

PRZEDMIOT:	Technika mikroprocesorowa		
KIERUNEK STUDIÓW:	<b>AUTOMATYKA I ROBOTYKA</b>	ROK STUDIÓW:	<b>3</b>
SPECJALNOŚĆ:	-		
SEMESTR:	<b>6</b>	ROK AKADEMICKI:	<b>2019/2020</b>

Temat ćwiczenia:

**Program 4 – Wykorzystanie UART do komunikacji**

<u>Ćwiczenie wykonali:</u>					
<u>Nazwisko:</u>		<u>Imię:</u>		<u>Nazwisko:</u>	
1.	<b>Janas</b>	<b>Kamil</b>	2.	<b>Indyk</b>	<b>Dominik</b>
3.	<b>Dittrich</b>	<b>Sebastian</b>			

<u>Uwagi:</u>	<u>Data:</u>	<u>Ocena za sprawozdanie:</u>

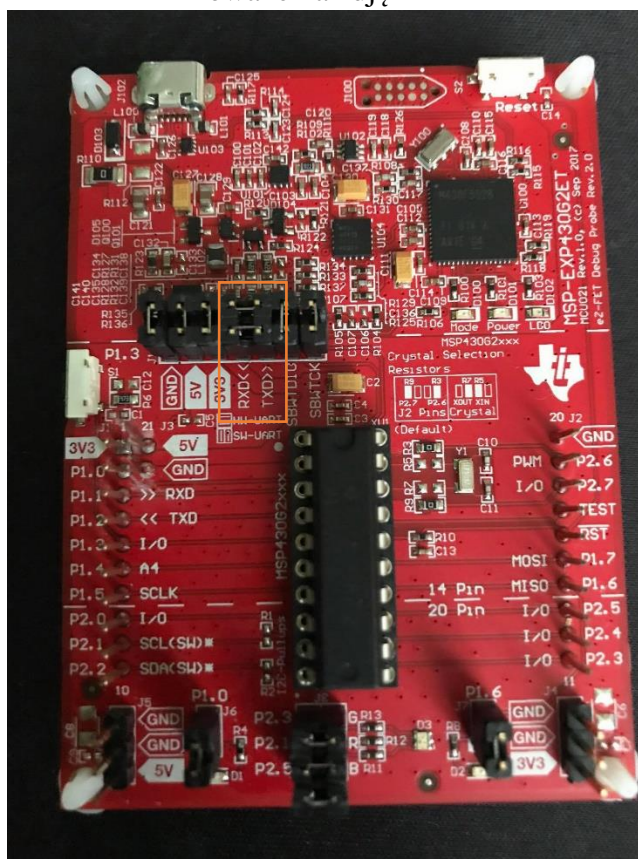
<u>Termin zajęć:</u>
<i>Zajęcia odbywały się zdalnie</i>

## 1. Cel ćwiczenia.

Głównym celem ćwiczenia jest napisanie programu wykorzystującego zaimplementowany w mikrokontrolerze MSP430 interfejsu UART, który pozwala na komunikację między mikrokontrolerami.

## 2. Opis ćwiczenia.

Ćwiczenie polegało na wykorzystaniu możliwości mikrokontrolerów MSP do komunikacji między sobą. Z powodu ograniczonych możliwości sprzętowych (posiadania jednego mikrokontrolera MSP) wykonano połączenie między mikrokontrolerem, a komputerem. Aby wykonać takowe połączenie należy wykorzystać zworki na pinach TXD oraz RXD. TXD wykorzystywane jest do wysyłania informacji natomiast RXD jest wykorzystywane do odczytu danych. Połączenie zwerek zostało zobrazowane na zdjęciu 1.



Zdj 1. Wykorzystanie zwerek do konfiguracji UART.

