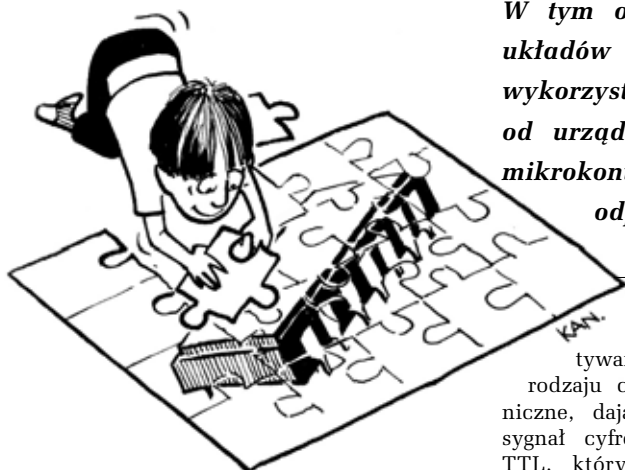


Podstawy projektowania systemów mikroprocesorowych, część 7



Podłączanie czujników do mikrokontrolera

W zależności od funkcji, jaką ma spełniać projektowany system, niejednokrotnie zachodzi potrzeba wykorzystania w nim różnego rodzaju czujników. Ze względu na specyfikę działania układów cyfrowych, bez poważnych komplikacji układowych (np. w postaci przetworników analogowo/cyfrowych) można jedynie wykorzystać czujniki dające na wyjściu sygnał cyfrowy. Mimo takiego ograniczenia, w wielu zastosowaniach takie proste rozwiązania w zupełności wystarczają.

W praktyce najczęściej korzysta się z różnego rodzaju czujników stykowych. Zasada ich działania opiera się na mechanicznym zwarciu styków łącznika i zamknięciu w ten sposób obwodu elektrycznego. Do tej grupy czujników zaliczamy różnego rodzaju przyciski i przełączniki, wyłączniki krańcowe, czujniki kontaktronowe i styki przekaźników. W praktyce spotyka się dwie wersje takich elementów: normalnie rozwarte - wymuszenie działające na czujnik powoduje zwarcie jego styków oraz normalnie zwarte - wymuszenie powoduje rozwarcie styków, które przez pozostały czas są zwarte.

W tym odcinku Kursu rozpoczniemy omawianie praktycznych układów sterujących konkretnymi urządzeniami wykorzystywanymi w systemach mikroprocesorowych. Zaczniemy od urządzeń wejściowych, za pośrednictwem których do mikrokontrolera docierają sygnały z zewnątrz i będąc odpowiednio interpretowane, wpływają na sposób wykonywania programu.

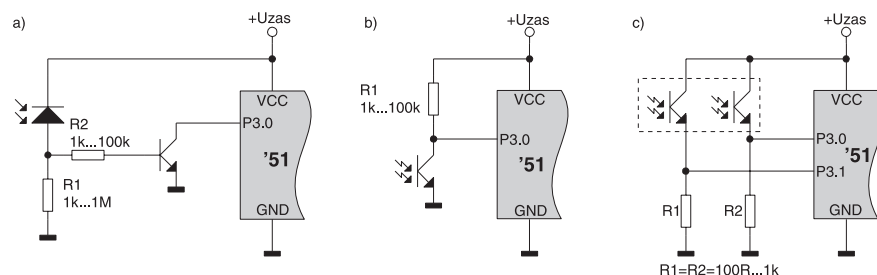
Równie często wykorzystywanymi elementami są różnego rodzaju czujniki i podzespoły elektroniczne, dające na wyjściu bezpośredni sygnał cyfrowy zgodny ze standardem TTL, których obsługa sprzętowa nie stwarza żadnego problemu (bezpośrednie połączenie z liniami portów). Często stosuje się także fotodetektory wykorzystywane jako czujniki światła (jasno/ciemno) lub - w połączeniu z jakimś źródłem światła - mogą one tworzyć fotokomórki zliczające przedmioty, wykrywające kierunek ruchu itp.

