1. Mikrokontroler 8051

1.1. Wstęp

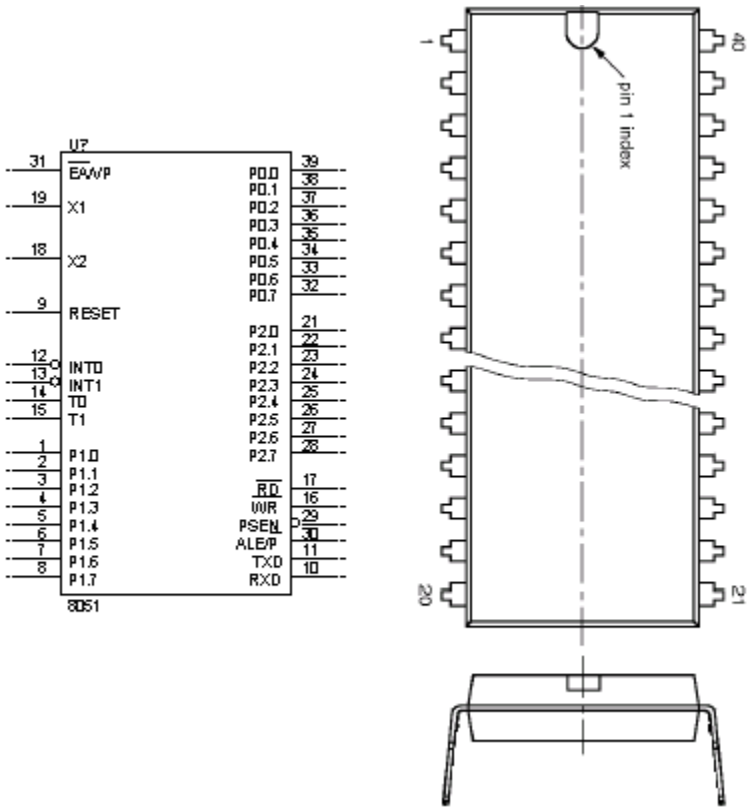
Mikroprocesor jest układem elektronicznym, którego zadaniem jest przetwarzanie danych według określonego programu. Typowy mikroprocesor składa się z jednostki arytmetyczno-logicznej odpowiedzialnej za wykonywanie działań, bloku rejestrów (oraz niekiedy wewnętrznej pamięci RAM), wewnętrznej szyny danych, układu sterowania, wewnętrznego kontrolera przerwań oraz bloku współpracy z zewnętrznymi magistralami. Niektóre mikroprocesory zawierają także inne bloki, np. układy czasowo-licznikowe, porty we/wy, układy transmisji szeregowej czy pamięć podręczną (*cache*). Tego typu układy często są nazywane mikrokontrolerami. Do poprawnej pracy mikrokontrolery wymagają:

- zasilania;
- sygnału zegara lub w przypadku, gdy układ zegara jest zawarty wewnątrz mikrokontrolera, jedynie przyłączenia rezonatora;
- układu wytwarzającego sygnał Reset, który umożliwia rozpoczęcie pracy po ustabilizowaniu się napięcia zasilania.

Mikrokontroler wraz z przyłączonymi do niego urządzeniami we/wy tworzy system mikroprocesorowy.

1.2. Informacje ogólne

Układ 8051 jest jednoukładowym mikrokontrolerem 8-bitowym. Może być umieszczony w 40-nóżkowej obudowie typu DIL, zaprezentowanej wraz z układem wyprowadzeń na rys. 2.1.



Rys. 2.1. Mikrokontroler 8051; opis wyprowadzeń oraz wygląd dla obudowy DIL

Poszczególne wyprowadzenia układu spełniają następujące funkcje opisane w tabl. 2.1.

Tablica 2.1. Wyprowadzenie mikrokontrolera 8051 oraz funkcje jakie spełniają

Wyprowadzenia		Funkcja
Nr	oznaczenie	
1 ÷ 8	P1_0-P1_7	wyprowadzenia uniwersalnego portu we/wy P1
9	RST	końcówka Reset służąca do zerowania mikrokontrolera
10 ÷ 17	P3_0-P3_7	wyprowadzenia uniwersalnego portu we/wy P3; poszczególne końcówki mogą pełnić także inne funkcje - omówione przy opisie portu P3
18, 19	XTAL1, XTAL2	wyprowadzenia służące do przyłączenia rezonatora lub zewnętrznego generatora impulsów zegarowych
20	GND	wyprowadzenie masy układu
21 ÷ 28	P2_0-P2_7	wyprowadzenia uniwersalnego portu we/wy P2; port ten pełni również funkcję starszego bajtu szyny adresowej przy dostępie do pamięci zewnętrznej
29	PSEN	końcówka informująca o odczycie zewnętrznej pamięci programu
30	ALE	sygnał wyjściowy służący do zatrzaskiwania młodszej części adresu przy dostępie do pamięci zewnętrznej
31	EA	stan na tym wejściu określa sposób współpracy z pamięcią programu
32 ÷ 39	P0_0-P0_7	wyprowadzenia uniwersalnego portu we/wy P0.; port ten pełni również funkcje młodszego bajtu szyny adresowej multipleksowanego z szyną danych podczas dostępu do pamięci zewnętrznej
40	Vcc	końcówka zasilania układu; Vcc = 5V +/– 0,25V

Cechami charakterystycznymi układu 8051 są rozbudowane zasoby wewnętrzne, takie jak:

- programowalne układy czasowe,
- układ transmisji szeregowej,
- układ przerwań,
- wewnętrzna pamięć programu (4 kB) i danych (128 bajtów),
- możliwość przyłączenia zewnętrznej pamięci programu i danych (64 kB),
- rozwinięta, jednolita lista rozkazów.

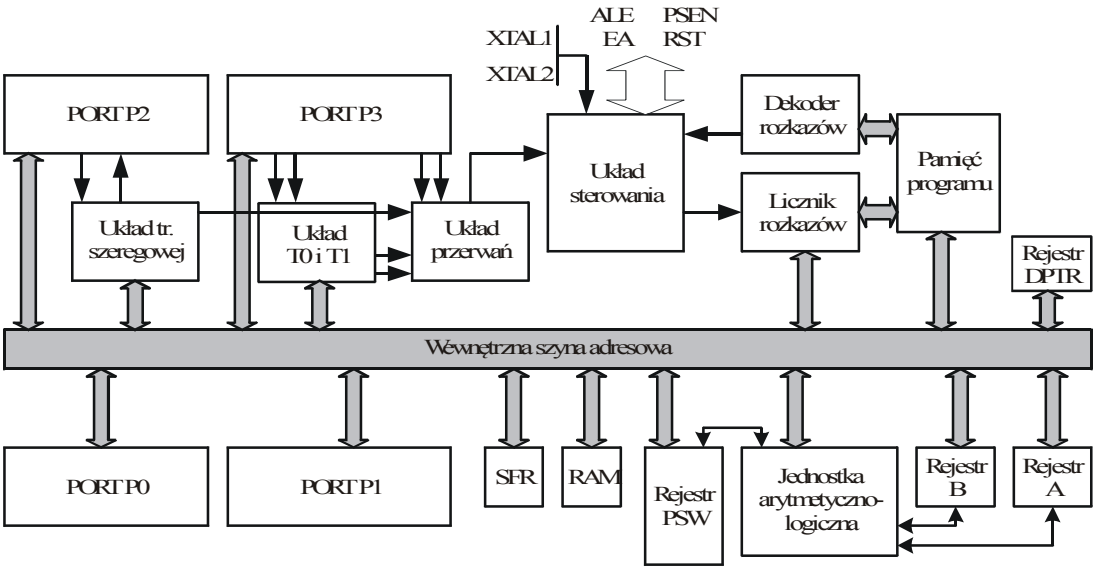
Mikrokontrolery rodziny MCS-51 są aktualnie produkowane w rozmaitych odmianach przez wiele firm. Poszczególne wersje układu są zgodne pod względem programowym z pierwowzorem, ale zazwyczaj mają pewne dodatkowe elementy.

1.3. Architektura mikrokontrolera

W mikrokontrolerze 8051 możemy wyróżnić następujące bloki:

- jednostka arytmetyczno - logiczna,
- pamięć danych (wewnętrzna użytkownika – 128 bajtów oraz zewnętrzna 64 kB),
- blok rejestrów specjalnych SFR – 128 bajtów, znajdujący się w obszarze wewnętrznej pamięci danych,
- pamięć programu (wewnętrzna 4 kB oraz zewnętrzna do 64 kB),
- porty we/wy,
- programowany układ czasowy,
- układ transmisji szeregowej,
- układ przerwań, generator sygnału taktującego,
- układ sterowania (wraz ze sterowaniem trybami uśpienia - obniżonego poboru mocy - tylko w 80C51).

