BIT0           // czerwony led -> P1.0
#define GREEN   BIT6           // zielony led -> P1.6

void main(void)
{
    WDTCTL = WDTPW | WDTHOLD;           // Stop watchdog timer

    P1DIR |= RED+GREEN;                 // Set LED pins -> Output
    P1DIR &= ~SW;                        // Set SW pin -> Input
    P1REN |= SW;                        // Włączenie rezystora dla
wejścia tact switcha
    P1OUT |= SW;                        // Pull up dla pinu switch (
switch przełącza do 0V)

    volatile unsigned int flag = 0; //zmienna do stanu
        volatile unsigned long i = 0; //zmienna do timera
        // przerwanie

    P1IES &= ~SW;                       // Wybór przerwania na
zbocze narastające sygnału
```

```

P1IE |= SW; // Włączenie
przerwania na przycisk (tactswicht)

__bis_SR_register(LPM4_bits + GIE); // LPM4 oraz dodanie flagi
przerwania procesora

while(1)
{
    if(flag)
    {
        P1OUT |= BIT0; //czerwony LED -> ON
        for(i = 0; i<10000; i++); //opoznienie
        P1OUT &= ~BIT0; //czerwony LED -> OFF
        for(i = 0; i<10000; i++); //opoznienie
    } else {
        P1OUT |= BIT6; //zielony LED -> ON
        for(i = 0; i<10000; i++); //opoznienie
        P1OUT &= ~BIT6; //zielony LED -> OFF
        for(i = 0; i<10000; i++); //opoznienie
    }
}

// część programu wywoływana naciśnięciem przycisku
#pragma vector=PORT1_VECTOR
__interrupt void Port_1(void)
{
    __delay_cycles(20000); //
    oczekanie 20ms ( drgania styku)

    flag = !flag; //
    negacja zmiennej odpowiadającej za kolor migającej diody

```

```
                P1IFG &= ~SW;  
// kasowanie flagi przerwania  
}
```