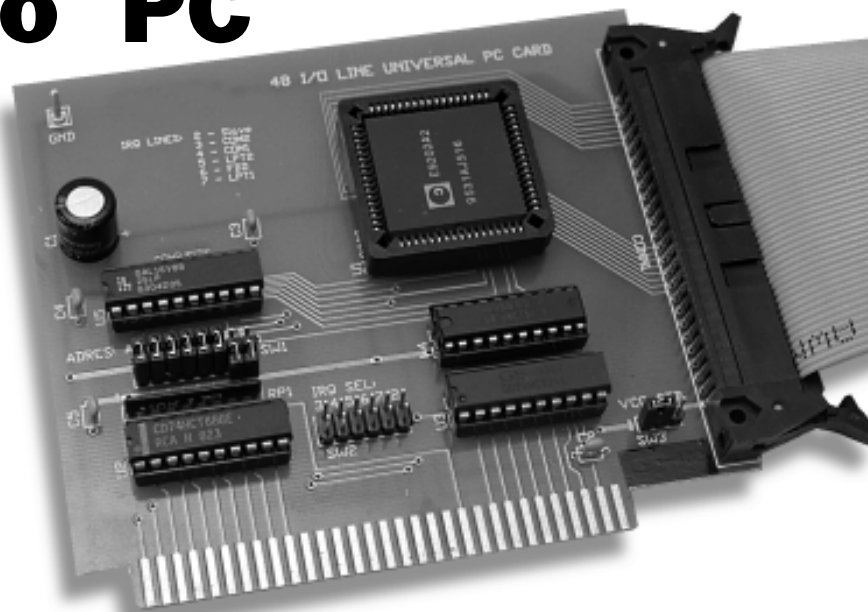


Uniwersalna karta I/O do PC

kit AVT-327

Przedstawiamy uniwersalną, wielowejściową kartę do komputerów PC. Dzięki zastosowaniu nowoczesnego układu firmy SAMES, użytkownik ma do dyspozycji 48 programowalnych linii wejścia/wyjścia, zorganizowanych w sześć 8-bitowych portów. Prosta konstrukcja oraz obsługa umożliwia zastosowanie karty wszędzie tam, gdzie istnieje potrzeba generowania bądź monitorowania wielu sygnałów cyfrowych.



Na łamach EP poruszaliśmy już temat sterowania urządzeń zewnętrznych za pośrednictwem komputerów PC. Opisywaliśmy uniwersalne karty wykorzystujące typowe układy serii TTL-LS oraz układy programowalnych portów I/O typu 8255.

Teraz proponujemy wykonanie prostej karty, opartej na układzie SA9203, produkowanym przez południowoafrykańską firmę SAMES.

W artykule przedstawimy opis układu elektrycznego karty, sposób programowania układu oraz podamy kilka przykładów programowania, dzięki którym każdy, nawet mało doświadczony programista-elektronik, będzie w stanie wykorzystać urządzenie dla potrzeb domowego lub szkolnego laboratorium. Zanim zapoznamy się ze schematem elektrycznym karty przyjrzyjmy się zastosowanemu w projekcie specjalizowanemu układowi firmy SAMES.

Układ SA9203

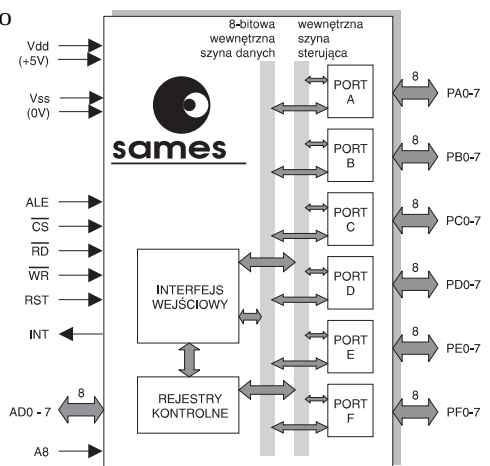
Zastosowany układ wielowejściowego portu I/O został wykonany w technologii CMOS. Posiada multipleksowaną szynę danych i adresów wewnętrznych portów, dzięki czemu doskonale nadaje się do zastosowania także w autonomicznych sterownikach wykorzystu-

jących mikrokontrolery jednokładowe (np. serii 8051). Kostka jest dostarczana przez producenta w typowej obudowie PLCC68.

Na rys.1 przedstawiono schemat blokowy układu. Składa się on z: interfejsu wejściowego, łączącego multipleksowaną szynę danych/adresów z zewnętrznym układem sterującym, rejestrów konfiguracyjnych oraz rejestrów sześciu portów A..F. Znaczenie zewnętrznych sygnałów sterujących podano w tab.1.