Architektura mikrokontrolera 8051 jest pokazana na rys. 2.2.



Rys. 2.2. Architektura mikrokontrolera 8051

Mikrokontroler 8051 można podzielić na kilka bloków funkcyjnych, które współpracują ze sobą. Układ sterowania dekoduje rozkazy przesłane z pamięci programu i na ich podstawie generuje odpowiednie sygnały, sterujące pozostałymi elementami mikrokontrolera. Wykorzystując wyprowadzenia XTAL1 i XTAL2, do układu sterowania jest przyłączony rezonator kwarcowy. Współpracuje z nim także układ przerwań (układ sterowania jest odpowiedzialny za sprzętową generację rozkazu LCALL wywołania procedury obsługi przerwania).

W mikrokontrolerze 8051 istnieje pięć możliwych źródeł przerwań: licznik T0, licznik T1 , przerwanie zewnętrzne INT0, przerwanie zewnętrzne INT1 oraz przerwanie od układu transmisji szeregowej.

Układ sterowania jest również odpowiedzialny za generację sygnałów zewnętrznych PSEN i ALE. Z układem sterowania współpracuje rejestr rozkazów (rejestr przechowujący ostatnio pobrany rozkaz).

Wszystkie operacje sterowania wewnętrznymi układami peryferyjnymi oraz przesyłania danych między nimi a CPU, odbywają się za pośrednictwem rejestrów SFR. Obszar rejestrów SFR jest wykorzystywany dwojako — z jednej strony umieszczone są w nim wszystkie rejestry sterujące pracą mikrokontrolera lub wykorzystywane bezpośrednio przy wykonywaniu programu, z drugiej zaś strony rejestry SFR stanowią rodzaj interfejsu między mikroprocesorem a układami peryferyjnymi, umieszczonymi wewnątrz mikrokontrolera. Dostęp do każdego z tych rejestrów jest możliwy wyłącznie w trybie adresowania bezpośredniego. Do adresowania pamięci kodu programu ROM służy rejestr DPTR (16 bitów).

Standardowy mikrokontroler 8051 ma 4 kB pamięci programu typu ROM. Pamięć programu przechowuje kody operacji przeznaczonych do wykonania przez mikroprocesor, może także służyć do przechowywania stałych używanych w programie. Pamięć programu jest adresowana przez 16-bitowy licznik rozkazów PC (ang. *Program Counter*). PC zawiera adres aktualnego rozkazu przeznaczonego do wykonania. Rozkaz ten jest pobierany z pamięci do rejestru rozkazów (rejestr ten nie jest dostępny programowo). Na podstawie zawartości rejestru rozkazów, dekoder rozkazów steruje wyborem źródła argumentu, miejscem umieszczenia wyniku, funkcjami arytmometru itp. — w ten sposób mikroprocesor wykonuje zadaną operację. Jeżeli nie jest wykonywany rozkaz skoku, zawartość licznika rozkazów jest inkrementowana po odczycie każdego bajtu z pamięci programu.

Arytmometr, składający się z jednostki arytmetyczno - logicznej rejestrów pomocniczych oraz dekodera rozkazów, współpracuje z dwoma rejestrami z bloku rejestrów specjalnych:

- akumulatorem ACC,
- rejestrem B.

Dodatkowo używa rejestru – słowa statusowego PSW, do zapamiętywania rezultatów wykonywanych operacji.

Na wewnętrzną pamięć danych składają się dwa bloki: obszar 128 bajtów pamięci RAM oraz obszar 128 bajtów rejestrów specjalnych (SFR - ang. *Special Function Registers*) mikrokontrolera. Blok rejestrów specjalnych SFR znajduje się w niespójnym obszarze pamięci danych mikrokontrolera 8051.

Zerowanie mikrokontrolera (sygnał RST) powoduje ustawienie licznika rozkazów w stan 0000H, zatem początek programu musi być umieszczony pod adresem 0000H. Zwykle jest tam umieszczana instrukcja skoku do dalszego obszaru pamięci programu, ponieważ począwszy od adresu 0003H pierwszych kilkadziesiąt bajtów jest wykorzystywanych przez procedury obsługi przerwań.

Linie we/wy mikrokontrolera 8051 są pogrupowane w cztery 8-bitowe porty P0, P1, P2, P3. Wszystkie linie portów P0 ÷ P3, pracujące jako standardowe linie we/wy są niezależne pod względem kierunku przesyłania informacji. Rejestry P0 ÷ P3, kontrolujące stany poszczególnych linii, wchodzą w skład bloku rejestrów specjalnych, przy czym możliwe jest adresowanie ich poszczególnych bitów, co umożliwia bezpośrednie sterowanie pojedynczymi liniami portów o odpowiadających im nazwach.

1.4. Jednostka ALU

Jednostka ALU odpowiada za wykonanie operacji arytmetycznych, logicznych, a także za operacje przesuwania.

Do typowych rozkazów wykonywanych przez tę jednostkę należą :

a) działania arytmetyczne:

- dodawanie,
- odejmowanie,
- dodawanie i odejmowanie jedności,
- zmiana znaku liczby,
- porównywanie dwóch liczb;

b) działania logiczne:

- suma logiczna,
- iloczyn logiczny,
- suma modulo 2 (różnica symetryczna),
- negacja;

c) przesunięcie bitów w lewo lub prawo,

d) działania na bitach.

Podstawowymi elementami składowymi jednostki ALU są:

- 8-bitowa jednostka arytmetyczno-logiczna z układem korekcji dziesiętnej,
- rejestry pomocnicze używane przy wykonywaniu obliczeń (niedostępne dla programisty).

Z arytmometrem współpracują ponadto dwa rejestry bloku rejestrów specjalnych: akumulator (ACC) i rejestr B. Akumulator najczęściej zawiera jeden z operandów i zapisywany jest w nim wynik operacji, rejestr B jest natomiast wykorzystywany przy operacjach mnożenia i dzielenia. Argumentami operacji wykonywanych przez arytmometr mogą być również inne rejestry specjalne, komórki pamięci lub dane podawane bezpośrednio. Arytmometr może wykonywać operacje logiczne również na pojedynczych bitach. Dla tych

operacji akumulatorem jest bit przeniesienia C słowa stanu PSW. Rejestr specjalny PSW, którego zawartość pokazano w tabl. 2.2, bezpośrednio współpracuje z arytmometrem. Jego zawartość opisuje cechy wyniku ostatnio wykonywanej operacji.

Tablica 2.2. Słowo stanu PSW

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
C	AC	F0	RS1	RS0	OV	-	P

Znaczenie poszczególnych bitów jest następujące:

- P (ang. *parity*) – znacznik parzystości; ustawiony na 1 informuje o nieparzystej liczbie jedynek w akumulatorze; jest ustawiany po wykonaniu każdego rozkazu, w zależności od zawartości akumulatora;
- OV (ang. *overflow*) – znacznik nadmiaru dla dodawania i odejmowania w kodzie U2; dla mnożenia '1' wskazuje niezerową wartość rejestru B, dla dzielenia 1 oznacza wystąpienie dzielenia przez zero.
- RS0, RS1 – bity wyboru bloku rejestrów roboczych (ich znaczenie podano w tabl. 2.3),

Tablica 2.3. Znaczenie bitów RS0 i RS1

RS1	RS0	Wybrany blok
0	0	blok 0
0	1	blok 1
1	0	blok 2
1	1	blok 3

- F0 – znacznik uniwersalny do dowolnego wykorzystania,
- AC (ang. *auxiliary carry*) – znacznik przeniesienia połówkowego; ustawiany w przypadku wystąpienia przeniesienia z ACC.3 na ACC.4.
- C (ang. *carry*) – znacznik przeniesienia; ustawiany w przypadku wystąpienia przeniesienia z pozycji siódmego bitu akumulatora.

1.5. Pamięć danych

Jak wspomniano we wstępie do niniejszego rozdziału mikrokontroler 8051 dysponuje dwoma rodzajami pamięci:

- wewnętrzną — 128 bajtów,
- zewnętrzną — 64 kB.

Pamięć wewnętrzna

W pamięci RAM, czyli pamięci do zapisu i odczytu, są przechowywane informacje, będące obiektem bieżącego przetwarzania (dane, programy, wyniki). Pamięć RAM jest pamięcią ulotną, jej zawartość ginie po wyłączenia komputera.

Na wewnętrzną pamięć danych mikrokontrolera 8051, której mapę przedstawia tabl. 2.4, składają się dwa bloki: 128 bajtów ciągłego obszaru pamięci RAM oraz obszar 128 bajtów niespójnego bloku rejestrów specjalnych (SFR) mikrokontrolera.

