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

LABORATORIUM - GRUPA L2

Technika mikroprocesorowa

ĆWICZENIE 2.

Wykonali:

97938- Kamil Moczygęba
96249- Konrad Riabowski

Prowadzący:

MGR INŻ. ANDREAS KOWOL

1. Opis działania programu.

Celem programu jest realizacja funkcji przerwań. Po uruchomieniu programu powinna mrugać dioda zielona z zadaną częstotliwością (Time 1). Po naciśnięciu przycisku zmienia się częstotliwość mrugania diody zielonej (Time 2) oraz dioda czerwona osiąga stan wysoki. Po ponownym naciśnięciu przycisku program wraca do stanu po uruchomieniu.

2. Kod programu.

```
#include <msp430.h>
// Definicja zmiennych

#define SW      BIT3
#define GREEN   BIT6
#define RED     BIT0
#define Time1   20000
#define Time2   40000

volatile int flag=0;


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

    P1DIR |= GREEN + RED;               // Ustawienie wyjść
    P1DIR &= ~SW;                       //Ustawienie wejść

    P1REN |= SW;                        //Aktywacja rezystora pull-up
    P1OUT |= SW;

    P1IES &= ~SW;
    P1IE |= SW;                           //Włączana jest obsługa przerwań dla wejścia

    __bis_SR_register( GIE);             // Odblokowanie obslugi przerwan

    while(1)
     {
          P1OUT ^= RED;
          if(flag%2) __delay_cycles(Time1);
          else __delay_cycles(Time2);
     };
}

#pragma vector=PORT1_VECTOR
__interrupt void Port_1(void)
{
    P1OUT ^= GREEN;
    flag++;
    P1IFG &= ~SW;                                  // Zerowana jest flaga przerwania:
    __bic_SR_register_on_exit( LPM4_bits );        //instrukcja która nakazuje zakończenie procedury obsługi przerwania
}
```

3. Podsumowanie.

Z powodu braku płytki, której używaliśmy na laboratorium, nie mamy możliwości przetestowania programu. Jednak uważamy, że program, który przedstawiliśmy w tym sprawozdaniu będzie działał poprawnie.