Na rys.2 przedstawiono adresy wewnętrznych rejestrów układu. Ze względu na 8-bitową szynę adresową AD0..AD7, dla zewnętrznego układu sterującego SA9203

Podstawowe dane techniczne	
✓	8-bitowa karta w standardzie ISA
✓	dostęp poprzez dwa porty w obszarze I/O komputera
✓	sześć 8-bitowych uniwersalnych portów wejścia wyjścia - 48 linii (standard TTL)
✓	możliwość indywidualnego programowania każdej linii jako wejście lub wyjście cyfrowe
✓	dwa tryby adresowania każdego wyprowadzenia: bajtowe lub bitowe
✓	programowalne rezystory podciągające każdy pin w trybie wejścia
✓	dotądowy port z sygnałem potwierdzenia przy odczycie danej
✓	programowalne rezystory przez kartę przerwania na wybranej linii IRQ komputera
✓	możliwość ustawienia dowolnego adresu bazowego karty (typowo: 300h i 301h)
✓	standardowe 50-pinowe złącze krawędziowe typu FC
✓	zasilanie: +5V, z szyny AT-BUS



Rys. 1. Wewnętrzna budowa układu SA9203.

Tabela 1.

Pin	Typ	Symbol	Opis
18,52		V _{DD}	zasilanie układu +5V
1,35		V _{SS}	masa zasilania 0V
61..68	we/wy	AD ₀ ..AD ₇	3-stanowa multipleksowana szyna danych/adresu: 8-bitowy adres zatraskiwany jest we wnętrzu układu SA9203 podczas opadającego zbocza sygnału ALE; dane zapisywane są lub odczytywane po podaniu odpowiednio sygnałów /WR (zapisu) lub /RD (odczytu)
2	we	A ₈	nie używany w układzie SA9203, powinien być połączony z masą (Vss), znaczenie tego sygnału opisujemy w dalszej części artykułu
3	we	/CS	aktywny niski sygnał na tym wejściu powoduje wybór układu
4	we	ALE	sygnał kontrolujący zatrzaśnięcie adresu na szynie AD0..AD7 podczas zapisu przez urządzenie zewnętrzne, następuje to podczas opadającego zbocza tego sygnału
5	we	/RD	poziom niski na tym wejściu pozwala na odczyt wewnętrznych rejestrów układu
6	we	/WR	poziom niski na tym wejściu powoduje zapis danej do wewnętrznego rejestru układu
7	wy	INT	programowane wyjście zgłoszenia przerwania do układu zewnętrznego, możliwość ustalenia polaryzacji oraz uaktywnienia tego sygnału
8	we	RST	wysoki poziom podany na to wejście powoduje wyzerowanie układu, parametry portów po tej operacji podane zostaną w dalszej części
9	we	STB	wejście zatraskiwania danej w porcie A, gdy port ten pracuje jako wejście, polaryzacja tego sygnału oraz sposób aktywacji podamy w dalszej części
10..17	we/wy	PA ₀ ..PA ₇	uniwersalny 8-bitowy port I/O; możliwość indywidualnego zdefiniowania każdej linii portu jako zatraskiwanego wyjścia lub wejścia; jako wejście port może pracować w trybie zatraskiwania (sygnałem STB) lub jako "przezroczysty" ("transparent")
19..26	we/wy	PB ₀ ..PB ₇	8-bitowy uniwersalny port I/O; wszystkie piny mogą być ustawione jako zatraskiwane wyjścia lub jako wejścia typu "transparent"
27..34	we/wy	PC ₀ ..PC ₇	identyczny jak port B
36..43	we/wy	PD ₀ ..PD ₇	identyczny jak port B
44..51	we/wy	PE ₀ ..PE ₇	identyczny jak port B
53..60	we/wy	PF ₀ ..PF ₇	identyczny jak port B

zajmuje obszar 256 bajtów w przestrzeni adresowej. Wybrane obszary układu są zajęte przez porty konfiguracyjne oraz porty A..F jak pokazano na rys.2.

Układ umożliwia dwojakie adresowanie każdego z rejestrów portów A..F. Pierwszy sposób polega na jednoczesnym adresowaniu całego portu, drugi pozwala na zaadresowanie pojedynczej linii każdego z portów. W tym drugim przypadku przy odczycie danej

MNEMONIK	REJESTR	ADRES W TRYBIE	
		BAJTOWYM	BITOWYM
PA	PORT A	00h	00h...07h
PB	PORT B	08h	08h...0Fh
PC	PORT C	10h	10h...17h
PD	PORT D	18h	18h...1Fh
PE	PORT E	20h	20h...27h
PF	PORT F	28h	28h...2Fh
PACR	REJESTR KONTROLNY PORTU A	70h	70h
PAICR	REJESTR KONFIGURACJI PORTU A W TRYBIE WEJŚCIA	71h	71h
IOCR	REJESTR KONFIGURACJI PORTÓW B..F	72h	72h
PAMR	REJESTR TRYBU ADRESOWANIA PORTÓW	73h	73h

Rys. 2. Mapa adresowa rejestrów układu SA9203.

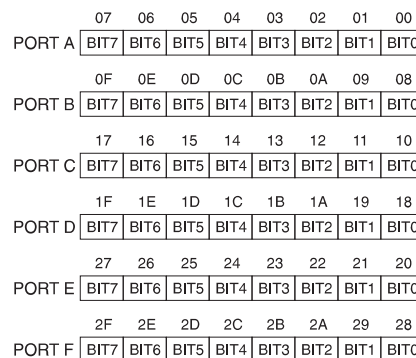
w postaci bajtu siedem najstarszych bitów nie ma znaczenia, jedynie najmłodszy D0 wskazuje na stan linii lub wymusza go w przypadku pracy portu jako cyfrowego wyjścia.