Tablica 2.4. Mapa pamięci RAM mikrokontrolera 8051

Adresy	Obszar pamięci
Adresy 48 ÷ 27 (30H ÷ 7FH)	pamięć danych użytkownika
Adresy 32 ÷ 47 (20H ÷ 2FH)	pamięć adresowana bitowo (adresy 0 ÷ 127 (0H ÷ 7FH))
Adresy 24 ÷ 31 (18H ÷ 1FH)	rejestry R0 ÷ R7 - blok 3
Adresy 16 ÷ 23 (10H ÷ 17H)	rejestry R0 ÷ R7 - blok 2
Adresy 8 ÷ 15 (8H ÷ 0FH)	rejestry R ÷ R7 - blok 1
Adresy 0 ÷ 7 (0H ÷ 7H)	rejestry R0 ÷ R7 - blok 0

Obszar pamięci o adresach 0 ÷ 31 (0H ÷ 1FH) zajmują cztery banki rejestrów roboczych, zawierające po osiem 8-bitowych rejestrów w banku. Wybierane są bitami RS1 i RS0 rejestru PSW. Rejestry te mają oznaczenia R0 do R7 i mogą być używane do przechowywania danych. Wyjątek stanowią rejestry R0 i R1 każdego bloku, które mogą być wykorzystane dodatkowo do indeksowego adresowania wewnętrznej i zewnętrznej pamięci danych. Po wyzerowaniu mikrokontrolera sygnałem RST, bity RS1 i RS0 są ustawiane na 0, zostaje uaktywniony zerowy bank rejestrów (adresy 0 do 7).

Obszar pamięci o adresach 32 ÷ 47 (20H ÷ 2FH) może być używany do przechowywania dowolnych danych wykorzystywanych w programie. Cechą szczególną tego obszaru jest fakt, że możliwe jest zaadresowanie pojedynczego bitu komórki pamięci. Bity te są dostępne pod adresami 0 ÷ 27 (0H ÷ 7FH), a adres bitu, który chcemy zaadresować, można określić wzorem:

$$(\eta - 32) * 8 + i \qquad (2.1)$$

gdzie: η – adres słowa pamięci, i – numer bitu w tym słowie.

Ze wzoru wynika, że adresy te pokrywają się z adresami komórek pamięci, jednak nie stanowi to przeszkody, gdyż do pojedynczych bitów odwołują się specyficzne rozkazy mikrokontrolera, co pozwala jednoznacznie stwierdzić czy adresowana jest komórka, czy pojedynczy bit.

Obszar pamięci o adresach 48 ÷ 127 (30H ÷ 7FH) nie ma już żadnych specyficznych własności i jest używany jak zwykła pamięć o bajtowej organizacji. Mimo różnych własności poszczególnych obszarów, cała pamięć może być adresowana jak pamięć o bajtowej organizacji, w sposób bezpośredni lub indeksowy, przy użyciu rejestrów R0 i R1. W wewnętrznej pamięci danych jest umieszczany stos mikrokontrolera. Jest to obszar pamięci danych, z którego mikroprocesor korzysta zgodnie z zasadą LIFO (ang. *Last in first out*) – ostatnio zapisana dana jest odczytywana jako pierwsza. Obszar ten może być wykorzystywany do przechowywania danych (np. zawartości rejestru) lub też adresu instrukcji powrotu z podprogramu. Zapis danej na stos odbywa się w czasie wykonywania instrukcji `PUSH`, a odczyt przy wykonywaniu instrukcji `POP`.

Zachowywanie adresów powrotu z podprogramów odbywa się w chwili wykonania instrukcji wywołania podprogramu (np. `LCAU`), a odczyt przy wykonaniu instrukcji `RET`. Może on być umieszczony w dowolnym miejscu przez zapis odpowiedniej wartości adresu do rejestru SP, będącego wskaźnikiem stosu. Jednak po wyzerowaniu mikrokontrolera sygnałem RST, wskaźnik stosu przyjmuje wartość 7. Rejestr SP należy do bloku rejestrów specjalnych, a jego zawartość wskazuje na ostatnie zajęte przez stos miejsce pamięci. Wskaźnik stosu jest inkrementowany przed każdym zapisem danej na stos i dekrementowany po każdym odczycie.

Pamięć zewnętrzna

Oprócz pamięci wewnętrznej danych mikrokontroler może wykorzystywać zewnętrzną pamięć danych o pojemności do 64 kB. Ponieważ mikrokontroler nie ma osobnych rozkazów dotyczących dostępu do urządzeń we/wy, więc w obszarze adresowym 64kB zewnętrznej pamięci danych mogą być również umieszczane rejestry przyłączanych do systemu urządzeń we/wy.

1.6. Blok rejestrów specjalnych (SFR)

Blok rejestrów specjalnych SFR (ang. *Special Function Registers*) znajduje się w niespójnym obszarze pamięci danych mikrokontrolera o adresach 128 ÷ 40 (80H ÷ 0F0H). Obszar rejestrów SFR mikrokontrolera '51 jest wykorzystywany dwojako – z jednej strony są w nim umieszczone wszystkie (za wyjątkiem licznika rozkazów i czterech banków rejestrów

R0-R7) rejestry sterujące pracą mikrokontrolera lub używane bezpośrednio przy wykonywaniu programu; z drugiej zaś strony rejestry SFR stanowią rodzaj interfejsu pomiędzy mikroprocesorem, a układami peryferyjnymi (takimi jak sterownik łącza szeregowego) umieszczonymi wewnątrz mikrokontrolera. Wszystkie operacje sterowania wewnętrznymi układami peryferyjnymi oraz przesyłania danych między nimi a CPU, odbywają się właśnie za pośrednictwem rejestrów SFR. Dostęp do każdego z tych rejestrów jest możliwy wyłącznie w trybie adresowania bezpośredniego (np. `MOV P1,A`, gdzie P1 jest wskazywany przez adres 144 (90H)). Nazwy rejestrów SFR są zwykle nazwami predefiniowanymi w asemblerach mikrokontrolerów rodziny '51, więc najczęściej nie trzeba znać adresu danego rejestru — wystarczy pamiętać jego nazwę. W tablicy 2.5 przedstawiono wszystkie rejestry SFR mikrokontrolera wraz z ich adresami i opisami pełnionej funkcji.

Tablica 2.5. Blok rejestrów specjalnych mikrokontrolera 8051

Nazwa	Adres	Pełniona funkcja
P0	128 (80H)	port 0 we/wy
SP	129 (81H)	wskaźnik stosu
DPL	130 (82H)	rejestr indeksowy DPTR (mniej znaczący bajt)
DPH	131 (83H)	rejestr indeksowy DPTR (bardziej znaczący bajt)
PCON	135 (87H)	rejestr sterujący stanami uśpienia
TCON	136 (88H)	rejestr sterujący układów czasowych 0 i 1
TMOD	137 (89H)	rejestr trybu pracy układów czasowych 0 i 1
TL0	138 (8AH)	rejestr danych układu czasowego 0 (mniej znaczący)
TL1	139 (8BH)	rejestr danych układu czasowego 1 (mniej znaczący)
TH0	140 (8CH)	rejestr danych układu czasowego 0 (bardziej znaczący)
TH1	141 (8DH)	rejestr danych układu czasowego 1 (bardziej znaczący)
P1	144 (90H)	port 1 we/wy
SCON	152 (98H)	rejestr sterujący układu transmisji szeregowej
SBUF	153 (99H)	rejestr danych układu transmisji szeregowej
P2	160 (0A0H)	port 2 we/wy
IE	168 (0A8H)	rejestr maski przerwań
P3	176 (0B0H)	port 3 we/wy
IP	184 (0B8H)	rejestr priorytetów przerwań
PSW	208 (0D0H)	słowo stanu procesora
ACC	224 (0E0H)	akumulator
B	240 (0F0H)	rejestr ogólnego przeznaczenia

Wszystkie rejestry umieszczone pod adresami podzielnymi przez 8 są dostępne także bitowo. Adres określonego bitu można określić posługując się zależnością:

$$\text{adres} = m + n \tag{2.2}$$

gdzie: m – jest adresem rejestru SFR, n – jest numerem adresowanego bitu (licząc od 0).

W tablicy 2.6 pokazano wszystkie rejestry SFR mikrokontrolera 8051, które są dostępne bitowo.

