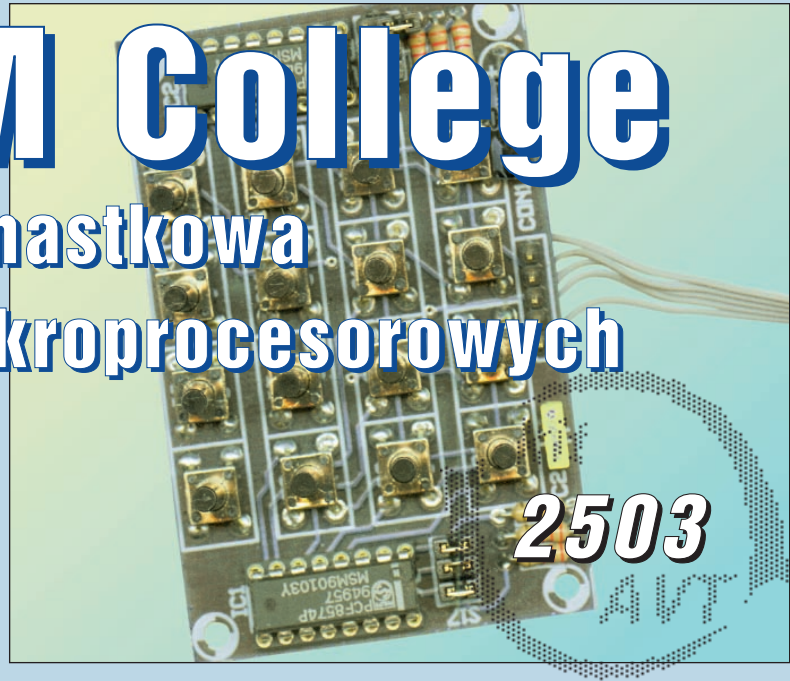
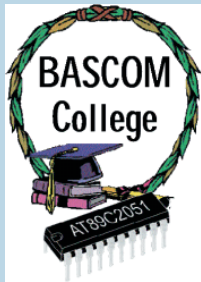




# BASCOM College

## Klawiatura szesnastkowa do systemów mikroprocesorowych



### Do czego to służy?

Prawie każdy system mikroprocesorowy musi posiadać wbudowane układy, które umożliwiają mu komunikację ze światem zewnętrznym. Procesor musi skądś otrzymywać dane, przetwarzać je i wysyłać dalej, najczęściej do interfejsu czytelny dla człowieka. Najlepszym przykładem takiego systemu jest komputer PC. Posiada on szereg układów przechwytyjących informację ze świata zewnętrznego, czyli klawiaturę, myszkę, modem telefoniczny lub skaner. Informacja przetworzona przez komputer przekazywana jest jego użytkownikowi za pomocą monitora, karty dźwiękowej, lub drukarki. O ile jednak dodawanie dodatkowych układów komunikacyjnych do komputera nie sprawia prawie nigdy większych trudności, to inaczej ma się sprawa z miniaturowymi komputerkami jakimi są nasze systemy mikroprocesorowe. Przeszkodą jest tu przede wszystkim ograniczona ilość wyprowadzeń nawet "największego" procesora. Na szczęście, projektanci systemów mikroprocesorowych wymyślili kilka standardów przekazywania informacji pomiędzy elementami systemu, a także do i ze świata zewnętrznego. Jednym z najpopularniejszych sposobów transmisji danych jest dwuprzewodowa magistrala I<sup>2</sup>C.

Specyfikacją magistrali I<sup>2</sup>C zajmiemy się w najbliższym czasie, a na razie zadowolimy się tylko informacją, że w dwuprzewodowej magistrali dane mogą być przekazywane w obydwóch kierunkach: od układu nadrzędnego (master) do układów podporządkowanych (slave), a także w odwrotnym kierunku. Magistrala I<sup>2</sup>C została tak zaprojektowana, że w zasadzie nic nie ogranicza ilości dołączonych do niej urządzeń. Warunek jest jeden, a jego niespełnienie może prowadzić do nieprawidłowej transmisji danych: każde