Dodatkowe rejestry sterujące pracą całego układu zawsze adresowane są bajtowo. Dość użyteczną funkcją w przypadku tych rejestrów jest możliwość odczytu ich zawartości. Dzięki temu programista nie musi zapamiętywać ich stanu po zapisie w dodatkowych zmiennych wykorzystywanych w programie.

Na **rys.3** jest przedstawiona mapa adresowa poszczególnych portów w trybie adresowania bitowego.

Poniżej opisujemy znaczenie poszczególnych portów kontrolnych układu SA9203:

PACR - rejestr kontrolny portu A. Na **rys.4** pokazano znaczenie poszczególnych bitów rejestru. Odpowiednie ustawienie lub wyzerowanie pozycji pozwala na indywidualne ustale-



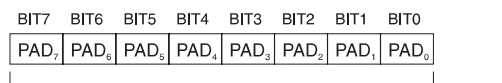
Rys. 3. Adresy bitowe poszczególnych rejestrów.

nie linii portu A jako wejścia lub wyjścia cyfrowego.

PAICR - rejestr konfiguracyjny portu A w trybie wejścia. **Rys.5** przedstawia znaczenie bitów rejestru. W przypadku ustawienia linii jako zatraskiwanego wejścia (z ang. latched input) dane zostają zapamiętane po nadejściu sygnału na wejściu STB. W przypadku ustawienia dowolnej linii portu A jako wyjścia, odpowiadający mu bit w rejestrze PAICR nie ma wpływu na działanie tej linii portu.

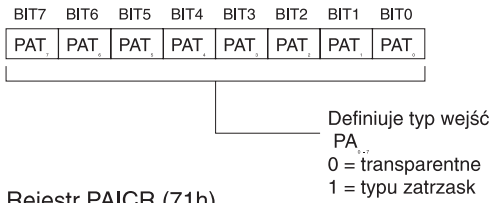
IOCR - rejestr konfiguracyjny wejścia-wyjścia. Rejestr pozwala na konfigurację portów B..F, załączanie wewnętrznych rezystorów podciągających w trybie wejścia, oraz określa polaryzację sygnału STB (patrz **rys.6**). Jak wspomniano wcześniej, wszystkie wyprowadzenia portów układu SA9203 posiadają wbudowane rezystory podciągające końcówkę portu, kiedy ten pracuje jako wejście. Zwalnia to użytkownika od stosowania dodatkowych elementów rezystancyjnych w niektórych aplikacjach. Rezystory te mogą być uaktywnione lub wyłączone oddzielnie dla każdego z portów A..F

Bit IOCR.2 definiuje polaryzację sygnału STB (ang. strobe), który zatraskuje daną w porcie A (lub wybranych jego liniach), pracującym jako wejście.



Rejestr PACR (70h)

Rys. 4. Znaczenie poszczególnych bitów rejestru PACR.



Rejestr PAICR (71h)

Rys. 5. Znaczenie bitów rejestru PAICR.

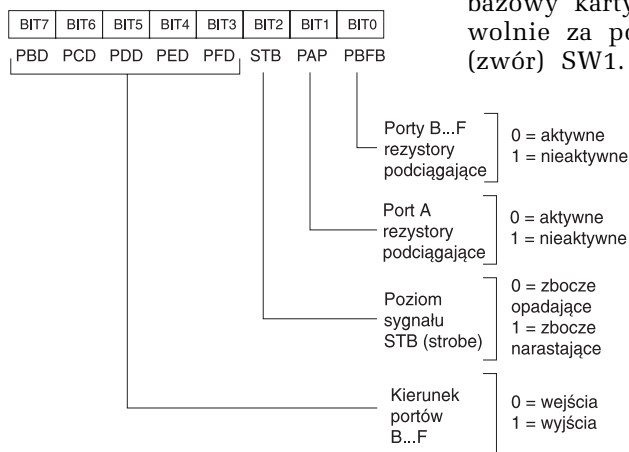
W przypadku ustawienia bitu na 0, dane są zatraskiwane podczas opadającego zbocza sygnału STB, a dla 1 podczas narastającego zbocza sygnału.

PAMR - rejestr trybu adresowania. Zgodnie z **rys.7**, ustawienie lub wyzerowanie odpowiedniego bitu w tym rejestrze pozwala na zmianę trybu adresowania poszczególnych portów A..F, aktywację oraz polaryzację sygnału zgłoszenia przerwania INT.

INT - wyjście zgłoszenia przerwania w przypadku zatrzaśnięcia danej w porcie A po nadejściu sygnału STB. Aktywacja linii następuje po odpowiednim ustawieniu bitu PAMR.1. Polaryzacja zgłoszenia przerwania ustalana jest poprzez bit PAMR.0.

RST - podanie wysokiego poziomu na to wejście zeruje cały układ. Zawartość wszystkich rejestrów zostaje wyzerowana. Następnym tego jest:

- wszystkie piny portów A..F ustawione zostają jako wejścia;
- wejścia portu A pracują jako transparentne;
- rezystory podciągające są uaktywnione we wszystkich portach;
- wyjście przerwania jest nieaktywne;



Rys. 6. Znaczenie bitów rejestru PACR.

- adresowanie portów jest ustalone jako bajtowe.

Aktywny (wysoki) stan sygnału RST powinien trwać minimum 100ns.