### 3. Kod skryptu programu.

```
1#include <msp430.h>
2// WYSYLANIE ZNAKU
3void print(char *text){
4    unsigned int i = 0;
5    while(text[i] != '\0')
6    {
7        // DOPÓKI TX JEST GOTOWY
8        while (!(IFG2&UCA0TXIFG));
9        // ZWRACAJ ZNAKI
10       UCA0TXBUF = text[i];
11       i++;
12    }
13}
14void main(void)
15{
16    WDTCTL = WDTPW + WDTHOLD;           // Stop Watchdog
17#define RED BIT6                        // Red LED P1.6
18#define GREEN BIT0                     // Green LED P1.0
19    P1DIR |= RED;                       // USTAWIENIE DIODY CZERWONEJ
20    P1OUT &= ~RED;                      // WYLACZENIE DIODY CZERWONEJ
21    P1DIR |= GREEN;                    // USTAWIENIE DIODY ZIELONEJ
22    P1OUT &= ~GREEN;                   // WYLACZENIE DIODY ZIELONEJ
23    // KONFIGURACJA UART:
24    if (CALBC1_1MHZ==0xFF)             // SPRAWDZAJ CZY KALIBRACJA JEST STALE USUNIĘTA
25    {
26        while(1);
27    }
28    DCOCTL = 0;                        // WYBIERZ NAJMNIEJSZĄ WARTOŚĆ
29    BCSCTL1 = CALBC1_1MHZ;             // USTAW WARTOŚĆ 1MHZ
30    DCOCTL = CALDCO_1MHZ;
31    // USTAWIENIE PINÓW RX I TX:
32    P1SEL = BIT1 + BIT2 ;
33    P1SEL2 = BIT1 + BIT2;
34    // TRANSMISJA UART:
35    UCA0CTL1 |= UCSSEL_2;              // UART Clock -> SMCLK
36    UCA0BR0 = 104;                     // Baud Rate Setting for 1MHz 9600
37    UCA0BR1 = 0;                       // Baud Rate Setting for 1MHz 9600
38    UCA0MCTL = UCBRS_1;                // Modulation Setting for 1MHz 9600
39    UCA0CTL1 &= ~UCSWRST;              // Initialize UART Module
40    //USTAWIENIA PRZERWANIA:
41    IE2 |= UCA0RXIE;
42    __bis_SR_register(LPM0_bits + GIE); // Enter LPM0, Enable Interrupt
43}
```

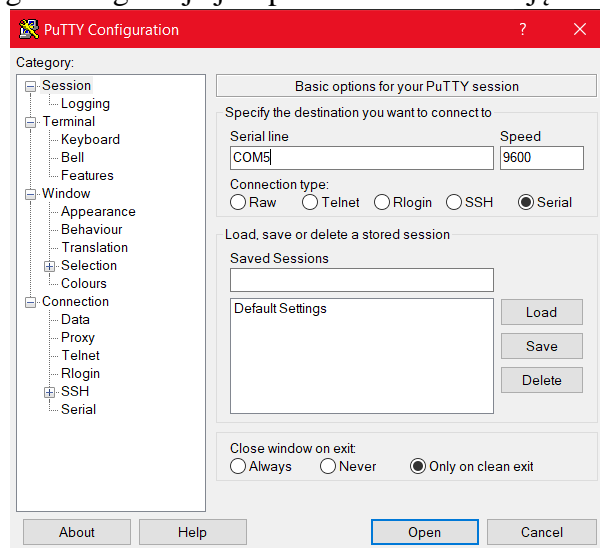
```

44// WCISNIECIE PRZYCISKU SPOWODUJE PRZERWANIE
45#pragma vector=USCIAB0RX_VECTOR
46__interrupt void USCI0RX_ISR(void)
47{
48    //ODBIERAJ ZNAKI DOPOKI:
49    while (!(IFG2&UCA0TXIFG));
50    UCA0TXBUF = UCA0RXBUF;
51    if (UCA0TXBUF=='d' || UCA0TXBUF=='D'){
52        print("\t- W grupie jest osoba, której imię zaczyna się na literę D to Dominik\r\n");
53        print("\t Ta osoba podlega ocenie.\r\n");
54        P1OUT |=GREEN;
55        P1OUT &= ~RED;
56    }
57    else if (UCA0TXBUF=='s' || UCA0TXBUF=='S'){
58        print("\t- W grupie jest osoba, której imię zaczyna się na literę S to Sebastian\r\n");
59        print("\t Ta osoba podlega ocenie.\r\n");
60        P1OUT |=GREEN;
61        P1OUT &= ~RED;
62    }
63    else if (UCA0TXBUF=='k' || UCA0TXBUF=='K'){
64        print("\t- W grupie jest osoba, której imię zaczyna się na literę K to Kamil\r\n");
65        print("\t Ta osoba podlega ocenie.\r\n");
66        P1OUT |=GREEN;
67        P1OUT &= ~RED;
68    }
69    else{
70        print("\t Nie ma w grupie osoby , której imię zaczyna się na literę\r\n");
71        print("\t- To nie dla tej osoby ocena\r\n");
72        P1OUT &= ~GREEN;
73        P1OUT |=RED;
74    }
75}