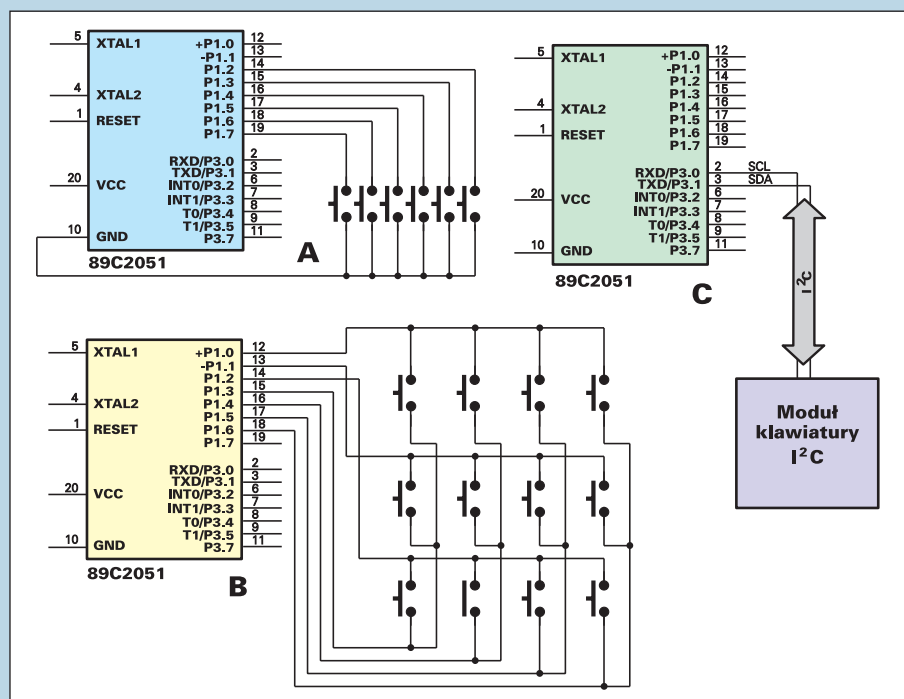
urządzenie dołączone do magistrali musi posiadać swój własny, niepowtarzalny adres. Magistrala I<sup>2</sup>C jest szeroko stosowana w profesjonalnych systemach mikroprocesorowych, m. in. w sterowanym cyfrowo sprzęcie video i audio.

Jednym z najważniejszych elementów umożliwiających wydawanie rozkazów i przekazywanie informacji do procesora przez człowieka, jest klawiatura. Nie musi być to bynajmniej klawiatura typu "komputerowego". Niekiedy za klawiaturę uważamy zespół kilku klawiszy, a nawet jeden klawisz. Tak czy inaczej klawisze te muszą być jakoś dołączone do procesora i w każdym wypad-

ku zajmują pewną ilość cennych wyprowadzeń układu.

Na **rysunku 1** pokazane zostały trzy najpopularniejsze sposoby łączenia klawiszy z procesorem. Najprostszą metodą połączenia klawiatury do procesora jest bezpośrednie połączenie klawiszy z wejściami układu, pokazane na części rysunku oznaczonej "A". Jest to także najprostsze rozwiązanie z punktu widzenia programisty, ale posiada jedną, przy większej ilości klawiszy dyskwalifikującą je wadę: na każdy klawisz przypada jedno wyprowadzenie procesora.

Rys. 1



Często stosowanym rozwiązaniem jest klawiatura matrycowa, której schemat został pokazany w części rysunku "B". Zależność pomiędzy ilością klawiszy, a ilością użytych wyprowadzeń procesora jest tu nieco lepsza niż w poprzednim przykładzie. Do obsługi klawiatury 12 przyciskowej potrzebne jest tylko 3+4 czyli 7 wyprowadzeń, klawiatura 16 przyciskowa wymagać będzie zaabsorbowania 8 nóżek procesora. To już trochę lepiej, ale w przypadku procesorów 89CX051 o 15 aktywnych wyprowadzeniach, też bardzo dużo. Ponadto, programowa obsługa klawiatury matrycowej (nie będziemy jej w tym miejscu opisywać) jest znacznie bardziej skomplikowana niż pojedynczych przycisków i wymaga stałego zaangażowania ze strony skanującego ją procesora.

Moim zdaniem, najlepszym sposobem skonstruowania klawiatury o większej ilości przycisków jest zastosowanie dekodery umożliwiających przesyłanie ośmiobitowego słowa wejściowego do procesora za pomocą dwuprzewodowej magistrali I<sup>2</sup>C ("C"). Najlepszym przykładem takiej klawiatury jest konsola komputera PC, z której dane przekazywane są także za pomocą dwóch przewodów (nie jest to jednak magistrala I<sup>2</sup>C, ale idea pozostaje taka sama).