Po niezbędnych informacjach dotyczących układu SA9203 zapoznajmy się z budową karty.

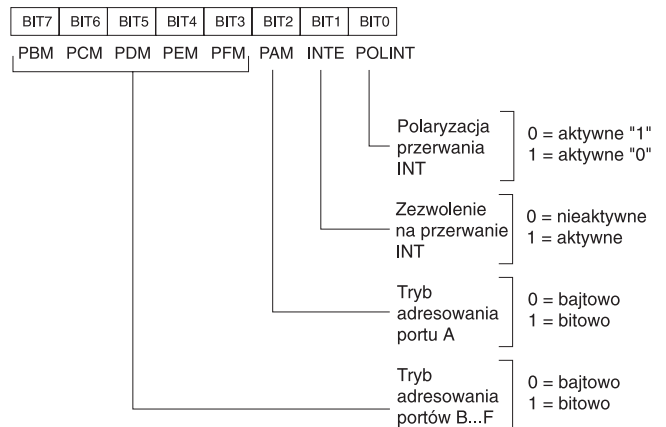
Opis układu

Schemat elektryczny układu karty jest przedstawiony na **rys.8**. W celu uproszczenia konstrukcji w układzie zastosowano programowany układ logiczny GAL16V8 - IC5. Aby ułatwić analizę całego układu na **rys.9** przedstawiono jego strukturę wewnętrzną.

Głównym elementem całego układu jest oczywiście IC1. Układ IC2 spełnia rolę dekodera adresu. Obsługa karty z punktu widzenia programu jest 2-stopniowa. Najpierw należy zapisać adres rejestru, do którego chcemy się odwołać, a następnie należy wpisać lub odczytać daną z tego rejestru. Taki, indeksowy sposób adresowania karty, minimalizuje zajęty obszar z przestrzeni adresowej I/O komputera do dwóch portów.

Pierwszy port (przesunięcie +0) umożliwia zaadresowanie wewnętrznych rejestrów układu SA9203, drugi (przesunięcie +1) zapis/odczyt danej spod zaadresowanego wcześniej rejestru. Adres bazowy karty można ustawić dowolnie za pomocą przełączników (zwór) SW1. W typowych aplikacjach adres bazowy powinien zawierać się w zakresie 300..31Eh, co jest standardem przewidzianym dla kart prototypowych w komputerach PC. Drabinka rezystorowa RP1 podciąga wejścia Q0..Q7 komparatora IC2 do plusa zasilania.

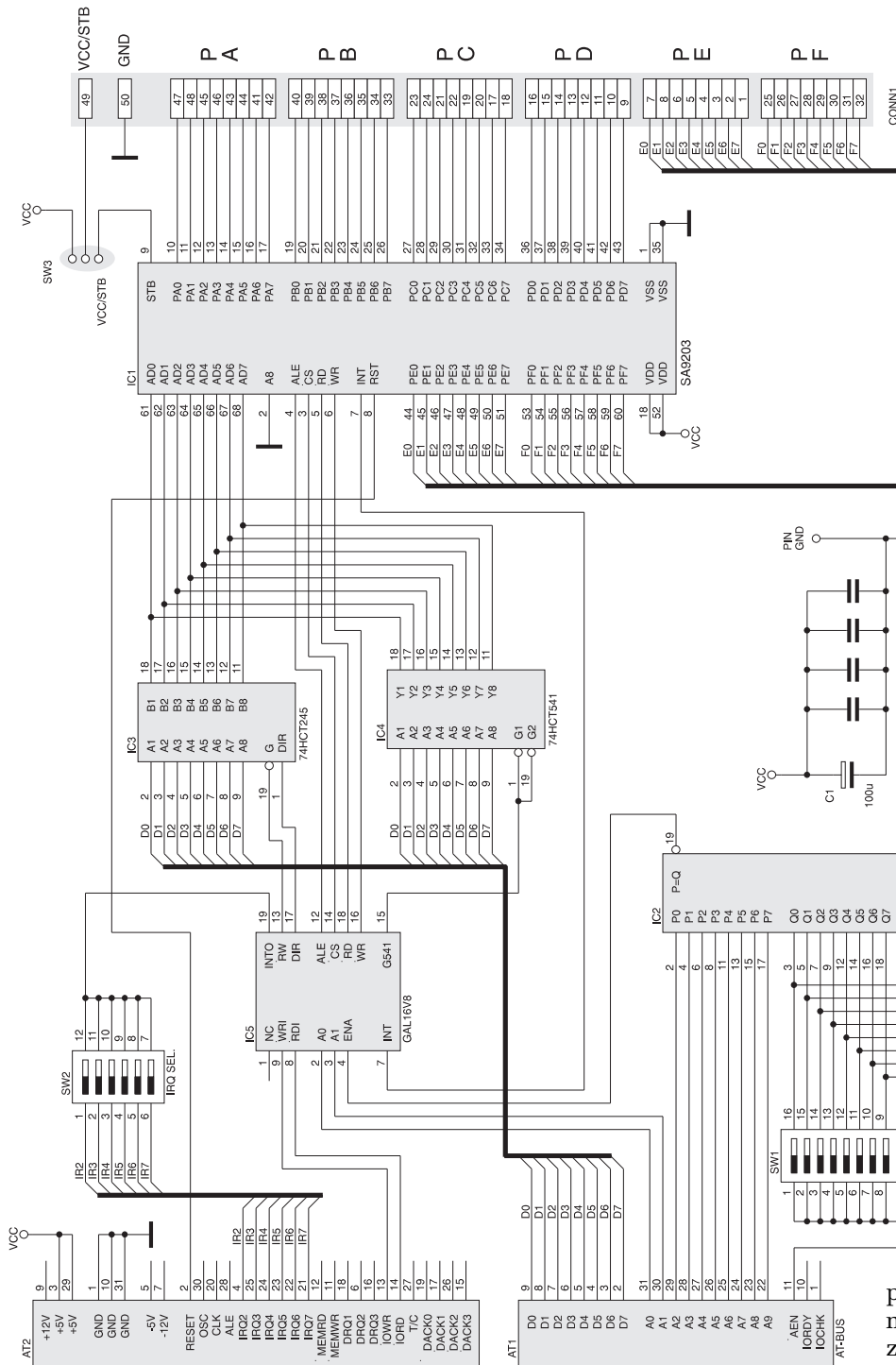
Ze względu na multipleksowaną szynę adresu/danych, w układzie IC1 konieczne stało się zastosowanie dodatkowych układów pośredniczących pomiędzy szyną AT-BUS komputera PC. Rola tę pełnią układy IC3 i IC4 oraz IC5. Układ IC3 to ośmiokrotna, dwukierunkowa brama, poprzez którą zapisywane są lub odczytywane dane z układu SA9203. Drugi układ IC4, to jednokierunkowa brama. Jej zadaniem jest przekazanie 8-bitowego adresu rejestru w układzie IC1.



