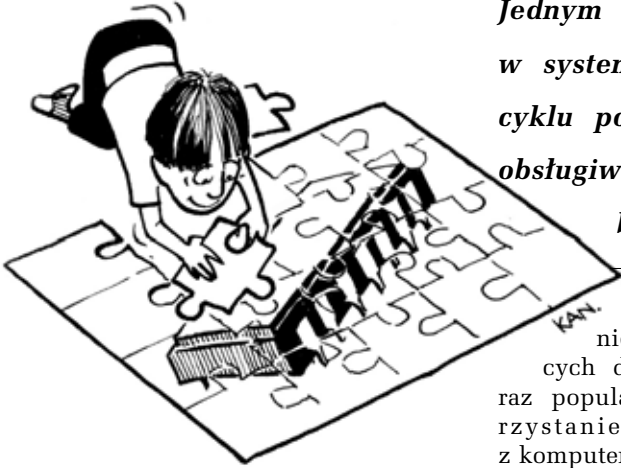


# Podstawy projektowania systemów mikroprocesorowych, część 8



*Jednym z podstawowych elementów interfejsu użytkownika w systemach cyfrowych jest klawiatura. W tej części cyklu pokazujemy, w jaki sposób dołączyć i jak obsługiwać lokalne klawiatury o stykach dołączanych bezpośrednio do portów mikrokontrolera.*

## Obsługa klawiatury

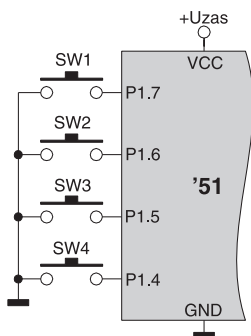
Chociaż klawiatura bez wątpienia należy do grupy omówionych w poprzednim odcinku elementów stykowych, to jednak sposób obsługi dużej liczby przycisków jest zagadnieniem nieco bardziej skomplikowanym (choć przy niewielkiej liczbie przycisków sposób obsługi nie różni się od obsługi pojedynczych elementów stykowych). Klawiatura jest jednym z najpopularniejszych urządzeń wejściowych umożliwiających komunikację użytkownika z systemem mikroprocesorowym i wpływające na sposób jego działania. W zależności od funkcji pełnionej przez dany system klawiatura może osiągać rozmiary układu kilkunastu czy kilkudziesięciu klawiszy. W zależności od liczby klawiszy zmienia się także sposób obsługi programowej, poczynając od bezpośredniego odczytu stanu portu, a kończąc na rozbudowanych procedurach obsługi klawia-

tur matrycowych. Obecnie w układach potrzebujących dużej liczby klawiszy coraz popularniejsze staje się wykorzystanie gotowych klawiatur z komputerów PC.

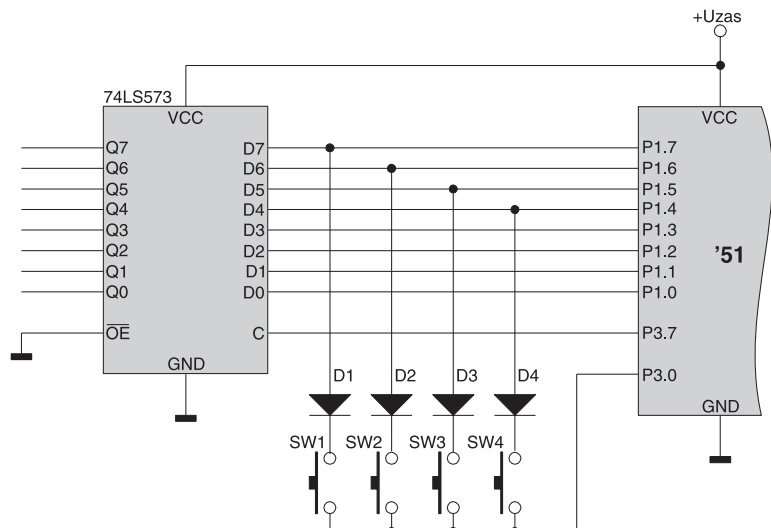
Jeżeli w projektowanym systemie nie jest potrzebna duża liczba klawiszy, wykorzystuje się bezpośrednie połączenie mikrowyłączników tworzących klawiaturę do linii portu mikrokontrolera. Sytuację taką przedstawiono na **rys. 27**. Połączenie wykonano bezpośrednio, bez żadnych dodatkowych rezystorów podciągających (oczywiście, o ile wykorzystywane linie posiadają rezystory wewnętrzne), gdyż umieszczona zazwyczaj w pobliżu mikrokontrolera klawiatura nie jest narażona na zakłócenia oraz nie ma konieczności stosowania prądów gwarantujących samooczyszczanie styków (następuje ono w sposób mechaniczny, wskutek działania znacznej siły nacisku palca). Prostota układowa tego rozwiązania

