



WYDZIAŁ ELEKTROTECHNIKI, AUTOMATYKI I INFORMATYKI
INSTYTUT AUTOMATYKI
KIERUNEK AUTOMATYKA I ROBOTYKA
STUDIA NIESTACJONARNE I STOPNIA

LABORATORIUM - GRUPA L1

Technika Mikroprocesorowa

ĆWICZENIE 2.

Program realizujący sekwencje przerwania sygnału wyjść mikrokontrolera

Wykonali:

Adam Bunzel
Olaf Karch

Prowadzący:

MGR INŻ. Andreas Kowol

TERMIN ODDANIA: : 18.04.2020

1. Opis działania programu

Program ma za zadanie sterować dwiema diodami LED na wyjściu przy pomocy przycisku pod adresem P1.3. Po uruchomieniu mikrokontrolera powinna zapalić się dioda zielona migająca z częstotliwością 0.05s a dioda czerwona pozostać wyłączona. Po wciśnięciu przycisku częstotliwość diody zielonej powinna zmienić się na 0.1s a dioda czerwona osiągnąć stan wysoki. Ponowne wciśnięcie wraca do stanu początkowego.

2. Kod programu.

```
#include <msp430.h>
/* ----- Adresowanie symboliczne -----
Przypisanie adresów symbolicznych do bitów
-----*/

#define przycisk      BIT3           // Przycisk -> P1.3
#define GREEN        BIT6           // LedRed LED -> P1.0
#define RED           BIT0           // LedGreen LED -> P1.6
#define Time1         100000
#define Time2         50000

volatile int flag=0;

void main(void) {
    WDTCTL = WDTPW | WDTHOLD;         // Stop watchdog timer
    /* ----- Hardware -----
    Ustawienie wejść wyjść rezystorów itd
    -----*/

    P1DIR |= GREEN + RED;             // Ustawienie ledów jako wyjścia
    P1DIR &= ~przycisk;               // ustawienie przycisku jako wejście
    P1REN |= przycisk;                // włączenie rezystora
    P1OUT |= przycisk;                // ustawienie rezystora jako pull up
    P1IES &= ~przycisk;               //Włączana jest obsługa przerwań dla
    wejścia

    __bis_SR_register( GIE);          // ustawia bity rejestru stanu
```

```

while(1) // pętla while
{
    P1OUT ^= RED;
    if(flag%2) delay_cycles(Time1); // Warunek
warunkujacy swiecenie diody czerwonej
    else delay_cycles(Time2);
};
}

#pragma vector=PORT1_VECTOR
__interrupt void Port_1(void)
{
    P1OUT ^= GREEN; // Zmiana stanu
    Green LED //
    flag++; //
    inkrementacja zmiennej globalnej
    P1IFG &= ~przycisk; //
    Zerowana jest flaga przerwania:
    __bic_SR_register_on_exit( LPM4_bits ); //instrukcja
    która nakazuje zakończenie procedury obsługi przerwania
}

```