Zwracając uwagę na zalety czujników optoelektronicznych, należy powiedzieć, że są to elementy bardzo szybkie i praktycznie nieużywające się (w odróżnieniu np. od czujników stykowych). Zapewnienie odpowiedniego źródła światła (najczęściej w postaci podczerwonej diody LED) pozwala na wykorzystanie fotodetektorów w wielu aplikacjach, w których z powodzeniem zastępują mniej trwałe i bardziej zawodne elementy stykowe. Jako że czujniki optoelektroniczne stanowią „krok naprzód” w stosunku do elementów stykowych, to właśnie rozwiązania układowe wykorzystujące fotodetektory jako źródła sygnału zostaną omówione jako pierwsze.

Na rys. 24 przedstawiono sposoby dołączenia do mikrokontrolera popularnych fotodetektorów: fotodiody i fototranzystora. Na rys. 24a pokazano sposób podłączenia fotodiody. Ze względu na niewielki prąd płynący przez nią w kierunku zaporowym, sygnał generowany przez fotodiody musi zostać

wzmocniony przez tranzystor. Jeżeli na fotodiody znacznie padać światło, to spowoduje to przepływ prądu i odłożenie się napięcia na rezystorze R1. Napięcie to poprzez ograniczający prąd rezystor R2 podawane jest na bazę tranzystora i jeżeli wystarcza ono, aby tranzystor zaczął przewodzić, to wyprowadzenie mikrokontrolera zostaje zwarte do masy - na linii portu pojawia się stan niski. Czułość tego układu można regulować w dość szerokich granicach poprzez dobór rezystora R1. Zwiększając wartość tego rezystora, zwiększamy równocześnie czułość układu. Nie należy jednak przesadzać ze zbyt dużym zwiększaniem rezystancji, gdyż może zaistnieć sytuacja, że układ będzie cały czas aktywny mimo całkowitej ciemności (wskutek prądu upływu diody). Dużą rolę odgrywa tutaj także wzmocnienie zastosowanego tranzystora - im większe wzmocnienie, tym oczywiście większa czułość. Jeżeli nie można w ten sposób osiągnąć pożądanej czułości (przy detekcji bardzo niewielkich natężeń oświetlenia), można zamiast fotodiody zastosować fototranzystor (podłączając kolektor w miejsce katody, a emiter w miejsce anody).

Na rys. 24b przedstawiono układ z fototranzystorem będącym jednocześnie elementem detekcyjnym, jak i sterującym linię portu mikrokontrolera. Czułość tego układu może być regulowana za pomocą rezystora R1 (duża rezystancja - duża czułość), maksymalną czułość można uzyskać nie montując rezystora R1 (o ile wykorzystywana linia portu posiada wewnętrzny rezystor podciągający - dla linii P3.0 wykorzystanej na zamieszczonym schemacie warunek ten jest spełniony). W zależności od typu zastosowanego fototranzystora można osiągnąć czułości lepsze niż w układzie z fotodiody i tranzystorem. Pod względem logicznym układ działa tak samo jak poprzedni - oświetlenie fototranzystora powoduje pojawienie się stanu niskiego na wyprowadzeniu P3.0.



Rys. 24

List. 8.

```

;KIERUNEK - etykieta zmiennej bitowej, określającej kierunek: 1-lewo, 0-prawo
;OBIEKT - etykieta zmiennej bitowej określająca, czy obiekt znajduje się na
czujniku
;          1 - jest na czujniku

;(program główny - blok inicjowania wartości zmiennych)
INICJUJ:
CLR KIERUNEK      ;zerowanie
CLR OBIEKT       ;zmiennych
...
...

;(podprogram wyznaczający kierunek ruchu, wywołanie przez LCALL KIERUNEK)
KIERUNEK:
MOV C,P3.0        ;oba fototranzystory oświetlone
ANL C,P3.1        ;(stan 1 na obu liniach),
JC BRAK_OBIEKTU  ;czyli brak obiektu na czujniku

JB OBIEKT,CZEKAJ_NA_ZEJSCIE ;jeżeli obiekt na czujniku, to czekaj, aż
;opuści czujnik
;jeżeli nie było obiektu na czujniku
;to znaczy, że zarejestrowano nowy obiekt
MOV C,P3.1        ;stan 1 na P3.1 oznacza, że jedzie w lewo
MOV KIERUNEK,C   ;wiec kopiujemy P3.1 do KIERUNEK
SETB OBIEKT      ;i ustawiamy zmienną OBIEKT, aby
;poczekać na zjechanie z czujnika
RET              ;czekamy na stan wysoki na P3.0 i P3.1)

BRAK_OBIEKTU:
CLR OBIEKT

CZEKAJ_NA_ZEJSCIE:
RET

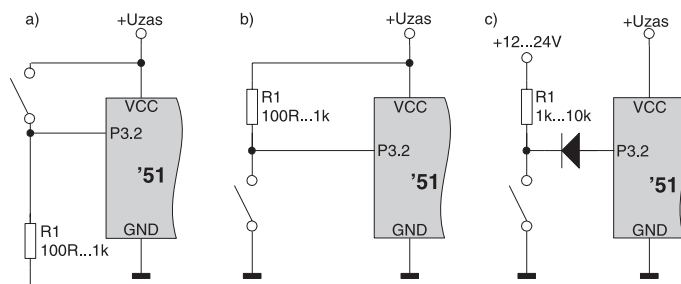
```