wpływa również na prostotę oprogramowania. Detekcja naciśnięcia przycisku odbywa się na zasadzie odczytu stanu konkretnej linii portu - „0” oznacza klawisz wciśnięty. Uwzględniając zjawiska łączeniowe (drżania styków) obsługa programowa może być identyczna jak omówiona wcześniej dla innych elementów stykowych. Należy jedynie zwielfokrotnić liczbę pomocniczych zmiennych odpowiadających liczbie stosowanych klawiszy. Wadą prezentowanego układu jest konieczność zarezerwowania jednej linii portu mikrokontrolera dla jednego klawisza i związane z tym trudności wygospodarowania linii sygnałowych przy większej liczbie klawiszy i innych urządzeń korzystających z portów mikrokontrolera.

Na **rys. 28** przedstawiono nieco bardziej ekonomiczne rozwiązanie. Linie jednego portu są wykorzystywane przez dwa urządzenia - w tym przypadku jednym urzą-



Rys. 27



Rys. 28

## List. 10. Procedura obsługująca prostą klawiaturę pokazaną na rys. 28

```

;KLAWISZE - zmienna bajtowa przechowująca stan klawiszy (1 - naciśnięty)
;BUFOR - zmienna bajtowa przechowująca daną do wysłania do zatrasku

;sekwencje napisano jako podprogramy wywoływane rozkazem LCALL

KLAWIATURA:
MOV P1,#0FFH ;ustawienie linii portu P1
CLR P3.0 ;zerowanie linii P3.0

MOV A,P1 ;odczyt stanu linii
CPL A ;negacja bitów (aby uzyskać 1 dla naciśniętego klawisza)

MOV KLAWISZE,A ;zapamiętanie stanu klawiatury

SETB P3.0 ;ustawienie linii P3.0 (aby naciskanie klawiszy nie
;wpływało na stan linii portu P1

RET ;powrót do programu głównego

ZATRASK:
MOV P1,BUFOR ;przesłanie danych do portu P1

SETB P3.7 ;generacja sygnału strobowego
CLR P3.7 ;(należy pamiętać o wyzerowaniu linii P3.7 po resecie
;mikrokontrolera - w części programu inicjującej
;zmienne i rejestry)

RET ;powrót do programu głównego

```

dzeniem jest klawiatura, natomiast drugim układem zatraskowy (*latch*) 74LS573 (może to być także dowolny inny układ posiadający wejście zezwalające/strobujące i pozostający obojętny na stan linii, gdy wejście to jest nieaktywne). Przedstawiony schemat uwzględnia jedynie klawiaturę czteroprzyciskową (dla zachowania czytelności rysunku), ale nic nie stoi na przeszkodzie, aby podłączona klawiatura wykorzystywała wszystkie osiem linii portu.

Układ ten działa następująco: jeżeli chcemy przesyłać dane do rejestru zatraskowego, wówczas konieczne jest ustawienie linii P3.0 w stan wysoki. Następnie wysyłamy do portu P1 bajt przeznaczony do przesłania, a potem generujemy sygnał strobowy (impuls): ustawiamy i zerujemy linię P3.7 - zbrocze opadające powoduje zatrzaśnięcie informacji w rejestrze i pamiętanie jej gdy P3.7=0. Gdy chcemy obsłużyć klawiaturę, należy wpisać do portu P1 wartość 0xFF (ustawienie wszystkich linii w stan wysoki w celu wykorzystania ich jako wejścia), a następnie wyzerować linię P3.0. Jeżeli któryś z klawiszy zostanie wciśnięty, to odpowiednia linia portu P1 zostanie wprowadzona w stan niski (wskutek połączenia z linią P3.0 pozostającą w stanie niskim). Odczytując stan portu P1 odczytujemy zatem stany klawiszy („0“ - klawisz wciśnięty) - po dokonaniu odczytu należy z powrotem ustawić linię P3.0 w stan wysoki.

Układ działałby również poprawnie, gdyby pominięto zastosowane diody włączone szeregowo z przyciskami, jednak mogłyby wystąpić przekłamania przy zapisie do rejestru, gdyby w czasie transmisji kilka przycisków było naciśniętych równocześnie. Dlatego też lepiej wspomniane diody zastosować. Przykładową sekwencję rozkazów obsługujących wspomniany układ zamieszczono na list. 10.

