

Dział „Projekty Czytelników” zawiera opisy projektów nadesłanych do redakcji EP przez Czytelników. Redakcja nie bierze odpowiedzialności za prawidłowe działanie opisywanych układów, gdyż nie testujemy ich laboratoryjnie, chociaż sprawdzamy poprawność konstrukcji.

Prosimy o nadsyłanie własnych projektów z modelami (do zwrotu). Do artykułu należy dołączyć podpisane **oświadczenie**, że artykuł jest własnym opracowaniem autora i nie był dotychczas nigdzie publikowany. Honorarium za publikację w tym dziale wynosi 250,- zł (brutto) za 1 stronę w EP. Przesyłanych tekstów nie zwracamy. Redakcja zastrzega sobie prawo do dokonywania skrótów.

## Przycisk ANY do PC, część 2

Projekt  
103

W drugiej części artykułu przedstawiono tajniki oprogramowania sterującego pracą mikrokontrolera, który emuluje sterownik klawiatury PC. Projekt może być interesujący dla projektantów urządzeń elektronicznych współpracujących z PC poprzez szeregowo łącze klawiatury. W artykule zawarto wszystkie informacje niezbędne dla poznania mechanizmów wymiany danych między PC i dołączoną klawiaturą.



Podczas włączania jakiegось urządzenia między klawiaturę a PC musimy się liczyć z tym, że w interfejsie nie została przewidziana możliwość odłączenia klawiatury w czasie pracy systemu. Dlatego trzeba uważać, aby żadne z komunikujących się urządzeń nie odczuło zaistnienia tego zdarzenia. Gdyby tak się stało, może to spowodować zawieszenie pracy klawiatury, a wtedy pozostaje już tylko wyłączenie komputera.

Aby temu zapobiec, opracowałem dość skomplikowany program. Pomyślmy, kiedy klawiatura może się zawiesić? Ja wymyśliłem następujące sytuacje:

1. Podczas wymiany informacji pomiędzy PC-tem i klawiaturą układ wtrąci się „w pół zdania“.

2. Układ połączy klawiaturę z komputerem w momencie, gdy ta zaczęła już wcześniej coś wysyłać.

3. Komputer zechce wysłać jakąś komendę do klawiatury w czasie, gdy będzie ona odłączona.

Sytuacji pierwszej i drugiej zaradziłem w prosty sposób. Linia P3.2 procesora, będąca jednocześnie wejściem przerwania *Int0*, jest na stałe

połączona z linią zegara klawiatury. Układ łączy i rozłącza klawiaturę tylko wtedy, gdy na linii zegara nie ma sygnału przez przynajmniej 100 ms. W programie zostało to spełnione dzięki przerwaniom *Int0* oraz *Timer0*. Ich działanie jest banalne. Odpowiednie procedury przedstawiono na **list. 1**.

Zmienna *Bit\_linia\_zajeta*, która jest odpowiednio ustawiana i zerowana, jest następnie wykorzystywana w podprogramach *Odlacz\_klawiature()* i *Dolacz\_klawiature()*. Robią one dokładnie to, na co nazwa wskazuje, a ich działanie przedstawiono na **list. 2**.

Pomimo wielu prób nie udało mi się rozwiązać problemu występującego w sytuacji trzeciej. Jednak podczas eksperymentów z gotowym układem nie zdarzyło się, aby komputer wysłał jakąś komendę przy odłączonej klawiaturze.

Testy wykazują, że w pewnych sytuacjach, zaraz po wysłaniu do komputera kodów *ALT+CTRL+DELETE*, linia zegara jest przez komputer zwierana aż do restartu. Ponieważ program czeka z transmisją na jej zwolnienie, może to spowodować, że klawiatura zostanie dołączona

na do komputera już po przeprowadzeniu przez niego testów tego podzespołu. Spowoduje to błąd klawiatury. W sumie problem ten nie jest bardzo dotkliwy. Błędne zadziałanie układu udało mi się wykryć tylko na pocziwym 386 podczas gry „Prince of Persia“ v1.0. Jednak klawiatura działa! Jest to ciekawy przypadek, kiedy przy komunikacie „Keyboard error press F1 to continue“ przycisk F1 zdaje egzamin!