Tablica 2.6. Rejestry SFR adresowalne bitowo; nazwy i adresy poszczególnych bitów

SFR	Bit 7		Bit 6		Bit 5		Bit 4		Bit 3		Bit 2		Bit 1		Bit 0	
	nazwa	adres	nazwa	adres	nazwa	adres	nazwa	adres	nazwa	adres	nazwa	adres	nazwa	adres	nazwa	adres
P0	P0.7	135	P0.6	134	P0.5	133	P0.4	132	P0.3	131	P0.2	130	P0.1	129	P0.0	128
TCON	TF1	143	TR1	142	TF0	141	TR0	140	IE1	139	IT1	138	IE0	137	IT0	136
P1	P1.7	151	P1.6	150	P1.5	149	P1.4	148	P1.3	147	P1.2	146	P1.1	145	P1.0	144
SCON	SM0	159	SM1	158	SM2	157	REN	156	TB8	155	RB8	154	TI	153	RI	152
P2	P2.7	167	P2.6	166	P2.5	165	P2.4	164	P2.3	163	P2.2	162	P2.1	161	P2.0	160
IE	EA	175	-	174	ET2	173	ES	172	ET1	171	EX1	170	ET0	169	EX	168
P3	P3.7	183	P3.6	182	P3.5	181	P3.4	180	P3.3	179	P3.2	178	P3.1	177	P3.0	176
IP	-	191	-	190	PT2	189	PS	188	PT1	187	PX1	186	PT0	185	PX0	184
PSW	C	215	AC	214	F0	213	RS1	212	RS0	211	OV	210	-	209	P	208
ACC	ACC.7	231	ACC.6	230	ACC.5	229	ACC.4	228	ACC.3	227	ACC.2	226	ACC.1	225	ACC.0	224
B	B.7	247	B.6	246	B.5	245	B.4	244	B.3	243	B.2	242	B.1	241	B.0	240

Te spośród rejestrów SFR, które nie są aktualnie używane do sterowania zasobami mikrokontrolera, można wykorzystać jako pamięć o dostępie bezpośrednim, jednak pod warunkiem, że zmiana zawartości danego rejestru nie wpłynie na pracę programu i funkcjonowanie mikrokontrolera.

1.7. Pamięć programu

Jak wspomniano, mikrokontroler 8051 ma 4 kB wewnętrznej pamięci programu typu ROM, programowanej maską. Niektóre wersje układu mają pamięć EPROM lub EEPROM. Mikrokontroler 8051 może także korzystać z zewnętrznej pamięci programu o pojemności do 64 kB. To, z której z tych pamięci są pobierane rozkazy, zależy od stanu wyprowadzenia EA' oraz aktualnego adresu w liczniku rozkazów.

W przypadku korzystania z wewnętrznej pamięci programu, wyprowadzenie EA' musi być ustawione w stan wysoki. Jeśli pojemność wewnętrznej pamięci programu jest niewystarczająca, to część programu może być umieszczona w pamięci zewnętrznej. Dopóki wartość licznika rozkazów nie przekracza rozmiaru wewnętrznej pamięci programu, dopóty rozkazy są pobierane z pamięci wewnętrznej. Przekroczenie przez licznik rozkazów wartości 0FFFH powoduje pobieranie rozkazów z zewnętrznej pamięci programu. Jeśli mikrokontroler ma korzystać wyłącznie z zewnętrznej pamięci programu, to wyprowadzenie EA' musi być ustawione w stan niski. W większości mikrokontrolerów rodziny '51 wyprowadzenie EA' nie ma wewnętrznego rezystora polaryzującego i nie może pozostawać nie przyłączone.

Zerowanie (podane na wyprowadzenia RST jest stanem jedynki logicznej) mikrokontrolera powoduje ustawienie licznika rozkazów w stan 0000H, z tego względu początek programu (pierwsza instrukcja programu) musi być umieszczony pod adresem 0000H. Zwykle pod tym adresem umieszczana jest instrukcja skoku do dalszego obszaru pamięci programu, ponieważ począwszy od adresu 0003H pierwszych kilkadziesiąt bajtów jest wykorzystywanych przez procedury obsługi przerwań. W tablicy 2.7 umieszczono adresy procedur obsługi przerwań wraz z ich zwięzłym opisem.

Tablica 2.7. Umieszczenie procedur obsługi przerwań mikrokontrolera 8051 w pamięci programu.

Adres	Zawartość
0003H	początek procedury obsługi przerwania zewnętrznego INT0'
000BH	początek procedury obsługi przerwania z układu czasowego T0
0013H	początek procedury obsługi przerwania zewnętrznego INT1'
001BH	początek procedury obsługi przerwania z układu czasowego T1
0023H	początek procedury obsługi przerwania z układu transmisji szeregowej

Rejestr PC zawiera adres aktualnego rozkazu przeznaczonego do wykonania. Rozkaz jest pobierany z pamięci programu do rejestru rozkazów (rejestr ten nie jest dostępny programowo). Na podstawie zawartości rejestru rozkazów, dekoder rozkazów steruje wyborem źródła argumentu, miejsca umieszczenia wyniku, funkcjami arytmometru itp. – w ten sposób mikroprocesor wykonuje zadaną operację. Jeżeli nie jest wykonywany rozkaz skoku, to zawartość licznika rozkazów jest inkrementowana po odczycie każdego bajtu z pamięci programu.

1.8. Porty mikrokontrolera

Linie wejścia/wyjscia mikrokontrolera 8051 są pogrupowane w cztery 8-bitowe porty:

- P0 – linie P0_0/AD0 - P0_7/AD7,

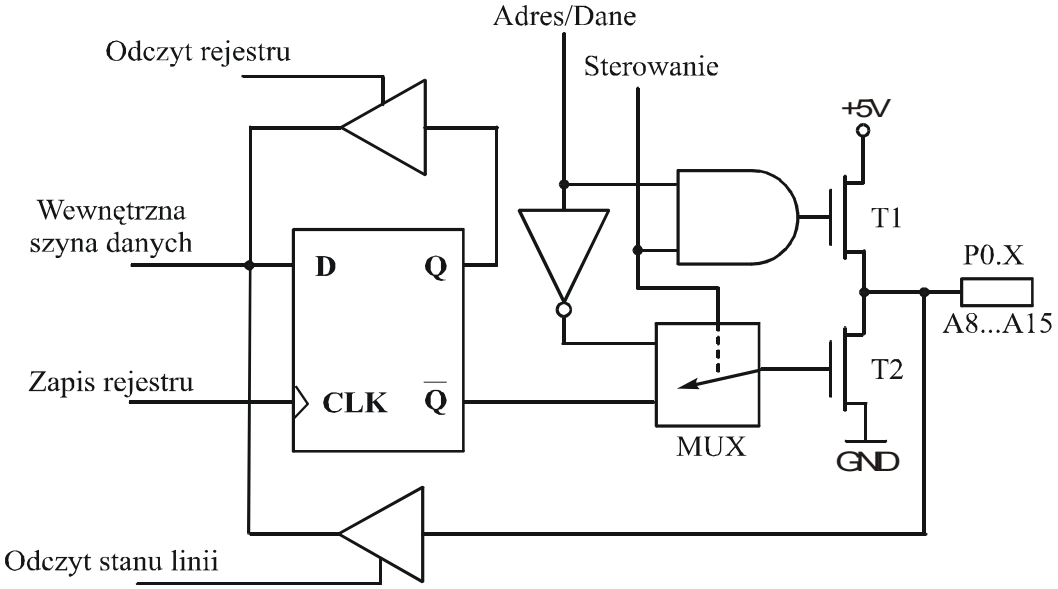
- P1 – linie P1_0 - P1_7,
- P2 – linie P2_0/A8 - P2_7/A15,
- P3 – linie P3_0/RxD', P3_1/TxD', P3_2/INT0', P3_3/INT1', P3_4/T0, P3_5/T1, P3_6/WR', P3_7/RD'.

Wszystkie linie portów P0-P3, pracujące jako standardowe linie we/wy są niezależne pod względem kierunku przesyłania informacji. Jak wspomniano, rejestry P0-P3 wchodzą w skład bloku rejestrów specjalnych, przy czym możliwe jest adresowanie ich poszczególnych bitów, co umożliwia bezpośrednie sterowanie pojedynczymi liniami we/wy.

Funkcje i struktura linii we/wy są odmienne dla każdego z układów P0-P3.

1.8.1. Struktura portu P0

Port PO składa się z 8 jednakowych układów dostępnych poprzez blok rejestrów specjalnych. Strukturę jednego z nich pokazano na rys. 2.3.