Na rys. 14c przedstawiono typowy układ wykorzystywany do wykrywania kierunku poruszania się obiektów. Wykorzystuje on podwójny fototranzystor umieszczony w trójkątówkowej obudowie (połączone kolektory). Oświetlenie któregoś z fototranzystorów powoduje przepływ prądu poprzez odpowiedni rezystor i wzrost napięcia na wyprowadzeniu portu - oświetleniu fototranzystora odpowiada stan wysoki linii. Jeśli fototranzystory nie przewodzą, stan na liniach portu jest interpretowany jako niski, gdyż rezystory R1 i R2 mają rezystancję kilkadziesiąt razy mniejszą niż wbudowane rezystory podciągające. Jeśli wymagane byłoby znaczne zwiększenie czułości układu, należy tutaj zastosować tranzystory pośredniczące tak jak pokazano na układzie z rys. 14a - układ taki wprowadzi negację konieczną do uwzględnienia przy pisaniu programu (oświetlenie fototranzystora wywoła stan niski na linii portu). Zwiększanie wartości rezystancji R1 i R2 zwiększy nieco czułość, jednak łatwo jest osiągnąć stan, w którym nie będzie można rozpoznać na liniach portu faktu zatkania któregoś z fototranzystorów - wartość rezystancji R1 i R2 stanie się porównywalna z wartością rezystancji rezystorów podciągających i nie będzie możliwe uzyskanie na wyprowadzeniach napięcia odpowiadającego stanowi niskiemu. Należy również pamiętać, aby nigdy programowo nie zerować wyprowadzeń portu, gdyż może to doprowadzić do uszkodzenia fototranzystorów lub mikrokontrolera. Rozwiązaniem problemu jest zastosowanie układu z tranzystorem pośredniczącym lub doprowadzenie sygnału do wyprowadzeń mikrokontrolera przez rezystory o wartości około 1 kΩ.

W takim układzie rozróżnianie kierunku ruchu jest możliwe przez sprawdzanie stanów obu linii i odpowiednią ich interpretację. Wykrycie faktu zasłonięcia jednego z fototranzystorów w sytuacji, gdy drugi pozostaje oświetlony, informuje nas o kierunku ruchu - np. stan niski na linii P3.0 i wysoki na P3.1 informuje o ruchu w lewo, a sytuacja gdy P3.1=0 i P3.0=1 świadczy o ruchu w prawo. Można to zrealizować za pomocą programu pokazanego na list. 8.

Przedstawiony podprogram wykrywa ruch obiektu i przechowuje informację o kierunku w zmiennej *KIERUNEK* do czasu pojawienia się następnego obiektu. Należy zwrócić uwagę na prędkość przemieszczających się przed czujnikiem obiektów i odpowiednie częste wywoływanie procedury, aby nie wystąpiło zjawisko „gubienia” przejść wskutek zbyt rzadkiego sprawdzania stanu czujnika.

Zwróćmy teraz uwagę na zasady postępowania przy projektowaniu urządzeń z wykorzystaniem elementów stykowych. Na rys. 25 przedstawiono kilka sposobów podłączenia elementów stykowych do portów mikrokontrolera. Na rys. 25a pokazano bardzo rzadko wykorzystywaną konfigurację połączeń