Udało mi się trochę złagodzić ten problem w następujący sposób: *Timer1* odlicza czas przez jaki linia zegara znajduje się na poziomie niskim. Jeśli nastąpi jego przepełnienie, procedura *On\_timer1* ustawi zmienną *Bit\_time\_up*. Pierwotnie program w tym momencie przerwiał natychmiast transmisję, jednak okazało się, że powodowało to błąd interfejsu klawiatury, a przez to permanentne zawieszanie komputera. Ostatecznie, jeśli zmienna ta jest ustawiona, program kończy wysyłanie aktualnego bajtu i zaraz potem łączy klawiaturę.

Najważniejszym blokiem programu jest procedura *Wyslij\_do\_pc*. Wysyła ona do komputera daną z akumulatora. Działanie każdego z podprogramów, mających przesłać kody przycisków, polega na odłączeniu klawiatury, a potem kilkukrotnym wywołaniu właśnie tej procedury z odpowiednim parametrem. Dla przykładu, na **list. 3** zamieszczono procedurę wysyłania kodu przycisku ANY.

*Bit\_error* jest ustawiany podczas wysyłania danej, jeśli nastąpiło przepełnienie *Timer1* (patrz wyżej). Przy

### List. 1

```
On_int0:
  Set Bit_linia_zajeta
  Counter0 = Const_opoznienie_zajetosci
  Start Timer0
  Return

On_timer0:
  Stop Timer0
  Reset Bit_linia_zajeta
  Return
```

### List. 2

```
*****
' Odłącza klawiaturę
Sub Odlacz_klawiature()
'Oczekiwanie na zwolnienie linii
jb Bit_linia_zajeta, *+0
'Rozłączenie klawiatury od PC-ta
Reset Clk_con
Reset Dat_con
End Sub

*****
' Dołącza klawiaturę
Sub Dolacz_klawiature()
'Oczekiwanie na zwolnienie linii
jb Bit_linia_zajeta, *+0
Set Dat_con
Set Clk_con
End Sub
```

```

List. 3
Wysylaj_any:
Reset Led_busy
If Zw1 = 0!Then
  B_any = Rnd(48)
  B_any = Lookup(b_any,Data_kody)
  Odłącz_klawiature
  Reset Bit_error
  Wyslij_do_pc B_any
  jb Bit_error, .error_any
  Waitms 5
  Wyslij_do_pc &HFO
  jb Bit_error, .error_any
  Wyslij_do_pc B_any
Error_any:
  Dolącz_klawiature
Else
  Waitms 6
End If
Set Led_busy
    
```

okazji należy zauważyć, że kod danego przycisku jest pobierany z odpowiedniej tablicy. Możliwych kodów jest 48, a przyciski, które mogą być „ANY“, zostały na rys. 2 (EP9/2002) oznaczone na szarą.

Zrozumienie kodu źródłowego (dostępny na stronie WWW EP) nie powinno być trudne, jeżeli wie się, że działanie programu jest zależne od zmiennych, które są przedstawiane w odpowiednich przerwanach. Zachęcam do „pomajstrowania“ przy kodzie.

**Montaż i uruchomienie**

Na rys. 6 został przedstawiony schemat montażowy, a wzór mozaiki ścieżek opublikowaliśmy w EP9/2002. Płytką została przygotowana jako jednostronna, dlatego nie udało mi się uniknąć dwóch zworek (oprócz zworek konfiguracji).

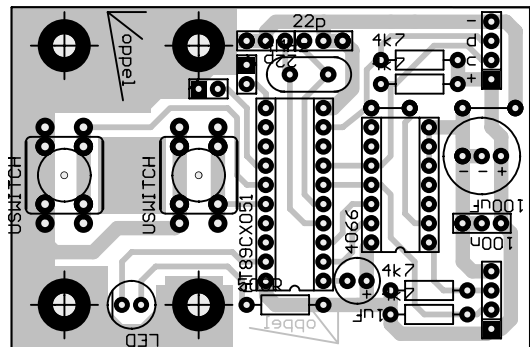
Zanim rozpoczniemy montaż warto przygotować sobie obudowę. Jej zastosowanie jest wskazane ze względu na ryzyko zwarcia zasilania podawanego przecięć bezpośrednio z płyty

głównej naszego komputera. Układ (płytką) nie jest przewidziany do żadnej konkretnej obudowy. Model został umieszczony w obudowie typu KM33B.