Rys. 2.3. Struktura pojedynczej linii portu P0

W czasie normalnej pracy jako we/wy (stan 0 na linii sterowanie), tranzystor T1 jest wyłączony, a stan tranzystora T2 zależy bezpośrednio od stanu przerzutnika D, który wchodzi w skład rejestru P0 bloku SFR. Jeżeli w przerzutniku zapisana jest jedynka, to tranzystor T2 jest wyłączony i wyjście układu przechodzi w stan wysokiej impedancji. W przypadku wpisania zera, tranzystor T2 jest włączony i na wyjściu panuje stan niski. Wyjścia układu P0, podczas wyprowadzania informacji, zachowują się jak linie typu otwarty dren. Z tego względu, w celu uzyskania wysokiego poziomu napięcia, odpowiadającego jedynce wpisanej w przerzutnik, należy do linii przyłączyć zewnętrzny rezystor polaryzujący. Podczas wprowadzania informacji stan linii zewnętrznej jest wprowadzany poprzez bufor na wewnętrzną szynę danych, jednak aby odczyt linii portu był możliwy, wcześniej w przerzutnik musi zostać wpisana jedynka.

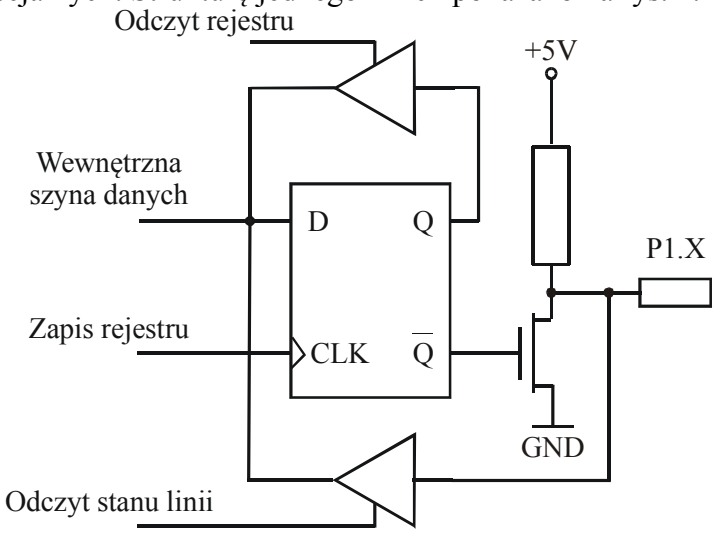
Istnieje grupa rozkazów mikrokontrolera, które odczytują, modyfikują i ponownie zapisują zawartość rejestru P0 (P1, P2, P3). Są to rozkazy typu RMW (ang. *Read-Modify-Write*). Poszczególne bity rejestru są odczytywane przez bufory dołączone do wyjścia Q przerzutnika, ich wartość jest przekazywana do arytmometru, gdzie następuje wykonanie operacji, a następnie nowa wartość jest ponownie wprowadzana do przerzutnika. Port P0, oprócz funkcji we/wy, odgrywa rolę szyny danych multipleksowanej z mniej znaczącą częścią szyny adresowej, podczas realizacji dostępu do pamięci zewnętrznej. Wtedy stan linii sterującej S = 1 i tranzystory T1 i T2 są sterowane wewnętrzną linią adresu/danych. Poziom niski

włącza T2 i wyłącza T1, co powoduje wymuszenie stanu "0" na wyjściu układu; poziom wysoki wyłącza T2 i włącza T1, powodując stan wysoki na wyjściu – nie jest przy tym konieczne instalowanie rezystora polaryzującego. Dane z szyny danych są odczytywane w taki sam sposób, jak przy pracy portu jako we/wy.

Wykonanie dostępu do pamięci zewnętrznej powoduje zapisanie samych jedynek w rejestr P0, niszcząc tym samym przechowywaną w nim informację. Z tego powodu korzystanie z pamięci zewnętrznej wyklucza w zasadzie możliwość korzystania z portu P0 jako 8 linii we/wy. Każdą linię portu P0 można obciążyć ośmioma wejściami typu TTL-LS.

1.8.2. Struktura portu P1

Port P1 składa się z 8 jednakowych układów, dostępnych poprzez blok rejestrów specjalnych. Strukturę jednego z nich pokazano na rys. 2.4.

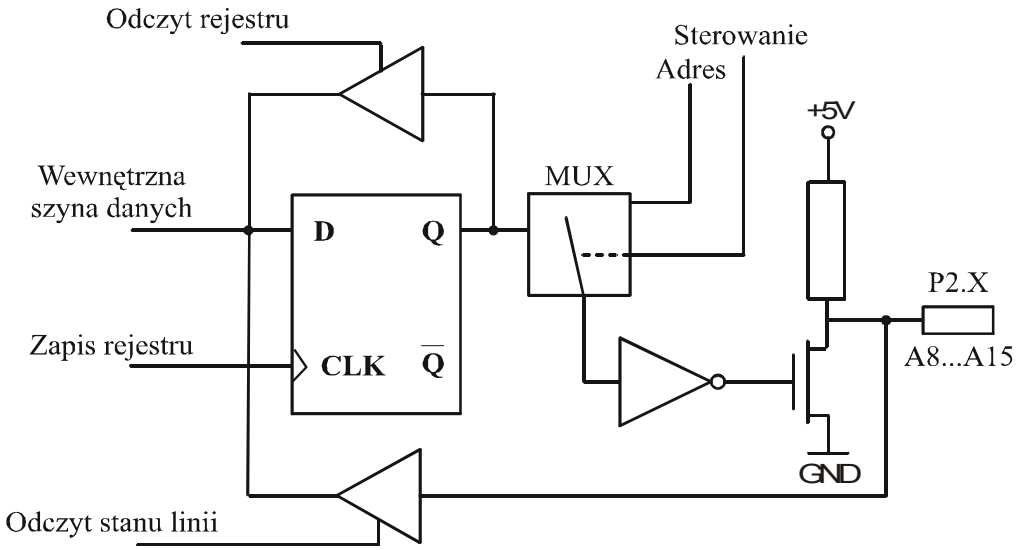


Rys. 2.4. Struktura pojedynczej linii portu P1

Przerzutnik D, będący głównym elementem układu, wchodzi w skład rejestru P1, umieszczonego w pamięci wewnętrznej RAM (blok SFR). Modyfikacji stanu przerzutnika D można dokonać rozkazami zapisu lub RMW, tak jak w przypadku portu P0. Przerzutnik D steruje bezpośrednio tranzystorem wyjściowym. Rezystor zapewnia ustalenie wysokiego poziomu napięcia przy wyłączonym tranzystorze (jedynka w przerzutniku). Podczas wprowadzania informacji rozkazami odczytu linii zewnętrznej, wymagane jest zapisanie jedynki w przerzutniku D. Linie portu P1 nie pełnią żadnych dodatkowych funkcji. Można je obciążać czterema wejściami TTL-LS.

1.8.3. Struktura portu P2

Port P2 składa się z 8 jednakowych układów, dostępnych poprzez blok rejestrów specjalnych. Strukturę jednego z nich pokazano na rys. 2.5.

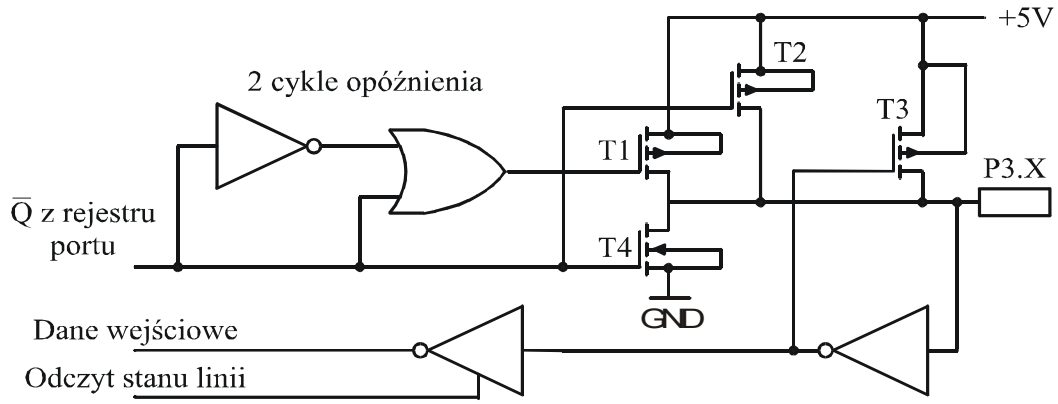


Rys. 2.5. Struktura pojedynczej linii portu P2

Zasada działania linii portu P2 jest taka sama jak portu P1, pod warunkiem, że dodatkowe funkcje tego układu nie są wykorzystywane (stan niski na linii sterowanie). Ponadto, podczas dostępu do pamięci zewnętrznej, port P2 pełni funkcję bardziej znaczącej części szyny adresowej (A8-A15). Wtedy, za pomocą linii S = 1 (sterowanie), zostaje przełączony multiplekser i tranzystor wyjściowy jest sterowany z wewnętrznej linii adresu. Zawartość przerzutnika D nie jest niszczona i zapisany w nim stan pojawia się na wyjściu portu po zakończeniu cyklu dostępu do pamięci zewnętrznej. W przypadku realizacji dostępu do zewnętrznej pamięci danych przy użyciu rejestrów indeksowych R0 i R1 (np. przy wykorzystaniu instrukcji `movx @RO, A`) mikroprocesor ustawia tylko młodsze osiem bitów adresu, a stan portu P2 jest wtedy ustawiony zgodnie z zawartością rejestru P2. Linie portu P2 można obciążać czterema wejściami TTL-LS.