```

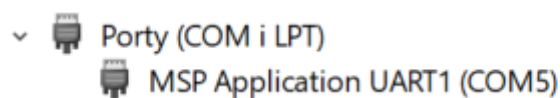
#### 4. Opis programu.

Program wykorzystuje interfejs komunikacyjny UART. Wykorzystanie to jest stosowane poprzez komunikację mikrokontrolera z komputerem za pomocą programu putty.exe, którego konfiguracja jest przedstawiona na zdjęciu 2.



Zdj 2. Ustawiono typ połączenia jako serial oraz wpisano linie połączenia COM5.

Linia połączenia (port) jest odczytywana w menadżerze urządzeń jak poniżej:

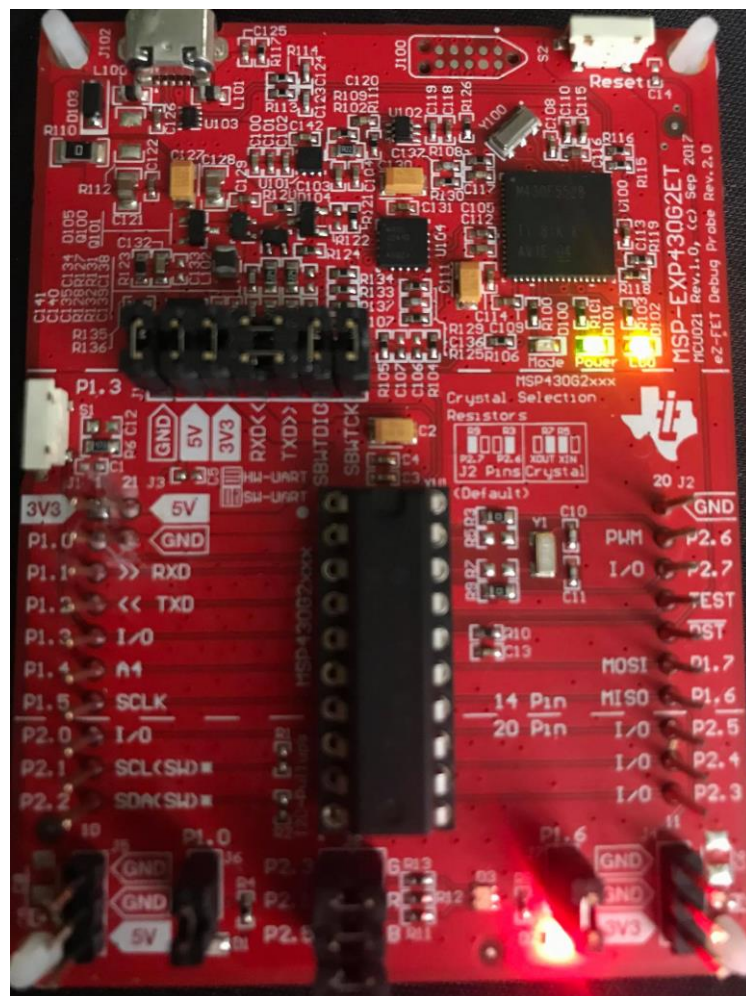


Zdj 3. Odczyt numeru portu COM.