Przy projektowaniu płytki szczególnie nacisk położyłem na wzmocnienie części, na której znajdują się przyciski. W związku z tym śruby są rozstawione w nietypowych miejscach.

Przy przygotowywaniu obudowy można się posłużyć płytką jako szablonem do wiercenia otworów. Trzeba na tym etapie wiedzieć, że przyciski będą montowane od strony druku! Płytkę przykładamy od wewnętrznej strony obudowy drukiem do dołu. Otwory pod śruby możemy od razu wierceć wiertłem 3 mm. Miejsca pod przyciski należy najpierw zaznaczyć za pomocą cieniutkiego wiertelka 0,8 mm. Obok diody LED nie został co prawda umieszczony żaden punkt „pilotażowy“, lecz otwór na ten element nie musi być umieszczony precyzyjnie.

Odkładamy teraz płytkę na bok i wykańczamy obudowę. Najpierw rozwiercamy otwory pod przyciski. Ich średnica będzie zależna od wybranych przez nas elementów. Mogą to być zarówno standardowe mikroprzełączniki, jak również ich duże odpowiedniki. Myślę, że dobrym pomysłem jest umieszczenie dużego przycisku ANY i małego ALT+CTRL+DELETE. Ale to tylko moja sugestia. Otwory pod śruby dobrze jest rozwiercić od strony zewnętrznej grubym wiertłem do pewnej głębokości tak, aby łebki śrub nie wystawały. Nie polecam tutaj używania wiertarki, bo



Rys. 6

wiertło „przejdzie“ na wylot. Dużo bezpieczniej jest obracać je palcami.

Teraz możemy już wrócić do płytki. Montaż musimy rozpocząć niestety od dwóch zworek zaznaczonych na płytce kreskami. Dalej montujemy wszystkie elementy oprócz przycisków i diody. W miejsce złączy PC i KBD stosujemy goldpiny. Rezystory R1...R4 nie są konieczne. Egzemplarz modelowy dobrze spisywał się bez nich. Jednak gdyby występowały błędy w działaniu układu, można wlutować w to miejsce rezystory o wartości około 10 kΩ.

Przyciski i diody montujemy od strony druku. Najpierw „na sucho“ układamy przyciski i sprawdzamy, czy dobrze pasują do obudowy. W razie konieczności na tym etapie możemy wlutować je pod lekkim skosem. Następnie, już po ich przylutowaniu, wpasowujemy diodę odpowiednio przycinając jej wyprowadzenia. Lutujemy tylko jej jedną końcówkę. Dzięki temu przed ostatnim lutem możemy ustawić ją w odpowiedniej pozycji.

Przed zamknięciem obudowy pozostaje nam wykona-

nie gniazdka i wtyczki mini DIN lub dużej DIN zakończonych listwami nakładanymi na goldpiny oraz wybranie odpowiadającej nam konfiguracji za pomocą zworek. Złącze PC zostało odpowiednio opisane na płytce. Złącze KBD posiada identyczne rozmieszczenie wyprowadzeń jak PC.

Uwaga! Przed pierwszym podłączeniem układu do komputera zalecam wypróbowanie najpierw jego działania. W tym celu do odpowiednich końcówek portu PC lub KBD należy podłączyć stabilizowane źródło napięcia o wartości 5V. Jeżeli dioda mignie jednokrotnie i z żadnego miejsca nie wydobywa się dym, możemy uznać układ za sprawny.

**Uwagi końcowe**

Po podłączeniu układu i włączeniu komputera zauważymy poprawę komfortu obsługi. Nie powoduje już zmieszania prośba o naciśnięcie przycisku ANY. Zawieszenie systemu też już tak nie irytuje - po co się złościć, skoro problem można rozwiązać jednym przyciskiem.

**Radosław Koppel**