2.8.4. Struktura portu P3

W porcie P3 wszystkie linie pełnią dodatkowe funkcje, jednak w podstawowym trybie pracy jako we/wy działanie portu P3 nie różni się w niczym od działania portów P1 i P2. W zależności od tego, czy realizowana funkcja dodatkowa jest wyjściem (np. TxD, RD', WR') czy wejściem (np. INTO'), czy pełni obie funkcje (RxD), różna jest struktura linii portu. Wykorzystanie dodatkowej funkcji linii jest możliwe jedynie po zapisaniu jedynki w przerzutniku D. W przypadku wyjść (P3_1/TxD', P3_6/WR', P3_7/RD') tranzystor T jest sterowany odpowiednim sygnałem wewnętrznym, realizującym daną funkcję. Do realizacji funkcji będącej wyjściem, stan wyprowadzenia zewnętrznego jest podawany za pośrednictwem stale otwartego bufora na wewnętrzną linię mikrokontrolera. Linie portu P3 można obciążać czterema wejściami TTL-LS. Tranzystor. symbolizujący stopień wyjściowy linii portów, jest w rzeczywistości bardziej skomplikowaną strukturą, której schemat pokazano na rys. 2.6.



Rys. 2.6. Struktura pojedynczej linii portu P3

Zapis do przerzutnika następuje w ostatniej fazie cyklu maszynowego, a wpisana wartość pojawia się na wyjściu linii portu w pierwszej fazie następnego cyklu maszynowego – wyjściowy bufor linii portu sprawdza stan przerzutnika tylko podczas pierwszej fazy każdego cyklu zegarowego, a wykryty stan utrzymuje przez czas trwania drugiej fazy. Zmiana zawartości przerzutnika ze stanu niskiego na wysoki powoduje włączenie tranzystora T1, który pozostaje aktywny przez dwa cykle zegarowe. Przez złącze dren – źródło (DS) tego tranzystora może przepływać prąd o dużej wartości, co powoduje, że zachowuje się on jak rezystor polaryzujący o małej wartości rezystancji, a jego włączenie ma na celu przyspieszenie przejścia ze stanu logicznego 0 do 1. Wpisanie do przerzutnika 1 powoduje dodatkowo włączenie tranzystora T2, który pozostaje włączony do chwili zmiany zawartości przerzutnika na 0. Tranzystor T2 odgrywa rolę rezystora polaryzującego o dużej wartości rezystancji i ma za zadanie utrzymanie linii w stanie wysokim. Włączenie tranzystora T1 powoduje włączenie T3, który także pełni funkcję rezystora polaryzującego o dużej rezystancji. Tranzystor T3 jest jednak aktywny tylko wtedy, gdy napięcie na linii portu jest wyższe niż 1 ÷ 1,5 V. Jeżeli na linii portu zostanie wymuszony stan niski, to T3 jest wyłączony i jedynym obciążeniem pozostaje tranzystor T2, co wpływa korzystnie na zmniejszenie strat mocy w układzie. Jeżeli zaś linia pracuje jako wyjście, to jednoczesna praca tranzystorów T2 i T3 powoduje zwiększenie wydajności prądowej wyjścia. Jeżeli do przerzutnika zostaje wpisane 0, to jest aktywny tylko tranzystor T4.

1.9. Układy czasowo licznikowe – timery

Mikrokontroler 8051 jest wyposażony w dwa układy licznikowe T0 i T1. Każdy z liczników składa się z dwóch 8-bitowych połówek. Połówki te są widziane przez mikroprocesor jako rejestry specjalne TH0 i TL0 dla układu licznikowego T0 oraz TH1 i TL1 dla układu licznikowego T1. Każdy z obu układów może pracować jako licznik (zlicza wówczas impulsy zewnętrzne) lub jako czasomierz – *timer* (zlicza cykle maszynowe mikrokontrolera). Układy licznikowe mogą pracować w trybach 0, 1, 2 i 3. Wybór trybu pracy i sterowanie zliczaniem odbywa się za pośrednictwem rejestrów specjalnych: TCON i TMOD, których zawartość została przedstawiona w tabl. 2.8.

Tablica 2.8. Rejestry sterujące pracą układów licznikowych

Rejestr:	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TCON	TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0
TMOD	GATE	C/T'	M1	M0	GATE	C/T'	M1	M0

1.9.1. Rejestr TCON

Rejestr TCON składa się z 8 bitów opisanych poniżej:

TF1 – wskaźnik przerwania od licznika T1; bit ustawiany sprzętowo wskutek przepełnienia licznika T1, zerowany sprzętowo w chwili rozpoczęcia wykonywania procedury obsługi przerwania; możliwe programowe wywołanie przerwania;
TR1 – bit włączający (TR1 = 1) układ licznikowy T1;
TF0 – wskaźnik przerwania od licznika T0; bit ustawiany sprzętowo wskutek przepełnienia licznika T0, zerowany sprzętowo w chwili rozpoczęcia wykonywania procedury obsługi przerwania; możliwe programowe wywołanie przerwania;
TR0 – bit włączający (TR0 = 1) układ licznikowy T0;
IE1 – wskaźnik przerwania zewnętrznego $\overline{\text{INT1}}$; ustawiany sprzętowo w wyniku wykrycia opadającego zbocza (lub niskiego poziomu w zależności od trybu pracy) na wyprowadzeniu $\overline{\text{INT1}}$ mikrokontrolera; zerowany sprzętowo w wyniku rozpoczęcia wykonywania procedury obsługi przerwania (tylko jeśli przerwanie aktywowane jest zboczem); jeśli przerwanie jest aktywowane poziomem, to stan bitu odpowiada stanowi na wyprowadzeniu $\overline{\text{INT1}}$ mikrokontrolera;
IT1 – bit określający stan wyprowadzenia $\overline{\text{INT1}}$ aktywujący przerwanie: 0 - przerwanie aktywowane niskim poziomem, 1 - przerwanie aktywowane opadającym zboczem;
IE0 – jak IE1, dotyczy wyprowadzenia $\overline{\text{INT0}}$;
IT0 – jak IT1, dotyczy wyprowadzenia $\overline{\text{INT0}}$.

1.9.2. Rejestr TMOD

Rejestr jest podzielony na dwie 4-bitowe części zawierające bity o jednakowym znaczeniu. Cztery starsze bity rejestru dotyczą układu licznikowego T1, cztery młodsze natomiast dotyczą układu T0. Znaczenie bitów jest następujące:

GATE – bit sterujący bramkowaniem licznika; dla GATE = 1 zliczanie następuje, gdy sygnał $\overline{\text{INTn}}$ i bit TRn, odpowiadające danemu licznikowi, są w stanie wysokim; jeżeli GATE = 0, to zliczanie następuje, gdy bit TRn danego licznika jest ustawiony;

C/$\overline{\text{T}}$ – bit określający źródło impulsów zliczanych przez dany układ licznikowy: "0" oznacza zliczanie cykli maszynowych (impulsów o częstotliwości $f_{osc}/12$), 1 oznacza zliczanie impulsów zewnętrznych (z wyprowadzenia Tn odpowiadającego danemu licznikowi);

M1, M0 – bity wyboru trybu pracy danego układu licznikowego, o znaczeniu pokazanym w tabl. 2.9.