Po wykonaniu połączenia napisany kod realizował krótką interakcję pomiędzy komputerem, a mikrokontrolerem. Interakcja polegała na tym, że wciskając przycisk na klawiaturze MSP sprawdza czy jest to przycisk odpowiedzialny za pierwsze litery imion osób wchodzących w skład sekcji na tym ćwiczeniu. Jeśli nie na konsoli wyświetla się komunikat o źle wybranej literze oraz zapala się czerwona dioda P1.6 co ukazuje zdjęcie 4 oraz 5:

```
COM5 - PuTTY
1      Nie ma w grupie osoby , której imię zaczyna się na literę
      - To nie dla tej osoby ocena
k      - W grupie jest osoba, której imię zaczyna się na literę K to Kamil
      Ta osoba podlega ocenie.
z      Nie ma w grupie osoby , której imię zaczyna się na literę
      - To nie dla tej osoby ocena
```

*Zdj 4. Komunikat z konsoli putty.exe.*



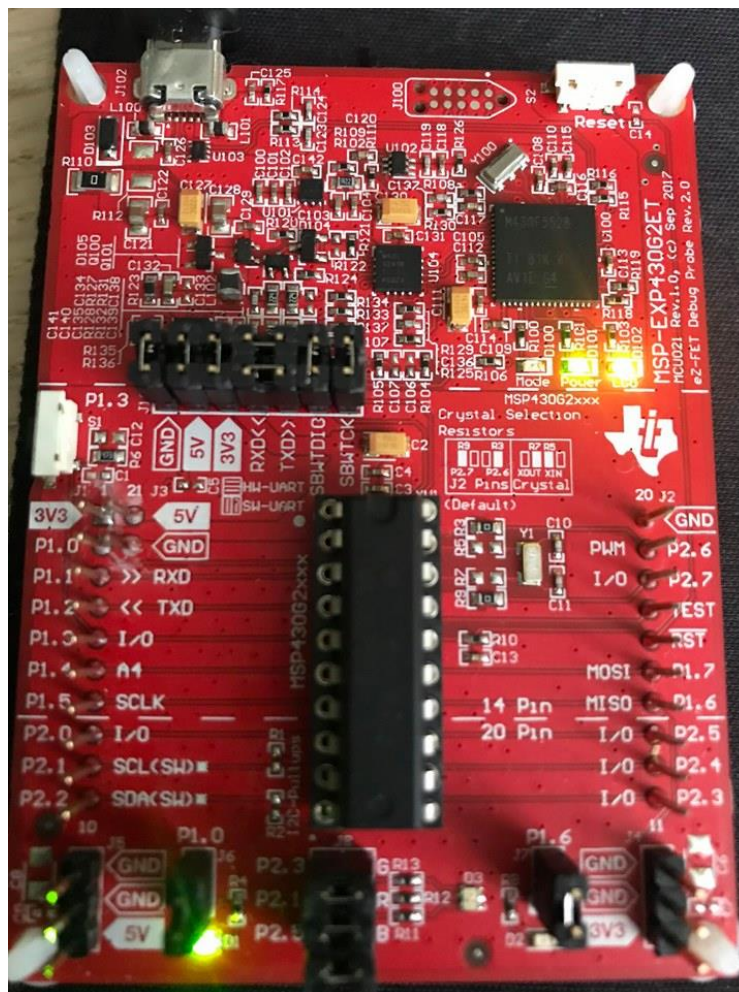
*Zdj 5. Odpowiedź układu na powyżej wybrane litery.*



Natomiast wybranie przez użytkownika litery D, S lub K powoduje wyświetlenie się komunikatu z poniższego zdjęcia oraz zapalenie się diody zielonej P1.0:

```
COM5 - PuTTY
D   - W grupie jest osoba, której imię zaczyna się na literę D to Dominik
    Ta osoba podlega ocenie.
S   - W grupie jest osoba, której imię zaczyna się na literę S to Sebastian
    Ta osoba podlega ocenie.
K   - W grupie jest osoba, której imię zaczyna się na literę K to Kamil
    Ta osoba podlega ocenie.
```

Zdj 6. Komunikat z konsoli putty.exe.



Zdj 7. Odpowiedź układu na powyżej wybrane litery.

## 5. Wnioski.

Powyżej przedstawiono przykładowe zastosowanie komunikacji sprzętowej UART pozwalające na tworzenie wielu różnych rozwiązań interakcji między komputerem, a mikrokontrolerem. Tym prostym przykładem zaprezentowano oraz potwierdzono poprawne działanie interfejsu względem napisanego kodu w programie CodeComposer.