Układ, z którego budowę zapoznamy się za chwilę zaprojektowany został jako uzupełnienie naszej płytki do testowania prototypowych układów procesorowych i nauki programowania w systemie BASCOM. Nic jednak nie stoi na przeszkodzie, aby zastosować go w układzie praktycznym, jako moduł ułatwiający budowę systemu. Już teraz mogą odsłonić Wam rąbek tajemnicy otaczającej czekającą nas przyszłość. Na opublikowanie jej opisu czeka kolejna uniwersalna płytka do budowy systemów mikroprocesorowych. Tym razem nie będzie to jednak tablica o sporych wymiarach, wygodna jedynie podczas nauki i te-

stowania nowych pomysłów. Nowa płytka ma wymiary typowego wyświetlacza LCD (dokładnie), ale udało mi się zmieścić na niej większość elementów potrzebnych do budowy typowych układów procesorowych. Posłużmy się prostym przykładem: dołączmy do tej płytki znany już Wam moduł wyświetlaczy siedmiosegmentowych LED, klawiaturę i ... mamy gotowy zegar o zależnej tylko od inwencji programisty ilości funkcji.

### Jak to działa?

Schemat elektryczny układu klawiatury został pokazany na **rysunku 2**. Dwoma podstawowymi elementami układu są znane już Wam terminale typu PCF8574A, umożliwiające wymianę danych pomiędzy magistralą I<sup>2</sup>C, a równoległą, ośmiobitową szyną danych. W przypadku modułu wyświetlaczy siedmiosegmentowych układy PCF przekazywały informację z magistrali do dekodery wyświetlaczy siedmiosegmentowych, a teraz stoi przed nimi odwrotne zadania: zbieranie informacji ze styków klawiszy i przekazywanie jej w postaci szeregowej do procesora. Zasadę działania układu najlepiej będzie poznać na podstawie analizy metod programistycznych stosowanych do jej obsługi.

Każdy z układów PCF8574 posiada osiem wejść - wyjść typu open collector, z wewnętrznymi rezystorami podciągającymi wejścia

w stronę plusa zasilania. A zatem, aby cokolwiek z wejść układu PCF8574 odczytać, musimy najpierw ustawić na nich stan wysoki. W tym celu należy zainicjować transmisję w magistrali I<sup>2</sup>C, przygotować odbiornik do przyjęcia informacji, a następnie wysłać do niego wartość 255, czyli binarnie 1111 1111. Wygląda to dość skomplikowanie, ale na szczęście dysponujemy naszym BASCOM'em. W języku MCS BASIC wszystkie wymienione operacje wykonywane są po wydaniu tylko jednego polecenia:

**I2CSEND** [adres odbiornika],  
[wartość, którą chcemy wysłać]

W naszym konkretnym przypadku polecenie to będzie miało postać:

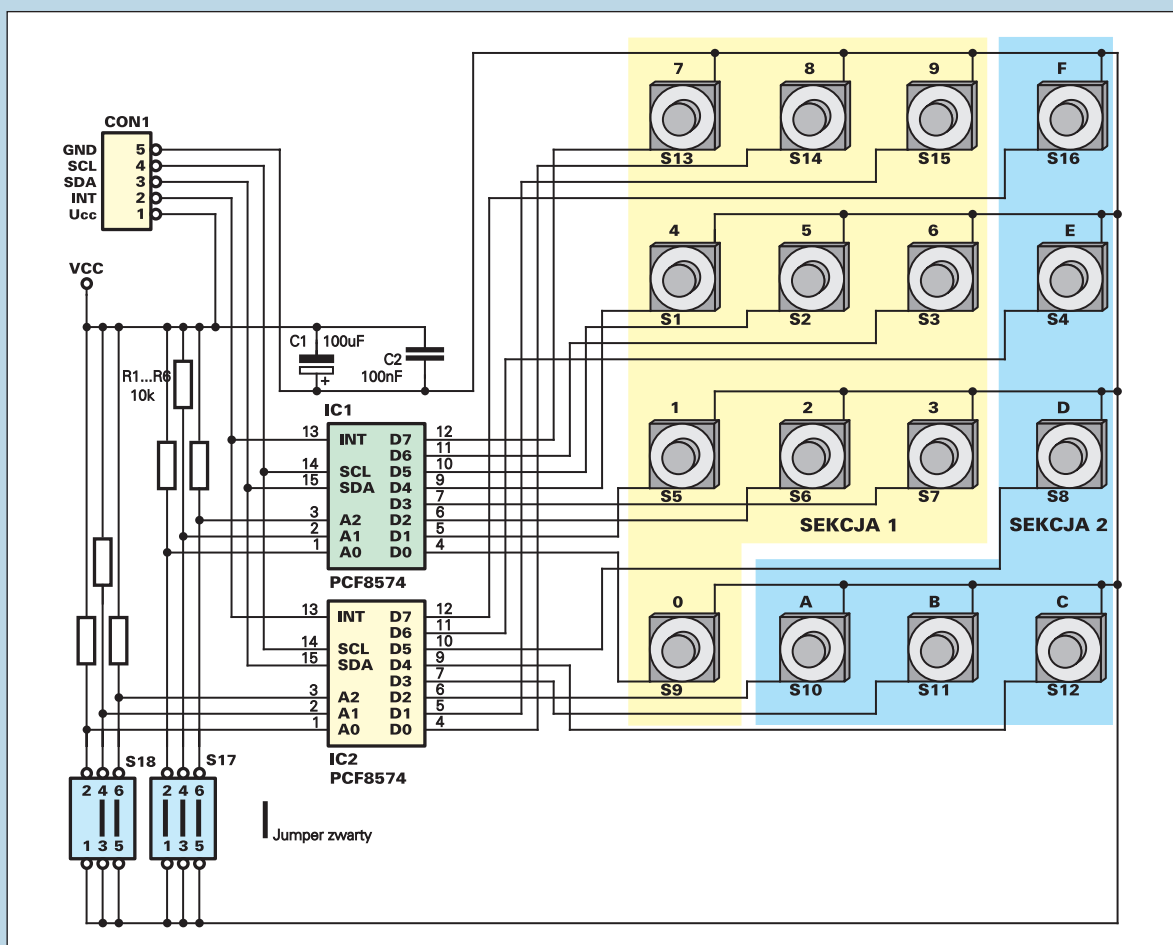
**I2CSEND 112, 255**  
(dla układu PCF z wszystkimi wejściami adresowymi w stanie niskim)

lub

**I2CSEND 112, 255** (dla układu PCF z wejściami adresowymi w stanie 001 (BIN))

Po wydaniu tych poleceń wszystkie wejścia układów IC1 i IC2 przyjmują stan

Rys. 2 Schemat ideowy



wysoki i układy te gotowe są do przyjęcia informacji z klawiatury.

Programista piszący program obsługujący naszą klawiaturę będzie miał teraz dwie możliwości: albo procesor będzie nieustannie sprawdzał stan wejść układów IC1 i IC2, albo wykorzysta sygnał przerwania występujący na wyjściach INT przy jakiegokolwiek zmianie stanu wejść układów PCF. Sprawa obsługi przerwań zostanie wyczerpująco omówiona w BASCOM College, a w tym momencie wystarczy tylko wspomnieć, że po wystąpieniu przerwania procesor przestaje wykonywać jakiejkolwiek czynności i natychmiast przystępuje do realizowania obsługi przerwania, czyli wykonania instrukcji zawartych w specjalnym podprogramie (w naszym przypadku będzie to sprawdzenie stanu klawiatury).

W każdym jednak wypadku odczytu danych dokonujemy za pomocą polecenia:

```
I2CRECEIVE [adres nadajnika],
           [zmienna]
```

Czyli konkretnie:

```
I2CRECEIVE 66, [nazwa zmiennej]
(dla układu PCF z wejściami adresowymi
 w stanie 001 (BIN))
```

lub

```
I2CRECEIVE 66, [nazwa zmiennej]
(dla układu PCF z wejściami adresowymi
 w stanie 001 (BIN))
```

Po odebraniu wartości innej niż 255, czyli w przypadku naciśnięcia przez użytkownika jakiegoś klawisza procesor przystępuje do

jej analizy, a następnie ponownie ustawia na wejściach układów PCF8574 stan wysoki przygotowując je w ten sposób do oczekiwania na naciśnięcie kolejnego klawisza.

Warto zauważyć, że nasza klawiatura została podzielona na dwie sekcje i wszystkie opisane czynności muszą donosić się kolejno do każdej z nich.