Tablica 2.9. Znaczenie bitów M1 i M0 rejestru TMOD

M1	M0	Tryb pracy:
0	0	tryb 0
0	1	tryb 1
1	0	tryb 2
1	1	układ T0 - tryb 3, układ T1 zatrzymany

Gdy licznik pracuje w trybie czasomierza, inkrementacja licznika następuje w każdym kolejnym cyklu maszynowym. W przypadku wykorzystywania układu w trybie licznika, jego zawartość jest zwiększana w odpowiedzi na opadające zbocze sygnału wejściowego. Detekcja zbocza jest jednak zsynchronizowana z cyklem pracy mikrokontrolera, przez testowanie stanu odpowiedniej linii wejściowej w każdym cyklu maszynowym. Jeżeli w pierwszym cyklu wykryto stan wysoki, a w następnym stan niski, to następuje zwiększenie zawartości licznika. Aby każde zbocze opadające zostało wykryte, żaden stan na linii wejściowej nie może trwać krócej niż czas trwania jednego cyklu maszynowego, zatem maksymalna częstotliwość

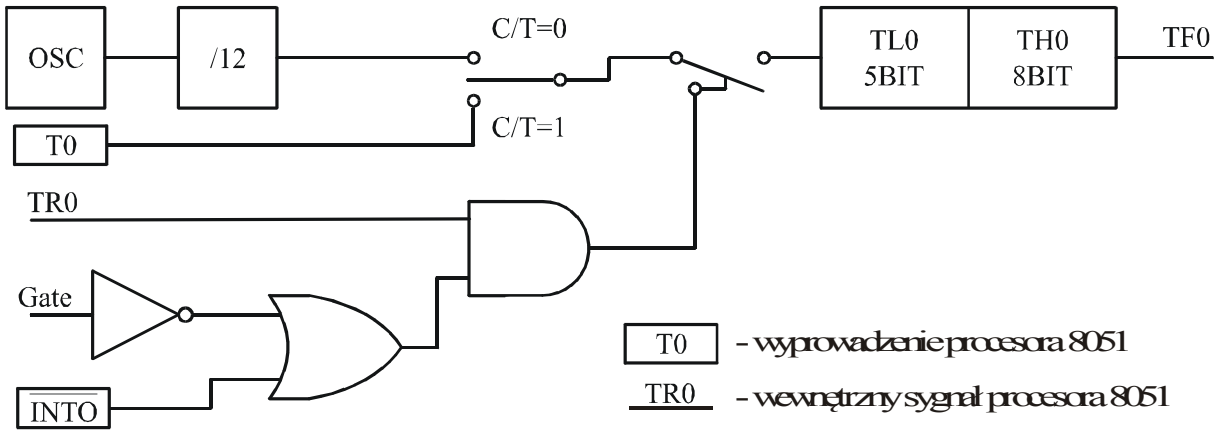
sygnału o wypełnieniu 50% jest ograniczona do wartości 1/24 f_{osc}, gdzie f_{osc} jest częstotliwością zegara mikrokontrolera.

1.9.3. Tryby pracy układów licznikowych

Tryby pracy układów licznikowych mikrokontrolera 8051 mają następujące właściwości:

Tryb 0

Tryb pracy 0 jest identyczny dla obu liczników. Na rysunku 2.7 przedstawiono strukturę logiczną licznika T0 pracującego w trybie 0 (analogiczna jest struktura licznika T1 w trybie 0).

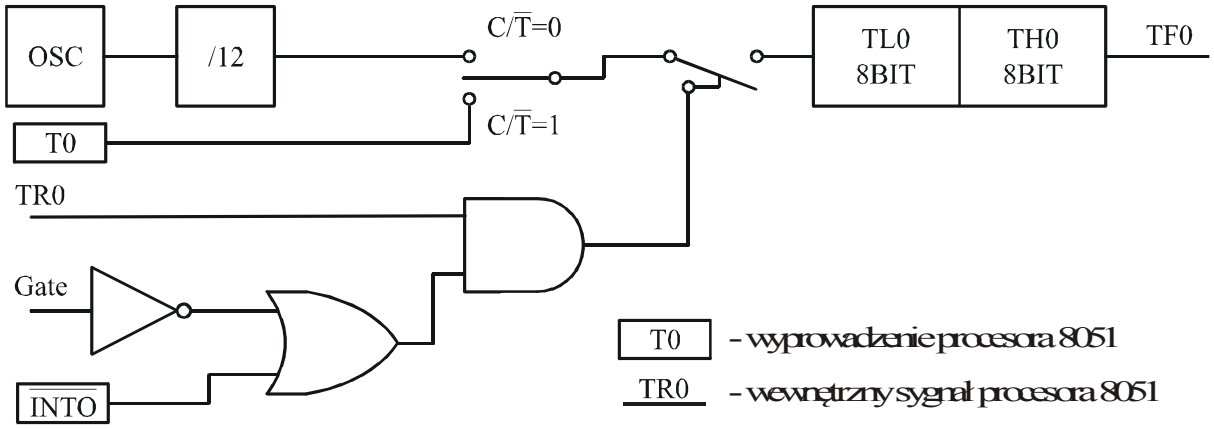


Rys. 2.7. Układ pracy licznika T0 w trybie 0

Stan licznika jest określony zawartością odpowiedniego rejestru THn oraz pięciu młodszych bitów rejestru TLn. Przepełnienie takiego 13-bitowego licznika spowoduje ustawienie odpowiedniego bitu przerwania TFn. Przebieg zliczania i źródło impulsów są określane za pomocą odpowiednich ustawień bitów sterujących pracą liczników (TCON, TMOD).

Tryb 1

Tryb 1 jest identyczny dla obu liczników. Pod względem logicznym jest identyczny z trybem 0. Różnica polega na długości licznika: w trybie 1 zliczanie następuje w 16-bitowym liczniku złożonym z rejestrów TLn i THn odpowiedniego układu licznikowego. Na rysunku 2.8 przedstawiono dla przykładu strukturę licznika T0 pracującego w trybie 1.

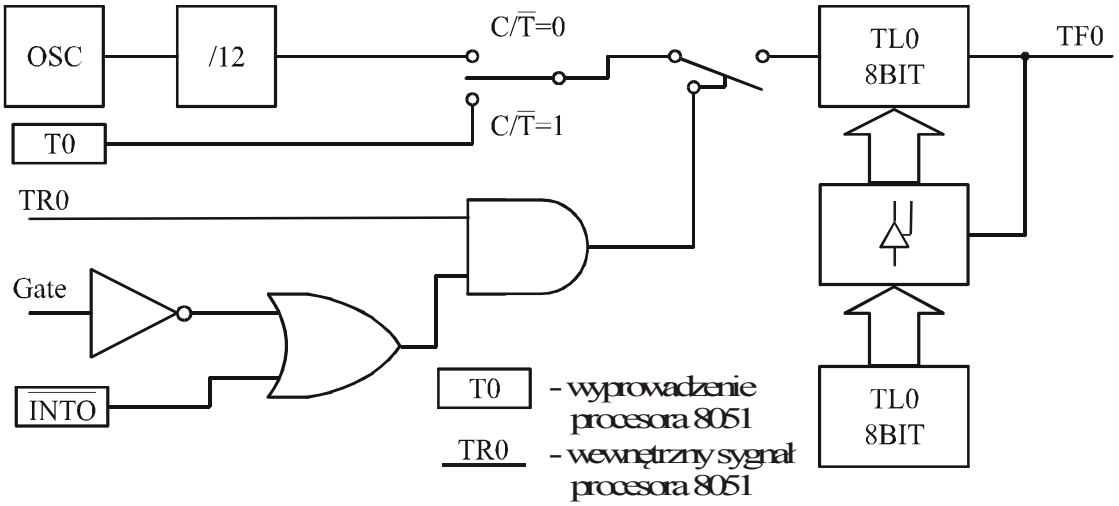


Rys. 2.8. Układ pracy licznika T0 w trybie 1

Tryb 2

Tryb 2 jest identyczny dla obu liczników. W tym trybie liczniki pracują jako 8-bitowe, z automatycznym przeładowywaniem. Przepełnienie licznika TLn nie tylko ustawia odpowiedni znacznik przerwania TFn, lecz również powoduje przepisanie zawartości THn do odpowiedniego rejestru TLn. Czynność ta nie powoduje zmiany zawartości rejestru THn.

Na rysunku 2.9 przedstawiono dla przykładu strukturę licznika T0 pracującego w trybie 2.