Rys. 25

elementu stykowego, czasami można spotkać się z opinią, iż jest to połączenie błędne. Układ jest jednak jak najbardziej prawidłowy, choć posiada pewne cechy, których zlekceważenie może doprowadzić do błędnego działania mikrokontrolera. Element stykowy jest tutaj włączony pomiędzy plus zasilania a wyprowadzenie portu. Jeśli styk jest rozarty, to na wyprowadzeniu portu będzie utrzymywał się stan niski wymuszony przez rezystor R1 o wartości niewielkiej w porównaniu z wewnętrznym rezystorem podciągającym mikrokontrolera. Zwarcie styku spowoduje podanie na wyprowadzenie portu pełnego napięcia zasilania, które oczywiście będzie zinterpretowane jako stan „1”. Wadą tego układu jest stosunkowo duży pobór prądu związany z koniecznością zapewnienia małej wartości rezystancji R1 - jednak mimo wszystko przy podanych wartościach rezystancji nie przekroczy on 50 mA. O wiele groźniejszy w skutkach może być błąd programisty polegający na wyzerowaniu programowym linii portu - jeżeli nastąpi to w momencie, gdy styk jest zwarty, to powstanie zwarcie napięcia zasilania do masy (przez linię portu) mogące doprowadzić do uszkodzenia mikrokontrolera (sytuacja podobna jak na rys. 24c) - mogąca być również w taki sam sposób rozwiązana).

Kolejny układ, będący chyba najbardziej klasycznym, przedstawiono na rys. 25b. Tutaj element stykowy zwierza wyprowadzenie portu do masy zasilania - zwarcie styku oznacza stan „0”, a rozwarcie stan „1”. Dodatkowy rezystor podciągający R1 nie jest konieczny (do poprawnej pracy wystarczy rezystor wbudowany w mikrokontroler), pozwala natomiast na ustawienie optymalnego prądu pracy dla elementu stykowego, gwarantującego jego samooczyszczanie (dotyczy zwłaszcza elementów o niewielkiej sile nacisku styków, np. kontaktronów lub małych przekaźników). Niekiedy zachodzi konieczność podania na element stykowy napięcia wyższego niż napięcie zasilające mikrokontroler (np. gdy styk jest elementem wyjściowym jakiegoś obwodu elektronicznego zasilanego z wyższego napięcia i nie może być od tego napięcia odseparowany). Z wiadomych nam powodów (patrz po-

przednie odcinki kursu) nie można takiego układu podłączyć bezpośrednio - stosuje się zatem podłączenie za pośrednictwem diody, jak w układzie z rys. 25c). Choć połączenie to było już omówione przy współpracy z elementami elektronicznymi, warto pamiętać, że może być wykorzystane również tutaj.

Często zachodzi potrzeba doprowadzenia sygnału do mikrokontrolera z elementu oddalonego o kilka...kilkanaście czy nawet kilkadziesiąt metrów. Należy wówczas zadbać, aby dostarczony do mikrokontrolera sygnał był odpowiedniej jakości i pozbawiony zakłóceń. Sytuację taką pokazano na rys. 26a (jest to rozwinięcie układu z rys. 25b). Pokazano tutaj, jak zabezpieczyć się przed zakłóceniami w przypadku stosowania długich przewodów połączeniowych. Zastosowane kondensatory C1 i C2 zwierają linię sygnałową dla przebiegów o wysokich częstotliwościach, natomiast niewielka rezystancja rezystora podciągającego R1 wymusza przepływ znacznego prądu w linii, co dodatkowo podnosi odporność na zakłócenia. Trzecim elementem zabezpieczającym jest obwód całkujący, przez który sygnał trafia na wyprowadzenie portu. Choć takie rozwiązanie przypomina układy stosowane w technice sygnałów analogowych, w przypadku sygnałów cyfrowych sprawuje się również bardzo dobrze. Dodatkowym elementem zabezpieczającym przed przepięciami w linii są diody D1 i D2 niepozwalające na przekroczenie napięcia linii poza przedział napięć zasilania. W tym układzie warto rozważyć jeszcze możliwość pojawienia się na wejściu linii krótkich impulsów wysokonapięciowych (zwłaszcza indukowanych od pobliskich wyładowań atmosferycznych, itp.). Dość skutecznie przed uszkodzeniem układu wskutek działania wysokiego napięcia można się zabezpieczyć przez włączenie równoległe do C2 warystora metalowotlenkowego (elementy takie są powszechnie stosowane w instalacjach telekomunikacyjnych i aparatach telefonicznych oraz w profesjonalnych sterownikach przemysłowych), który zabezpiecza przed przekroczeniem określonego napięcia na linii oraz jest ele-

mentem dużo szybszym i skuteczniejszym niż zastosowane diody. Wykorzystywany warystor powinien mieć nominalne napięcie pracy nieco większe od napięcia zasilania mikrokontrolera - zastosowanie elementu o zbyt wysokim napięciu pracy spowoduje sytuację, że warystor nie zadziała, mimo iż napięcie na wejściu układu przekroczy wartość bezpieczną dla mikrokontrolera.