Rejestr PAMR (73h)

Rys. 7. Znaczenie bitów rejestru PAMR.

W sytuacji kiedy następuje zaadresowanie układu IC1 zachodzą następujące zdarzenia. Na szynę komputera zostaje podany adres bazowy karty. W przypadku zgodności stanów na wejściach P0..P7 IC2 oraz ustawionej kombinacji wejść Q0..Q7 IC2, na wyjściu P=Q IC2 pojawia się niski stan logiczny, który jest następnie podawany na wejście ENA układu GAL - IC5. We wnętrzu IC5 (patrz **rys.9**) odblokowany zostaje dekoderek DEK (1 z 8). W przypadku adresowania układu IC1 na linii A0 i A1 panuje logiczne 0, wobec czego na wyjściu Y4 dekodera '138 pojawi się stan niski, co spowoduje wybranie układu IC1 (/CE=0). Ponieważ adresowanie rejestrów wewnętrznych układu SA9203 wiąże się z zapisem danej, sygnał WRIN przyjmie stan niski, co w konsekwencji spowoduje pojawienie się stanu „1“ na wyjściu bramki B3. Bramka ta jest połączona z wejściem zatraskiwania adresu ALE układu IC1. Teraz po podaniu danych na szynę AT-BUS D0..D7, przy narastającym



Rys. 8. Schemat elektryczny układu.

zbozcu sygnału IOWR (złącze AT2) w układzie IC1 zostanie za-trzaśnięty adres rejestru, do któ-rego ma nastąpić zapis lub odczyt danej.

Rys.10 pokazuje przebiegi sygnałów podczas operacji zapisu i odczytu. Ze względu na czytelność rysunku nie podano zakre-sów czasów pomiędzy kolejnymi zbozczami sygnałów sterujących. Zastosowany w naszej karcie

układ kombinacyjny IC5 spełnia założenia konstrukcyjne pod względem zależności czasowych, co zapewnia prawidłowe sterowa-nie układu SA9203.

W drugiej fazie komputer, chcąc np. odczytać daną z zaad-resowanego wcześniej rejestru, wystawia adres karty z przesunię-ciem +1, co powoduje pojawienie się stanu niskiego tym razem na wyjściu Y5 dekodera '138 (w IC5).

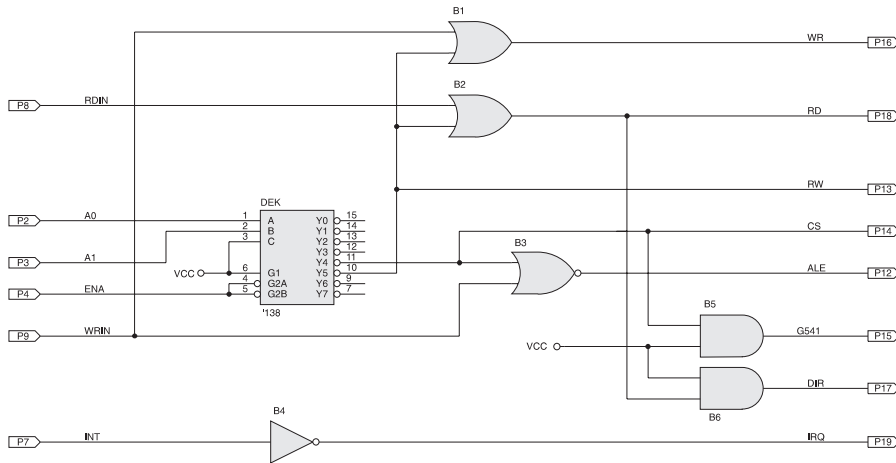
Ponieważ przy odczycie sygnał IORD (AT2) przyjmie stan „0”, odblokowana zostaje bramka B2. W efekcie na wejściu /RD układu IC1 oraz DIR - IC3 pojawia się stan niski, co powoduje przekazanie danej z portu AD0..AD7 IC1 poprzez IC3 na szynę danych D0..D7 komputera.