Należy jeszcze wspomnieć o roli jaką pełnią jumpery S18 i S17. Za ich pomocą ustawiamy adresy układów PCF tak, aby procesor mógł je jednoznacznie zidentyfikować. Każdy z stosowanych w systemie układów PCF musi mieć swój indywidualny adres. Jeżeli więc będziemy używać naszej klawiatury np. jednocześnie z modulem wyświetlaczy siedmiosegmentowych, w którym

## Montaż i uruchomienie

Na rysunku 3 została pokazana mozaika ścieżek płytki obwodu drukowanego oraz rozmieszczenia na niej elementów. Z uwagi na znaczną komplikację połączeń płytka została wykonana na laminacie dwustronnym z metalizacją. Montaż układu klawiatury wykonujemy w typowy sposób, rozpoczynając od wlotowania w płytkę rezystorów i podstawek pod układy scalone, a kończąc na zamontowaniu przycisków microswitch. Te ostatnie elementy musimy wlotować szczególnie starannie, bacząc aby wszystkie przyciski zostały ustawione idealnie równo względem powierzchni płytki.

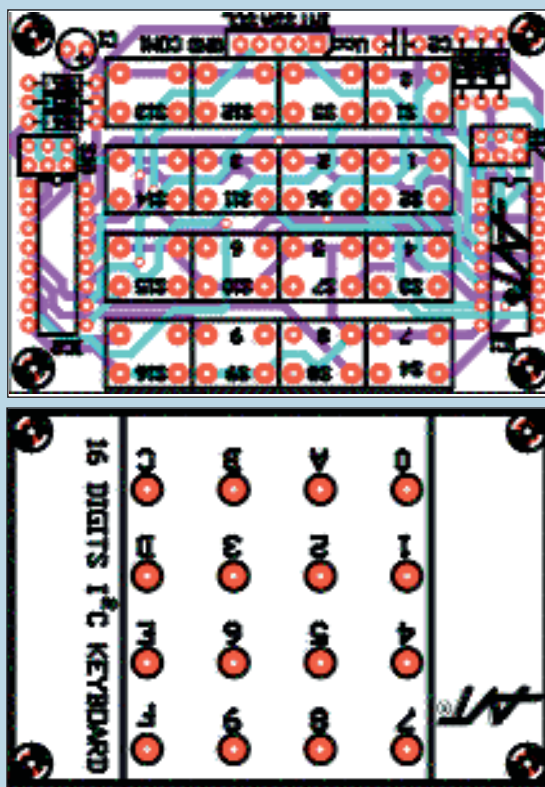
Do dyspozycji mamy jeszcze jedną płytkę, którą wprawdzie nie sposób nazwać płytką obwodu drukowanego. Może ona natomiast posłużyć do wykonania w miarę estetycznej płyty czołowej klawiatury, która ponadto stabilizuje położenie przycisków i czyni całą konstrukcję odporną na urazy mechaniczne. Sposób połączenia obydwóch płytek jest wyraźnie widoczny na fotografiach.

Warto jeszcze zaznaczyć, że napi-

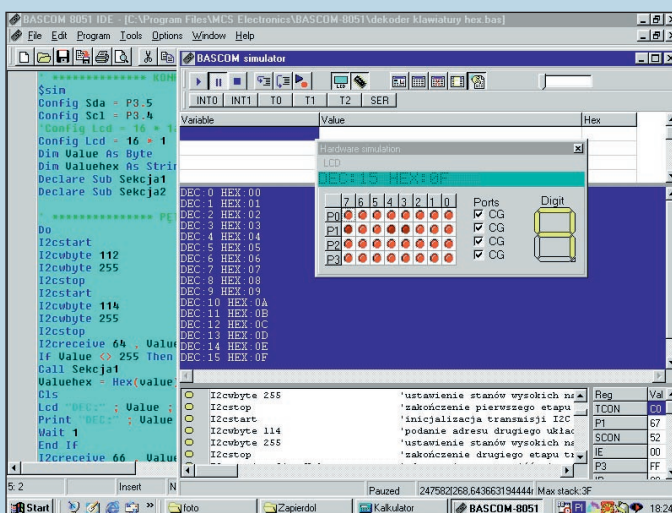
sy umieszczone na płycie czołowej klawiatury odnoszą się jedynie do wartości liczbowej wprowadzanych danych. W wielu zastosowaniach praktycznych trzeba będzie je zastąpić opisem czynności, jakie procesor ma wykonać po naciśnięciu danego klawisza.