Na rys. 29 przedstawiono sposób dołączenia klawiatury do mikrokontrolera z wykorzystaniem wspólnych linii z innym urządzeniem, którym jest... również klawiatura! Przedstawiony na rysunku układ podłączenia przycisków tworzy klawiaturę matrycową. Charakterystyczną cechą takiego rozwiązania jest możliwość wyróżnienia na schemacie wierszy i kolumn, na przecięciu których znajdują się elementy zwierające (przyciski). Możliwa do obsłużenia liczba klawiszy jest zależna od liczby wierszy i kolumn matrycy, i jest równa iloczynowi tych dwóch wartości.

Zasada działania układu jest następująca: odczytywanie stanu klawiszy odbywa się w sposób grupowy, tzn. jednocześnie odczytywany jest stan jednego rzędu klawiszy (jeden wiersz lub jedna kolumna) - założmy, że w naszym układzie będziemy klawiaturę sprawdzać odczytując stan klawiszy pogrupowanych w wiersze (zgodnie z kolejną numeracją na schemacie). W takiej konfiguracji linie P1.4 do P1.7 będą pracowały jako wejście, natomiast linie P1.0 do P1.3 jako wyjście (nie jest konieczne stosowanie rezysto-

## List. 11. Program obsługi klawiatury matrycowej (schemat na rys. 29)

```

;KLAW1_8 - pomocnicza zmienna bajtowa pamiętająca stan klawiszy SW1
; do SW8 (najmłodszy bit oznacza stan SW8, "1" oznacza
; klawisz wciśnięty)
;KLAW8_16 - pomocnicza zmienna bajtowa pamiętająca stan klawiszy SW8
; do SW16 (najmłodszy bit oznacza stan SW16, "1" oznacza
; klawisz wciśnięty)

CZYTAJ_KLAWISZE: ;procedura wywoływana przez LCALL
MOV P1,#0FFH ;ustawienie linii portu P1 w stan wysoki

CLR P1.3 ;wyzerowanie linii P1.3
MOV A,P1 ;odczyt stanu linii portu P1 (czytamy 1 wiersz)
SETB P1.3 ;ustawienie linii P1.3 (bo stan już odczytaliśmy)
CPL A ;negacja (aby 1 oznaczała wciśnięty klawisz)
ANL A,#0F0H ;maskowanie młodszych 4 bitów - interesują nas starsze
MOV KLAW1_8,A ;zapamiętanie stanu 4 klawiszy

CLR P1.2 ;wyzerowanie linii P1.2
MOV A,P1 ;odczyt stanu linii portu P1 (czytamy 2 wiersz)
SETB P1.2 ;ustawienie linii P1.2
CPL A ;negacja
ANL A,#0F0H ;maskowanie
SWAP A ;zamienienie miejscami połówek bajtu (bo wiersz 2 to 4
; młodsze bity KLAW1_8)
ORL A,KLAW1_8 ;połączenie informacji o obu wierszach
MOV KLAW1_8,A ;zapamiętanie stanu wszystkich ośmiu klawiszy

CLR P1.1 ;wyzerowanie linii P1.1
MOV A,P1 ;odczyt stanu linii portu P1 (czytamy 3 wiersz)
SETB P1.1 ;ustawienie linii P1.1
CPL A ;negacja
ANL A,#0F0H ;maskowanie
MOV KLAW8_16,A ;zapamiętanie stanu 4 klawiszy (SW9-SW12)

CLR P1.0 ;wyzerowanie linii P1.0
MOV A,P1 ;odczyt stanu linii portu P1 (czytamy 4 wiersz)
SETB P1.0 ;ustawienie linii P1.0
CPL A ;negacja
ANL A,#0F0H ;maskowanie
SWAP A ;zamienienie miejscami połówek bajtu
ORL A,KLAW8_16 ;połączenie informacji o obu wierszach
MOV KLAW8_16,A ;zapamiętanie stanu wszystkich ośmiu klawiszy

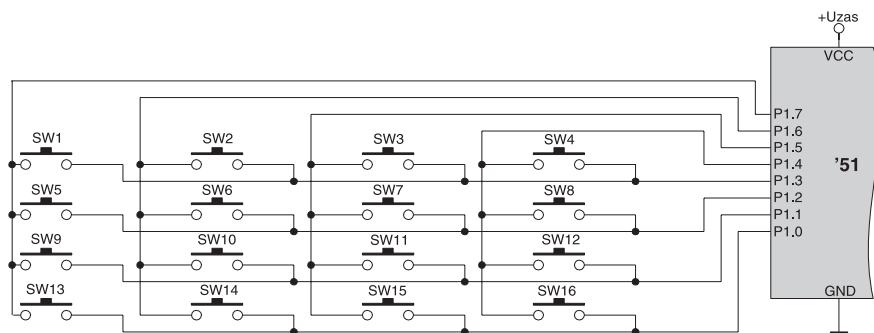
RET ;powrót do programu głównego