W przypadku zapisu danej do rejestru sytuacja jest podobna, z tym, że tym razem sygnał IOWR (AT2) przyjmie stan niski, co wraz z Y5=0 ('138) powoduje otwarcie bramki B1 (w IC5). Wejście WR IC1 przyjmie poziom „0”, a wejście DIR IC3 pozostanie w stanie wysokim, umożliwiając przekazanie danej z szyny komputera D0..D7 do układu IC1.

Dodatkowe bramki B5 i B6 w strukturze układu IC5 przekazują sygnał RD oraz CS na wyjścia odpowiednio DIR oraz G541. Ich zastosowanie było konieczne przy projektowaniu struktury GAL-a w programie kompilatora. W praktyce, jeżeli ktoś z Czytelników zechce zastosować klasyczne układy scalone w miejsce IC5, bramki B5 i B6 można pominąć, a odpowiednie sygnały zewrzeć ze sobą (RD z DIR, CS z G541).

Nieco kontrowersyjne z pozoru może wydawać się użycie sygnału linii adresowej A1 komputera. Otóż podanie adresu karty z przesunięciem +2 lub +3, blokuje ją, zawężając tym samym obszar zajęty w przestrzeni adresowej I/O komputera. Ta oszczędność jest oczywista, a jej realizacja byłaby niemożliwa z wykorzystaniem układu IC2, ze względu na zbyt małą liczbę wejść - 8, zajętych przez linie adresowe A9..A2 szyny AT-BUS.

W układzie IC5 zawarto także dodatkowy inwerter sygnału zgłoszenia przerwania INT (rys.9). Jest to konieczne ze względu na to, że układ IC1 po resecie (sygnał RST) ustala logiczne „1”, co w efekcie, przy zwartej którejś

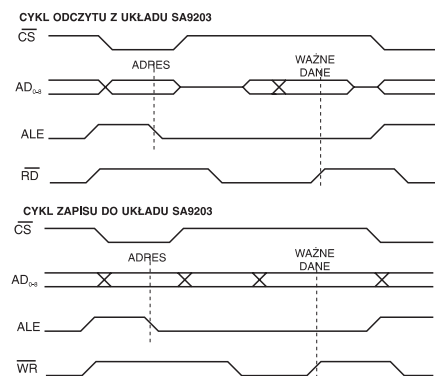


Rys. 9. Budowa dekodera adresowego.

zworze w SW2, mogłoby spowodować nieprzewidziane skutki po uruchomieniu komputera. Programując układ przerwań IC1 (polaryzacja INT) należy pamiętać o tym, że sygnał generujący przerwianie w komputerze jest zanegowany (przez B4).

Karta nie posiada zatrasku zapamiętującego stan pinu INT IC1, co umożliwiłoby monitorowanie zgłoszenia odebrania danych do portu A bez programowania kontrolera przerwań PC. Użycie tego trybu pracy portu A wymusza tę procedurę, dlatego do wejścia przełącznika SW2 doprowadzono wszystkie linie zgłoszenia przerwań IRQ2..IRQ7. Przy korzystaniu z tej opcji należy pamiętać, że w zasadzie wszystkie linie są używane przez inne urządzenia w komputerze PC. I tak:

- IRQ2** - obsługuje drugi układ kontrolera przerwań (w komputerach AT i lepszych)
- IRQ3** - obsługuje port szeregowy COM2,
- IRQ4** - COM1;



Rys. 10. Przebiegi charakterystyczne dla pracy układu.

- IRQ5** - port drukarkowy LPT2;
- IRQ6** - stację dysków FDD;
- IRQ7** - port LPT1.

Wszystkich zainteresowanych wykorzystaniem kontrolera przerwań do obsługi karty odsyłam do lektury książki [1].

Ze względu na zgodność polaryzacji sygnału resetowania komputera PC, pin RST IC1 połączono bezpośrednio w szynę RESET - AT2. W ten sposób podczas resetu komputera skasowaniu ulega także układ IC1.

Wszystkie sygnały portów A..F oraz masa zasilania wyprowadzono na złącze CONN1. Ze względu na zbyt małą liczbę końcówek - 50, zastosowano dodatkową zworę SW3. Dzięki niej, w zależności od potrzeb, na wyprowadzenie 49 CONN1 można wyprowadzić sygnał STB układu IC1 lub zasilanie +5V.

Zastosowane w układzie kondensatory C1..C5 filtrują napięcie zasilające.

Montaż i uruchomienie

Cały układ elektryczny karty zmontowano na 2-stronnej płytce drukowanej, wykonanej w standardzie typowym dla krótkich kart ISA. W celu zwiększenia niezawodności działania, złącze ISA karty zostało dodatkowo pozłoczone.