Wykonany układ powinniśmy jeszcze przetestować, co możemy uczynić za pomocą naszej płytki testowej. Nie musimy w tym celu jeszcze programować procesora, ani nawet go posiadać. Piszemy w edytorze BASCOM'a program, którego listing został zamieszczony poniżej, kompilujemy go, a następnie po połączeniu płytki testo-

Rys. 3a Schemat montażowy



Rys. 5



## Wykaz elementów

### Kondensatory

C1 .....100µF/10  
C2 .....100nF

### Rezystory

R1 ... R6 .....10kΩ

### Półprzewodniki

IC1, IC2 .....PCF8574A

### Pozostałe

S1 ... S16 .....przyciski typu microswitch  
S17, S18 .....3x2 goldpin + 3 jumpery  
CON1 .....5x goldpin

**Komplet podzespołów z płytką jest dostępny w sieci handlowej AVT jako kit szkolny AVT-2503**



wej klawiaturą i emulatorem sprzętowym, uruchamiamy. Jeżeli wszystko jest w porządku, to po naciśnięciu któregośkolwiek klawisza na wyświetlaczu LCD powinna okazać

się odpowiadająca mu wartość, wyrażona w kodzie decymalnym i hexadecymalnym. Te same wartości będą przekazywane także na ekran monitora emulatora programowego i do emulowanego programowo wyświet-

tlacza LCD, o ile emulacja programowa zostanie także włączona (rys. 5).