Całkowite rozwiązanie problemu przepięć mogących uszkodzić układ sterownika mikroprocesorowego przedstawiono na rys. 26b. Wykorzystano tutaj oddzielne zasilanie obwodu czujnika wyższym napięciem (większa odporność na zakłócenia) wraz z separacją galwaniczną obwodów czujnika i linii sygnałowej od obwodów mikrokontrolera. Wykorzystanie transoptora zabezpiecza nas przed przepięciami indukowanymi w linii oraz przed błędami instalatora (podanie zbyt wysokiego napięcia na wejście układu co najwyżej uszkodzi tylko transoptor). Układ zachowuje logikę zgodną z układem z rys. 26a (zwarcie styku oznacza stan „0”). Rezystor R2 zmniejsza czułość obwodu detekcyjnego transoptora (konieczność silniejszego oświetlenia fototranzystora w celu osiągnięcia przepływu prądu wystarczającego do spadku napięcia do poziomu stanu „0”), a co za tym idzie zwiększa odporność na zakłócenia.

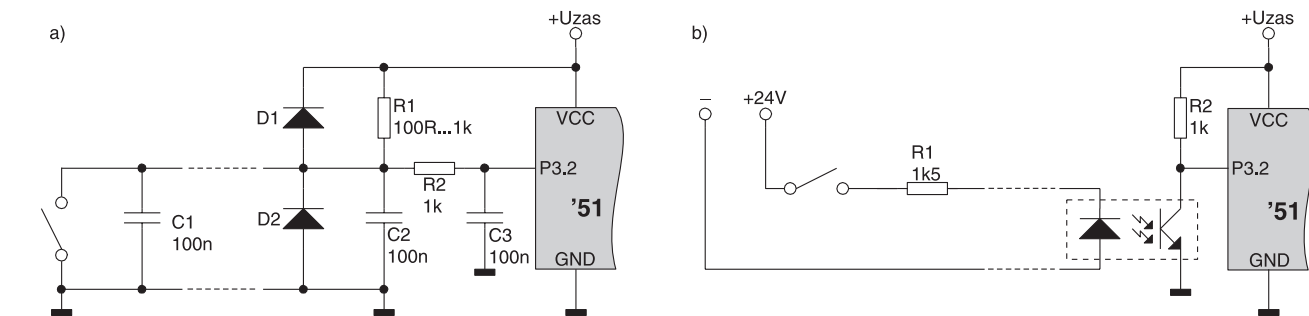
Uważny Czytelnik zapewne zauważy, że przy przedstawianiu układów z elementami stykowymi ani razu nie zwrócono uwagi na problem drgania styków, ani też na problem wolnego narastania napięcia wskutek stosowania kondensatorów filtrujących. Powodem, dla którego mówimy o tym dopiero teraz, jest fakt, że w systemach mikroprocesorowych problemy związane z tymi zjawiskami załatwia się na drodze programowej, dążąc do jak największej prostoty sprzętu.

Zarówno zjawisko drgania styków, jak i zjawisko powolnego narastania napięcia jest wstępnie eliminowane już wskutek sposobu stosowanego przez mikrokontroler do sprawdzania stanów na liniach portów. Stany te są sprawdzane w każdym cyklu maszynowym w określonym takcie zegara, natomiast

przez pozostały czas zmiana stanu na liniach portów nie wpływa w żaden sposób na stan mikrokontrolera. Pierwszym elementem filtrującym jest zatem częstotliwość cykli maszynowych (1/12 częstotliwości zegarowej w przypadku omawianych mikrokontrolerów rodziny 51), od której zależy eliminowanie stanów przejściowych krótszych niż jeden cykl maszynowy. Zazwyczaj przy współpracy z różnego rodzaju elementami stykowymi ten podstawowy stopień filtracji nie jest wystarczający - stosuje się wówczas opóźnienie programowe. Program, którego działanie zależy od stanu określonej linii, sprawdza w tym przypadku wartość pomocniczego bitu, natomiast faktyczny stan linii odczytuje procedura wywoływana co określony czas (np. wskutek wystąpienia przerwania z któregoś z liczników) zapisująca stan linii do tego pomocniczego bitu. Przykładowe rozwiązanie przedstawiono na list. 9.