Rozmieszczenie elementów przedstawia rys.11. Montaż układu jest bardzo prosty. Zanim do tego przejdziemy należy lekko zeszlifować krawędzie płytki drukowanej pilnikiem lub drobnym papierem ściernym, zwracając uwagę, aby nie uszkodzić złoczonej powierzchni złączy AT1

i AT2. Następnie należy wlutować podstawki pod układy IC2..IC5, a następnie pod IC1, zwracając uwagę na odpowiednie jej ustawienie względem obrysu na płycie drukowanej.

Wlutowanie zwor SW1, SW2, SW3, r-packa RP1 oraz złącza CONN kończy montaż. Na płycie drukowanej przewidziano miejsce na wlutowanie typowego kołka Stocka, do którego doprowadzona jest masa zasilania. W pracach, szczególnie laboratoryjnych, ten przysłowiowy „punkt zaczepienia” okazuje się często bardzo przydatny.

Przykład 1. Skonfigurowanie rejestru portu A - PACR.

```

Asm x86
mov dx,300h ; u nas adres bazowy karty = 300h
mov al,71h ; adres rejestru PACR
out dx,al ; zapis adresu rejestru układu IC1
mov dx,301h ; adres zapisu/odczytu danych karty
mov al,11110000b ; PA7..PA4 jako
; wyjścia,PA3..PA0 - wejścia
out dx,al ; zapis konfiguracji do rejestru PACR
    
```

Turbo Pascal

```

port[$300]:=$71; { zapisanie adresu rejestru PACR }
port[$301]:=$F0; { zapisanie danej 11110000 w tym rejestrze}
    
```

Przykład 2. Zapis danej ("85") do portu C układu IC1.

```

Asm x86
; część konfigurująca IC1
mov dx,300h ; adres bazowy karty
mov al,72h ; adres rejestru IOCR - patrz rys.6
out dx,al ; zapis adresu IOCR
inc dx ; adres zapisu/odczytu danych karty
mov al,01000000b ; port C - wyjście, reszta wejścia
out dx,al ; zapis do rejestru IOCR
; część zapisująca dana do portu C
mov dx,300h ; adres bazowy
mov al,10h ; adres portu C
out dx,al ; no i zapis adresu portu C
inc dx ; adres zapisu/odczytu danych karty
mov al,55h ; 85 dziesiętnie
out dx,al ; i zapis danej do portu C
    
```

Turbo Pascal

```

port[$300]:=$72; { zapisanie adresu rejestru IOCR }
port[$301]:=$40; { port C - wyjście, reszta wejścia }
port[$300]:=$10; { zapis adresu portu C }
port[$301]:=$85; { i zapis danej do portu C }
    
```

Przykład 3. Odczyt danej z portu F układu IC1 z konfiguracją bez rezystorów "pull-ups"

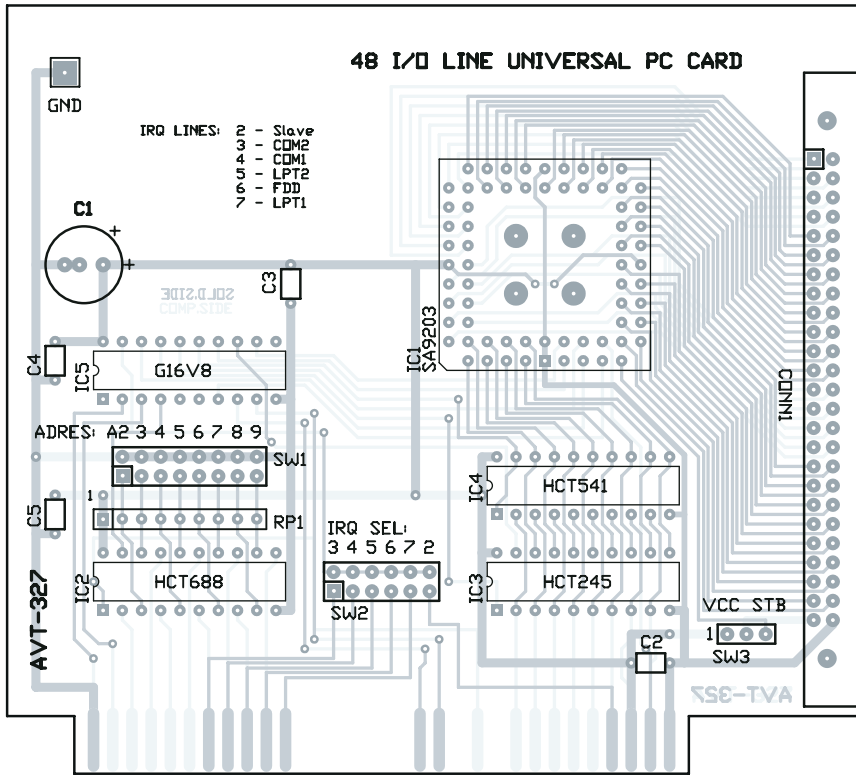
```

Asm x86
; część konfigurująca IC1
mov dx,300h ; adres bazowy karty
mov al,72h ; adres rejestru IOCR - patrz rys.6
out dx,al ; zapis adresu IOCR
inc dx ; adres zapisu/odczytu danych karty
mov al,11110001b ; port F - wejście bez
; "pull-ups", reszta
; wyjścia
out dx,al ; zapis do rejestru IOCR
; część odczytująca daną z portu F
mov dx,300h ; adres bazowy
mov al,28h ; adres portu F
out dx,al ; najpierw zapis adresu portu F
inc dx ; adres zapisu/odczytu danych karty
in al,dx ; i odczyt danej z portu F
    
```

Turbo Pascal

```

port[$300]:=$72; { zapisanie adresu rejestru IOCR }
port[$301]:=$F1; { port F - wejście bez "pull-ups", reszta
wyjścia }
port[$300]:=$28; { zapis adresu portu F }
dana:=port[$301]; { i odczyt danej z portu F }
    
```



Rys. 11. Rozmieszczenie elementów na płytce drukowanej.