Sposób połączenia wszystkich elementów potrzebnych do przetestowania klawiatury został pokazany na rysunku 4.

```

'***** KONFIGURACJA SPRZĘTOWA *****
$sim          'nie stosować w przypadku testowania programu z zaprogramowanym 'procesorem
Config Sda = P3.5      'konfiguracja sprzętowa I2C
Config Scl = P3.4      'konfiguracja sprzętowa I2C
'Config Lcd = 16 * 1a  'stosować w przypadku testowania programu z zaprogramowanym '
    'procesorem
Config Lcd = 16 * 1    'stosować w przypadku testowania programu z emulatorem sprzętowym
Dim Value As Byte     'deklaracja zmiennej (wartości pobieranej z klawiatury)
Dim Valuehex As String * 2 'deklaracja ciągu znakowego o długości 2 znaków
Declare Sub Sekcja1   'deklaracja 1 podprogramu analizującego kod odebrany z klawiatury
Declare Sub Sekcja2   'deklaracja 1 podprogramu analizującego kod odebrany z klawiatury

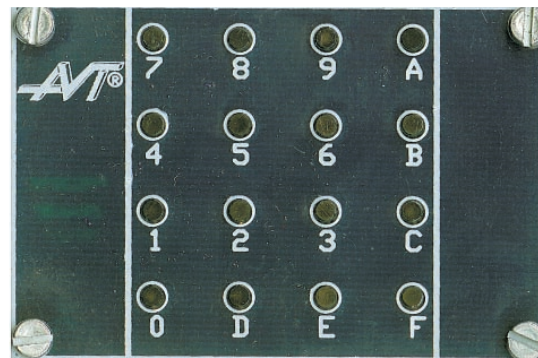
'***** PĘTLA GŁÓWNA PROGRAMU *****
Do          'początek pętli programowej

I2csend 112, 255 'ustawienie stanów wysokich na wyjściach pierwszego układu PCF8574A
I2csend 114, 255 'ustawienie stanów wysokich na wyjściach drugiego układu PCF8574A
I2creceive 64 , Value 'odczytanie stanu wejść pierwszego układu PCF8574A
If Value <> 255 Then 'jeżeli naciśnięty został klawisz w sekcji 1 to:
Call Sekcja1        'idź do podprogramu analizującego sekcji 1
Valuehex = Hex(value) 'konwersja odczytanej wartości na kod HEX
Cls                 'czyszczenie ekranu LCD
Lcd "DEC:" ; Value ; " HEX:" ; Valuehex 'wysłanie na wyświetlacz LCD otrzymanych wartości w formacie 'DEC i HEX
Print "DEC:" ; Value ; " HEX:" ; Valuehex 'wysłanie do portu RS otrzymanych wartości
Wait 1             'zwłoka 1 sek
End If             'zakończenie uwarunkowań
I2creceive 66 , Value 'odczytanie stanu wejść drugiego układu PCF8574A
If Value <> 255 Then 'jeżeli naciśnięty został klawisz w sekcji 2 to:
Call Sekcja2        'idź do podprogramu analizującego sekcji 2
Valuehex = Hex(value) 'konwersja odczytanej wartości na kod HEX
Cls                 'czyszczenie ekranu LCD
Lcd "DEC:" ; Value ; " HEX:" ; Valuehex 'wysłanie na wyświetlacz LCD otrzymanych wartości
Print "DEC:" ; Value ; " HEX:" ; Valuehex 'wysłanie do portu RS otrzymanych wartości
Wait 1             'zwłoka 1 sek
End If             'zakończenie uwarunkowań
Loop              'zamknięcie pętli programowej

'*****PODPROGRAMY KONWERTUJĄCE KODY OTRZYMANE Z KLAWIATURY *****

Sub Sekcja1
If Value = 254 Then
Value = 0
End If
If Value = 253 Then
Value = 1
End If
If Value = 251 Then
Value = 2
End If
If Value = 247 Then
Value = 3
End If
If Value = 239 Then
Value = 4
End If
If Value = 223 Then
Value = 5
End If
If Value = 191 Then
Value = 6
End If
If Value = 127 Then

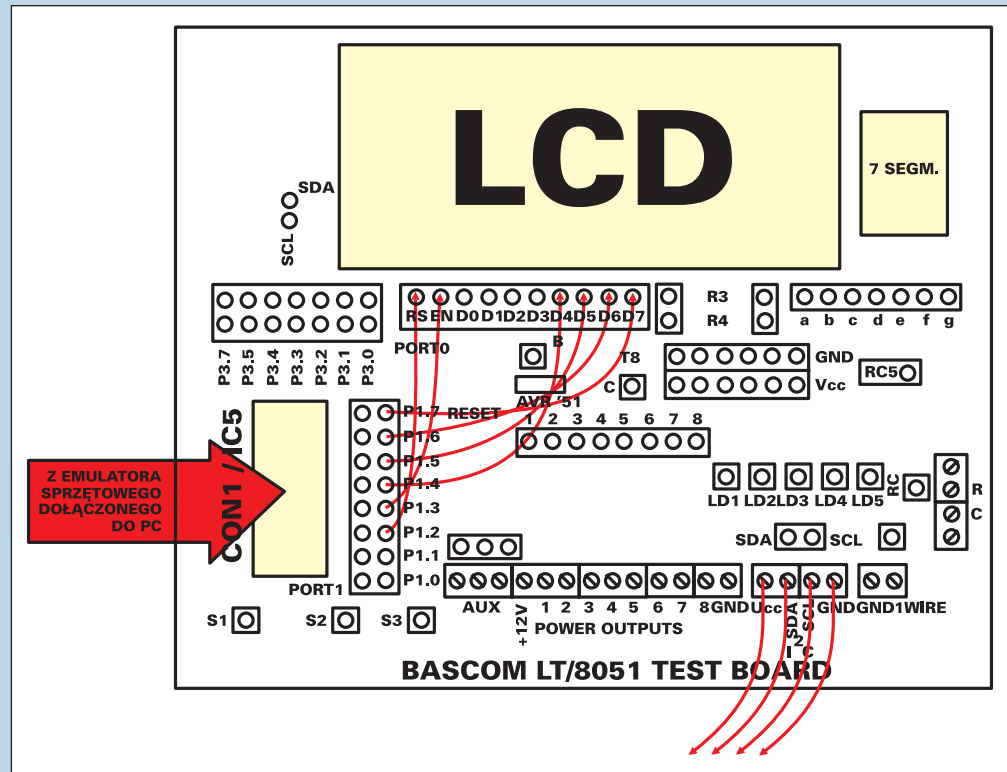
```



```

Value = 7
End If
End Sub

Sub Sekcja2:
If Value = 254 Then
Value = 8
End If
If Value = 253 Then
Value = 9
End If
If Value = 251 Then
Value = 10
End If
If Value = 247 Then
Value = 11
End If
If Value = 239 Then
Value = 12
End If
If Value = 223 Then
Value = 13
End If
If Value = 191 Then
Value = 14
End If
If Value = 127 Then
Value = 15
End If
End Sub
    
```



**Rys.5**

REKLAMA · REKLAMA · REKLAMA · REKLAMA · REKLAMA · REKLAMA · REKLAMA