

Wydział Informatyki Politechniki Białostockiej Laboratorium Architektury Komputerów	Data: 11.10.2021
Ćwiczenie nr 1 Temat: Sterowanie diodami przy użyciu mikrokontrolera MSP430 Realizacja: 1. Patryk Wójtowicz	Prowadzący: Dr. Inż. Mirosław Omieljanowicz Ocena:

1 CEL ĆWICZENIA

- Zapoznanie się z sterowaniem portami wejścia oraz wyjścia przy użyciu płytki MSP430 EasyWeb2.
- Obsługa diod i przycisków na płycie EasyWeb2.

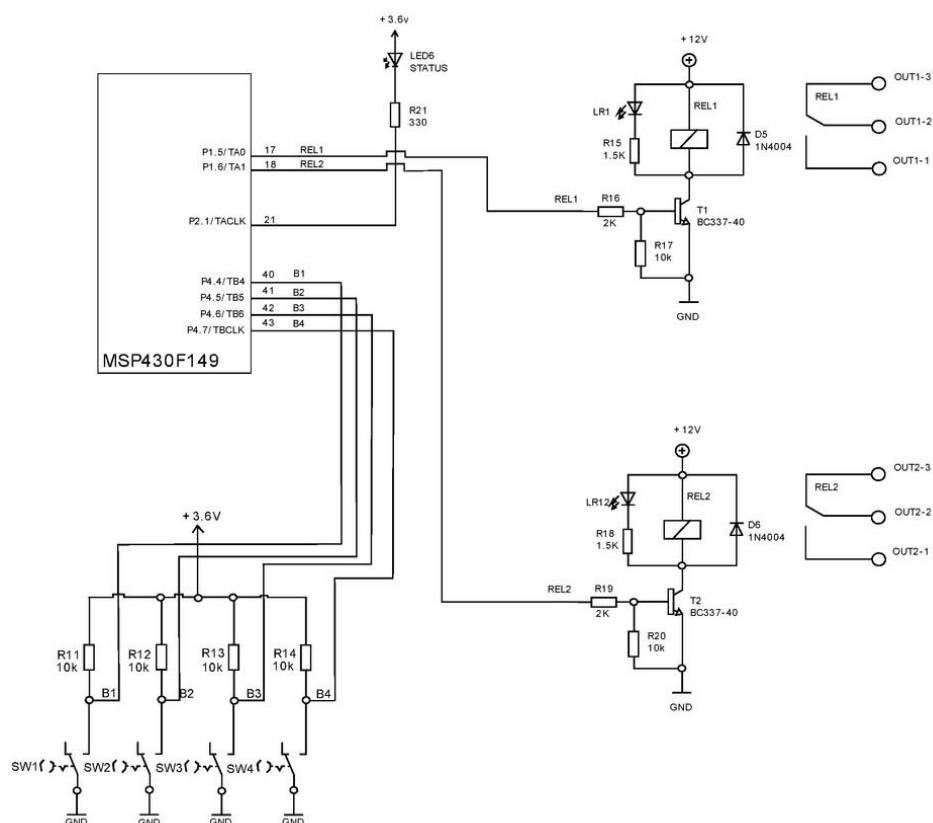
2 TEORIA

Kontrolery MSP430F149 posiadają 6 portów Input/Output. Porty te oznaczone są odpowiednio P1-P6. Porty P1, P2 obsługują pracę w przerwaniach a każdy port dodatkowo ma przyporządkowane 4 rejestry:

- Port Px direction PxDIR
- Port Px selection PxSEL
- Port Px output PxOUT
- Port Px Input PxIN

* x oznacza tutaj numer portu w tym przypadku od 1 do 6.

Płytką EasyWeb2 posiada 4 przyciski, którymi użytkownik może sterować inne urządzenia na płycie po wcześniejszym ich zaprogramowaniu. Przyciski są na portach od P.4.4 do P.4.7. Dodatkowo urządzenie posiada diody P.2.1 która odpowiada za status oraz diody P.1.5 oraz P.1.6.



3 OPIS ĆWICZENIA

Program zależnie od wciśniętego przycisku ma za zadanie wykonywać inne czynności:

1. Dla przycisku P.4.4 (P4 BIT4):
 - Zapalenie dwóch lampek jednocześnie (P.1.5 oraz P.1.6)
2. Dla przycisku P.4.5 (P4 BIT5):
 - Zapalanie na zmiennie lampki pierwszej (P.1.5) oraz (P.1.6) drugiej z takim odstępem by przypominało to galop konia.
3. Dla przycisku P.4.6 (P4 BIT6):
 - Zapalenie oraz zgaszenie lampki pierwszej (P.1.5) trzykrotnie, następnie powtórzenie czynności dla lampki drugiej (P.1.6)

```

1  #include <msp430x14x.h>
2
3  void main( void )
4  {
5      WDTCTL = WDTPW + WDTHOLD; // Wyłączenie układu WDT oraz inicjalizacja zegara
6
7      // |= Suma OR
8      // &= Iloczyn AND
9      // ~ Negacja NOT
10     // ^= Alternatywa Rozłączna XOR
11
12     P1DIR |= BIT5 + BIT6; // Ustawienie P1 Bitu 5 oraz 6 na wyjście
13     P4DIR &= ~BIT4; // Ustawienie P4 Bitu 4 na wejście
14
15     unsigned int i,j; // Deklaracja zmiennych typu integer i oraz j
16
17     while(1) // nieskończona pętla inna reprezentacja -> for(;;)
18     {
19         if((P4IN & BIT4) == 0) // Jeżeli guzik 1 jest wciśnięty zapal obie diody (P4IN <- wejście), ! Zero reprezentuje wciśnięcie !
20         {
21             P1OUT ^= BIT5 + BIT6; //Stan wysoki , przypisuje wartość 0 gdyż brak jest ~ na początku bitów
22         }
23         else if((P4IN & BIT5) == 0) // Jeżeli guzik 2 jest wciśnięty zapal diody w trybie "konika"
24         {
25             for(i=0;i<40000;i++); // Czas między wykonywanymi poleceniami inaczej interwał
26             P1OUT ^= BIT5;
27             for(i=0;i<40000;i++);
28             P1OUT ^= BIT6;
29         }
30         else if((P4IN & BIT6) == 0) // Jeżeli guzik 3 jest wciśnięty zapal każdą lampkę po 3 razy (jak kierunkowskaz)
31         {
32             for(j=0;j<6;j++) // Można zauważyć że mimo puszczenia guzika w trakcie wykonania schemat się dokończy
33             {
34                 if(j==0||j==1||j==2)
35                 {
36                     for(i=0;i<45000;i++) P1OUT ^= BIT5;
37                     for(i=0;i<45000;i++) P1OUT &= ~BIT5;
38                 }
39                 else
40                 {
41                     for(i=0;i<45000;i++) P1OUT ^= BIT6;
42                     for(i=0;i<45000;i++) P1OUT &= ~BIT6;
43                 }
44             }
45         }
46         else // Jeżeli guzik nie jest wciśnięty zgaś lampki
47         {
48             P1OUT &= ~BIT5 + ~BIT6; // Stan niski &= przypisuje wartość 1 gdyż jest ~ na początku bitów
49         }
50     }
51 }
52

```

4 WNIOSKI

Dzięki zastosowaniu mikrokontrolerów MSP można w łatwy sposób zaprogramować urządzenia przełączające. Zastawania tego typu mają niewyobrażalny wpływ na różnego rodzaju urządzenia w obecnych czasach.