Po włożeniu układów scalonych w podstawki należy ustawić zwory na SW1..SW3.

Do testów proponuję ustawienie:

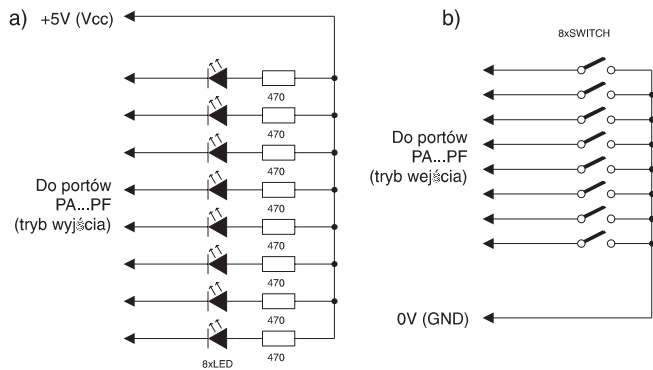
- SW1: adresu bazowego na 300h: A2,3,4,5,6,7 - zwarte, A8,9 - rozwarte;
- SW2: zgłoszenia przerwania - wszystkie rozwarte;
- SW3: wyprowadzenie zasilania +5V na pin 49 złącza CONN1.

Sprawdzenie karty najlepiej jest przeprowadzić stosując dodatkowy układ, którego schemat elektryczny przedstawiono na rys.12a i b. W przypadku a) można zbadać porty IC1 w trybie wyjścia, natomiast układ b) złożony z ośmiu mikroprzełączników (np. typu SW-DIP8) pozwoli na zbadanie

poszczególnych portów w trybie wejść cyfrowych.

Autor oba układy testowe umieścił na płytce uniwersalnej, którą połączył z kartą (ok. 1m) przewodem taśmowym AWG-50, zakończonym z obu stron wtykami FC-50.

Do przetestowania karty, oprócz proponowanego testera, niezbędny będzie program. Do kitu dołączana jest dyskietka z prostym programem testującym porty w komputerze PC. Za pomocą niego można wpisać lub odczytać dowolny port. Obsługa programu jest trywialna, a wszelkie wyjaśnienia zainteresowani czytelnicy znajdą w pliku „CZYTAJ.TO!“ także zamieszczonym na dyskietce.



Rys. 12. Przykładowe układy do testowania karty.

Tym, którzy zdecydują się na samodzielne wykonanie karty, należy się kilka wskazówek odnoszących się do programowania naszej, wielowięściowej karty.

Przykłady podane są w języku Turbo Pascal oraz Asembler x86.

W praktyce nie jest konieczne każdorazowe adresowanie rejestru wewnętrznego układu IC1. Jeżeli np. cyklicznie odczytujemy lub zapisujemy dane do tego samego portu, wystarczy zaadresować go raz na początku. Wbudowany w IC1 zatrask adresu (sterowany sygnałem ALE) przechowuje ostatnio zapisany adres do nadejścia kolejnego zbocza ALE.

Na koniec uwaga dotycząca dodatkowego, lecz nie wykorzystanego sygnału A8 układu IC1. Istnieje wersja układu IC1 pod oznaczeniem SA9202. Kostka odpowiada funkcjonalnie układowi SA9203, z tym, że dodatkowo zawiera w swojej strukturze 256 bajtów statycznej pamięci RAM (adresy 100h..1FFh). Właśnie końcówka A8 pozwala na zaadresowanie tego obszaru pamięci poprzez podanie na A8 logicznej jedynki. W naszym układzie karty nie jest możliwe wykorzystanie pamięci RAM układu SA9202, ze względu na brak sterowania pinu adresowego A8. Oczywiście karta będzie pracowała poprawnie z obiema wersjami układów. Zainteresowanych Czytelników odsyłam do lektury pozycji [2].

Sławomir Surowiński, AVT

Literatura

- [1] Piotr Metzger - „Anatomia PC“, Helion wyd. I lub II.
- [2] „Telecommunications IC Databook“ - SAMES 1993.

WYKAZ ELEMENTÓW

Rezystory

RP1: R-pack 4,7k..10kΩ (SIL-9)

Kondensatory

C1: 100µF/10V

C2..C5: 100nF

Półprzewodniki

IC1: SA9203 (Sames)

IC2: 74HCT688

IC3: 74HCT245

IC4: 74HCT541

IC5: GAL16V8 zaprogramowany

Różne

SW1: goldpin 2x8 + 8 jumperów

SW2: goldpin 2x6 + 6 jumperów

SW3: goldpin 1x3 + jumper

CONN1: gniazdo AWP-50 kątowe

podstawka PLCC68 - 1 szt.

podstawki DIL20 - 4 szt.