Stosując powyższe rozwiązanie, należy pamiętać o odpowiednim zaprogramowaniu (włączeniu) układu przerwań oraz o dobraniu liczby cykli zliczanych przez licznik w celu uzyskania pożądanego opóźnienia.

Jak już wspomniano wcześniej, układy stykowe mogą pracować w dwóch konfiguracjach - jako styki normalnie zwarte lub normalnie rozwarne. Z punktu widzenia sprzętowego rozwiązania podłączenia do systemu mikroprocesorowego rodzaj styków nie ma żadnego znaczenia, jedynie podczas pisania programu należy uwzględniać negację powstającą pomiędzy oboma rodzajami styków (określenie stanu aktywnego linii). Konfiguracja styków jest natomiast bardzo ważna z uwagi na bezpieczeństwo i pewność działania urządzenia. Należy pamiętać, że stosowanie styków normalnie zwartych umożliwia wykrycie przerwy w linii łączącej taki styk z systemem mikroprocesorowym. Łatwo sobie wyobrazić co mogłoby się stać w przypadku awarii linii łączącej układ z wyłącznikiem bezpieczeństwa lub innym elementem odpowiadającym za awaryjne wyłączenie układu. Przy zastosowaniu styków w konfiguracji normalnie otwartej urządzenie nie zadziała w przypadku



Rys. 26

List. 9.

```

;STAN - pomocnicza zmienna bitowa przechowująca stan linii portu
;(program główny)
...
...
JB STAN, ETYKIETA ;reakcja programu na stan linii portu (tutaj skok)
...
...
ETYKIETA:
...
;(koniec programu)

PRZERWANIE_T0: ;procedura obsługi przerwania od odpowiednio
;zaprogramowanego licznika T0
... ;(inne rozkazy, np. przeładowanie licznika)
...
MOV C,P3.0 ;sprawdzamy stan linii P3.0
MOV STAN,C ;i kopiujemy go do pomocniczej zmiennej
...
...
RETI ;koniec obsługi przerwania

```

uszkodzenia przewodu łączącego. Jeśli natomiast zastosujemy styki normalnie zwarte, to uszkodzenie przewodu (najczęściej jest to przerwanie ciągłości obwodu) spowoduje reakcję układu taką samą jak w przypadku zadziałania danego urządzenia (np. wciśnięcia przycisku) i praca systemu zostanie wstrzymana. Podobne zalety można przedstawić w stosunku do styków normalnie rozwartych - stosuje się je w układach, w których przerwa w połączeniu nie

wywoła żadnych groźnych następstw, a zastosowanie styków normalnie zwartych mogłoby wywołać niekontrolowane działanie urządzenia w razie awarii okablowania (np. włączanie jakiegoś urządzenia w chwili wciśnięcia przycisku). Najogólniej mówiąc, należy przestrzegać zasady, że elementy (przyciski, przekaźniki, krańcówki i inne) ze stykami normalnie rozwartymi stosujemy w obwodach odpowiadających za uruchamianie i sterowanie jakimiś urzą-

dzeniami współpracującymi z systemem mikroprocesorowym, natomiast elementy ze stykami normalnie zwartymi stosujemy w obwodach wyłączających i obwodach bezpieczeństwa.

Powyższe rozważania dotyczą zastosowania dwóch przykładowych grup urządzeń jako źródła sygnałów w systemie mikroprocesorowym. Chociaż zapewne najczęściej stosowane są układy z elementami stykowymi, to w takim systemie mogą być zastosowane również innego rodzaju elementy czujnikowe czy inne źródła sygnałów, zasadniczo nie zmieni się jednak sposób ich sprzęgnięcia z mikrokontrolerem. Należy pamiętać tylko o parametrach używanych elementów i w razie potrzeby zadbać o dopasowanie poziomów elektrycznych, na co zwrócono uwagę przy omawianiu ogólnych zasad wykorzystywania portów jako wejścia. Bardzo ważną sprawą jest także zadbanie o minimalizację zakłóceń przy zastosowaniu długich przewodów połączeniowych oraz zwrócenie uwagi na ewentualne niebezpieczeństwa związane z różnego rodzaju przepięciami mogącymi zakłócać pracę układu lub nawet doprowadzić do jego uszkodzenia.

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