

<b>Grupa:</b> B	<b>Temat:</b> Obsługa przycisków		
<b>Data:</b> 05.05.2021	<b>Wykonał:</b> Piotr Drabik		
<b>Godzina:</b> 12:30	<b>II rok Informatyka</b> <b>Stosowana</b>	<b>Ocena i uwagi</b> <b>prowadzącego:</b>	<b>Prowadzący:</b> dr hab. Witold Kozłowski

## Opis ćwiczenia

Celem ćwiczenia jest poznanie metodyki obsługi przycisków, klawiatury matrycowej i klawiatury komputerowej AT.

Klawiatura umożliwia użytkownikowi wprowadzanie danych do działającego już programu procesora. Wraz z wyświetlaczem tworzą podstawowy komplet urządzeń wejścia wyjścia.

Płytką procesora wyposażoną jest w 4 przyciski pozwalające na komunikację z programem, lecz to niewiele, lepszym rozwiązaniem jest zewnętrzna klawiatura posiadająca znacznie więcej przycisków i nieograniczoną ilość ich kombinacji.

Lecz podobnie jak w przypadku poprzednich zajęć stajemy przed problemem dużej ilości styków które należy obsłużyć. każdy przycisk w klawiaturze należy zaadresować i połączyć z procesorem. To wymaga dużego nakładu pieniężnego i wielu szpilek miedzi. Zamiast bezpośredniego podejścia stosuje się podejście matrycowe. Cechuje się ono dokładnym adresowaniem dużej ilości przycisków przy zastosowaniu minimalnej ilości linii adresowych. Linie układane są w rzędach i kolumnach w taki sposób by pomiędzy zestawem rzędów i kolumn trwały się przecięcia, na których układamy przyciski. Po podłączeniu przycisków w odpowiedni sposób, włączony przycisk aktywować będzie jedno z połączeń w zestawie rzędów i jedno w kolumnach, wiedząc która z linii w rzędach jest aktywowana i która w kolumnach możemy wywnioskować jaki przycisk jest wciśnięty.

To pozwala nam na oszczędność pieniędzy miedzi i czasu, lecz program musi być odpowiednio dostosowany. Musi istnieć układ tłumaczący dwie wartości (aktywowane połączenie w zestawie rzędów i kolumn) taki układ posiada każda klawiatura.

Przyciski nęka jeszcze jeden problem, ponieważ są to układy fizyczne, połączenia które łączą procesor z przyciskami nie są idealne, mogą na nich występować fluktuacje natężenia jak i same przyciski mogą w niewielki sposób drgać, tego typu błędy prowadzą do błędnych odczytów.

By wyeliminować te przypadki stosuje się algorytm nazwany debouncerem, jest to układ aktywowany po wciśnięciu przycisku, który ponownie bada stan przycisku po upływie kilkudziesięciu milisekund. Jeżeli po upływie tego czasu przycisk nadal jest wciśnięty, prawdopodobnie wciśnięty jest celowo, dopiero wtedy procesor informowany jest o wciśniętym przycisku.

# Instrukcje

---

- **Debounce** - odpowiedź na problem błędów występujących na połączeniach przycisków
- **Getatkbd** - pobranie z klawiatury id wciśniętego przycisku
- **IF .. THEN** - warunek pozwalający rozstrzygnąć stan przycisku

## Komponenty używane przy przeprowadzaniu ćwiczenia

---

- Wyświetlacz LCD
- Przyciski umieszczone na płytce procesora

## Program obsługujący Klawiaturę matrycową

---

```
$regfile = "m8def.dat"           Informuje kompilator o pliku dyrektyw
mikroprocesora
$crystal = 8000000               Informuje kompilator o częstotliwości oscylatora
taktuującego
                                mikrokontroler

Config PINB.0 = Input
Config PINB.1 = Input

Config PINB.2 = Output
Config PINB.3 = Output

Config Lcd = 16*2
Config Timer0 = Timer , Prescale = 1024

On timer0 Mult_k1

Dim temp1 As Byte
Dim temp2 As Byte

Dim button As Byte
Dim i As Byte

row1 Alias PINB.0
row2 Alias PINB.1

kol1 Alias PINB.2
kol2 Alias PINB.3

Enable Interrupts
Enable Timer0
La Timer0, 200

Set PORTB.0
Set PORTB.1
```

```

Set kol1
Set kol2

Do
    Cls
    Lcd button
    Waitms 200
Loop
End

Mul_k1:
    Load Timer0, 200
    For i = 1 To 2
        If i = 1 Then
            Reset kol1
        Else
            Set kol1
            Reset kol2
        End If
        If row1 = 0 Or row2 = 0 Then
            temp1 = PINB And &B00000011
            Exit For
        Else
            Temp1=0
        End If
    Next i
    IF temp2 = temp1 Then
        button = temp1
        IF i=2 Then
            button = button + 2
        End IF
    Else
        temp2 = temp1
    End IF

    Set kol1
    Ser kol2

Return

```

# Program wąż, reagujący na wciśnięcie przycisku płytki

---

```
$regfile = "m8def.dat"
$crystal = 8000000
Config Portd = Output
Config Pinb.0 = Input

Dim direction As Bit

button Alias Pinb.0

Set Portb.0

Portd = &B11111110
Reset direction

Do
  If button = 0 Then
    Waitms 50
    Toggle direction

    Do
      Loop Until button = 1

  End If
  If direction = 1 Then

    Rotate Portd, Left
  Else
    Rotate PORTD , Right
  End If
  Waitms 100
Loop
End
```

## Podsumowanie

---

Klawiatura to potężne i konieczne narzędzie, umożliwia ona aktywną komunikację z działającym programem. Lecz klawiatura składająca się z kilku przycisków nie jest dużą pomocą dlatego naturalnym jest stosowanie techniki matrycowania klawiatury. Pozwala ona w znacznym stopniu zwiększyć potencjał naszego układu.

Lecz klawiatura jest fizycznym komponentem układu cyfrowego, dlatego należy stosować cyfrowe techniki walki z defektami świata fizycznego, takimi jak błędne rozpoznawanie wciśniętego przycisku spowodowane wibracjami na sykach. By rozwiązać ten problem stosuje się "debouncer", algorytm próbujący san przycisku w czasie.