```

rów podciągających dla linii P1.0 i P1.1 mikrokontrolerów AT89Cx051, gdyż stan wysokiej impedancji i stan wysoki są tu nierozróżnialne - na pracę układu wpływa tylko stan niski).

Wstępnie należy do portu P1 wpisać same jedynki (ustawić wszystkie linie), następnie możemy przejść do odczytu pierwszego wiersza. Dokonujemy tego zerując programowo linię P1.3 (linie P1.0 do P1.2 pozostają w stanie wysokim) i odczytując stan wyprowadzeń P1.4...P1.7 - stan tych linii odzwierciedla stan klawiszy w pierwszym wierszu (SW1...SW4) - przykładowo stan niski na linii P1.6 oznacza wciśnięcie SW2. W podobny sposób postępujemy dla pozostałych wierszy, kolejno zerując pojedynczo linie P1.2, P1.1 i P1.0 (pozostałe linie w stanie wysokim) i odczytując stan czterech starszych linii portu. Rozwiązanie programowe będzie polegać na jednorazowym odczytaniu stanu wszystkich klawiszy, a następnie przekazaniu informacji o naciśnięciach do programu głównego (przez zapamiętanie w pomocniczej zmiennej). Przykładową procedurę obsługi klawiatury matrycowej zamieszczono na **list. 11**.

Gdyby zaistniała potrzeba odczytywania stanu klawiatury kolumnami, to zmieniają się jedynie funkcje wyprowadzeń: kolejno zerować będziemy wyprowadzenia P1.4...P1.7, a odczytywać P1.0...P1.3 (tym razem konieczne będą rezystory podciągające dla P1.0 i P1.1 mikrokontrolerów AT89Cx051).



Rys. 29

W przypadku naciśnięcia więcej niż jednego klawisza istnieje niebezpieczeństwo odczytania przez układ dziwnej kombinacji naciśniętych klawiszy (podobna cecha jak w układzie z rys. 28, choć tutaj nie zostaną przekłamanie żadne transmitowane dane - tylko klawiatura korzysta z tych linii portu). Aby temu zapobiec można zastosować diodę szeregowo włączoną z każdym klawiszem (patrz rys. 28). W naszym przypadku (odczytywane wiersze klawiatury) diody należy włączyć katodami w kierunku linii wierszy (P1.0...P1.3). Gdybyśmy klawiaturę chcieli odczytywać kolumnami, to należy diody podłączyć odwrotnie.

Zastosowanie klawiatury matrycowej w systemie mikroprocesorowym pozwala na zaspokojenie nawet bardzo rozbudowanych potrzeb dotyczących klawiatury. Istnieją jednak aplikacje, w których zastosowanie rozbudowanego układu klawiatury jest niemożliwe ze względu na wykorzystanie w innych celach linii portów mikrokontrolera - po prostu brakuje nam wolnych linii niezbędnych do obsługi dużej

liczby klawiszy. Rozwiązania problemu są dwa: można zastosować specjalizowany układ scalony lub lepiej odpowiednio oprogramowany drugi mikrokontroler (rozwiązanie niejednokrotnie tańsze i elastyczniejsze) odpowiedzialny tylko i wyłącznie za wykrywanie faktu naciśnięcia klawiszy i komunikujący się z głównym mikrokontrolerem za pomocą niewielkiej liczby linii (jedna, góra dwie linie) lub zastosować to samo rozwiązanie w gotowej postaci dostarczanej przez wielu producentów sprzętu komputerowego - standardową klawiaturę komputera PC-AT. Zastosowanie klawiatury PC ma tę zaletę, że jest stosunkowo tanie, a także bardzo łatwe w adaptacji i estetyczne - odpada konieczność borykania się z rozwiązaniami mechanicznymi i obudową klawiatury. Jedyną wadą takiego rozwiązania jest konieczność zapoznania się z dość nietypowym sposobem komunikacji klawiatury z mikroprocesorem. Szczegóły przedstawimy za miesiąc.

**Paweł Hadam, AVT**  
**pawel.hadam@ep.com.pl**