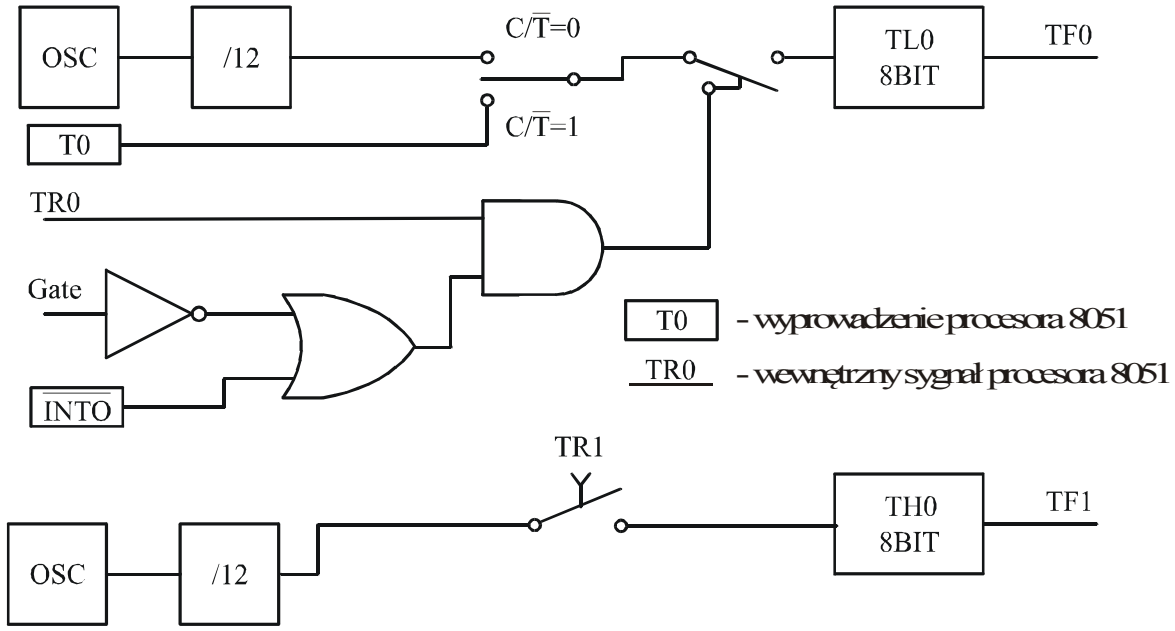


Rys. 2.9. Układ pracy licznika T0 w trybie 2

Tryb 3

Tryb 3 jest jedynym trybem różnym dla układów licznikowych T1 i T0. Schemat logiczny układu licznikowego T0 pracującego w trybie 3 przedstawiono na rys. 2.10.

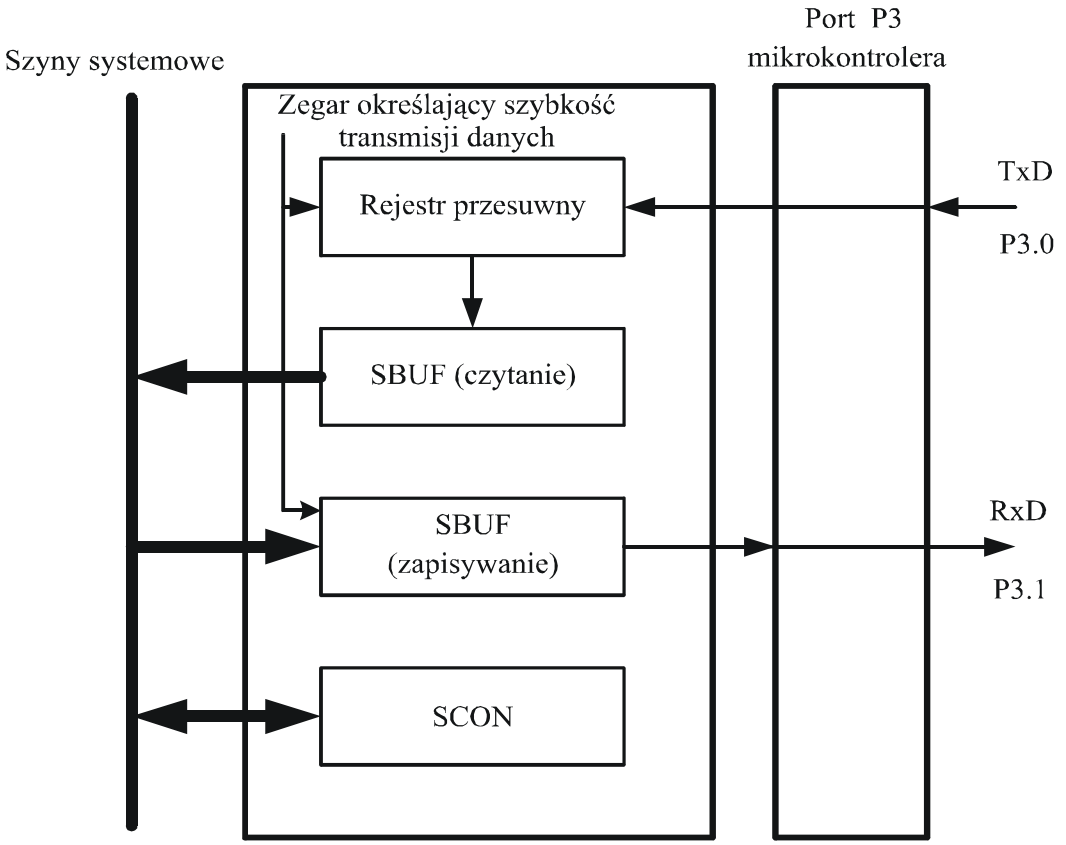
W trybie 3 może pracować tylko układ T0. Wprowadzenie układu T1 w tryb 3 powoduje jego zatrzymanie. W tym trybie układ T0 składa się z dwu niezależnych liczników 8-bitowych, utworzonych z rejestrów TL0 i TH0. Rejestr TL0 pracuje w układzie identycznym z trybami 0 i 1, z tą różnicą, że zliczanie następuje w liczniku 8-bitowym. Rejestr TH0 pracuje również jako licznik 8-bitowy, może on jednak zliczać wyłącznie impulsy o czasie trwania równym cyklom maszynowym mikrokontrolera. Do sterowania zliczaniem TH0 wykorzystano bity sterujące TR1 i TF1 układu czasowego T1. Wprowadzenie układu T0 w tryb 3 ogranicza zatem wykorzystanie układu T1 do zastosowań nie wymagających wykorzystania przerwania (wyłączenie licznika T1 następuje w wyniku wprowadzenia go w jego własny tryb 3) i wówczas układ T1 jest wykorzystywany do generacji sygnału taktującego w układzie transmisji szeregowej.



Rys. 2.10. Układ pracy licznika T0 w trybie 3

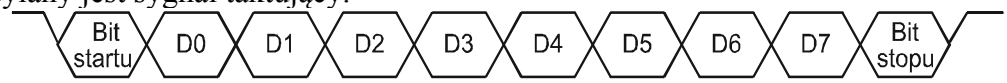
1.10. Układ transmisji szeregowej

Łącze szeregowe mikrokontrolera 8051 umożliwia prowadzenie synchronicznej lub asynchronicznej transmisji danych. Transmisja asynchroniczna jest transmisją *full-duplex*, natomiast synchroniczna jest transmisją *half-duplex*. Jako przykład przedstawiono na rys. 2.11 urządzenie transmisji asynchronicznej mikrokontrolera 8051. Układ odbiornika ma bufor odbiorczy, co pozwala na realizację procesu odbierania kolejnej danej przed pobraniem przez mikroprocesor danej już odebranej. Jest to jednak bufor jednobajtowy, jeśli dana nie zostanie odczytana przez mikroprocesor przed końcem kolejnej transmisji, odebrany wcześniej bajt zostanie utracony. Podczas realizacji transmisji asynchronicznej nadawane dane są wysyłane linią TxD, zaś odbierane linią RxD. Fizycznie bufor nadawczy i odbiorczy stanowią dwa osobne rejestry pod wspólna nazwą, operacja zapisu dotyczy więc innego rejestru niż operacja czytania.



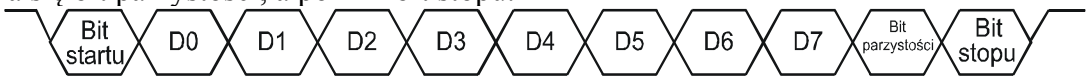
Rys. 2.11. Schemat budowy układu transmisji szeregowej mikrokontrolera 8051

Podczas transmisji synchronicznej dane są odbierane i nadawane po linii RxD, a na linię TxD wysyłany jest sygnał taktujący.



Rys. 2.12. Format danych dla transmisji asynchronicznej

Format danych w standardzie transmisji asynchronicznej pokazano na rys. 2.12. Transmisja rozpoczyna się od bitu startu, po którym następuje 8 bitów danych i jeden bit stopu. Format danych może wyglądać również jak na rys. 2.13, gdzie po 8 bitach danych pojawia się bit parzystości, a po nim bit stopu.



Rys. 2.13. Format danych dla transmisji asynchronicznej z bitem parzystości

Do konfiguracji pracy układu transmisji szeregowej służy rejestr SCON z bloku rejestrów specjalnych, którego rozmieszczenie bitów pokazano w tabl. 2.10.

Tablica 2.10. Rozmieszczenie bitów rejestru SCON

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
SM0	SM1	SM2	REN	TB8	RB8	TI	RI

Dwa bity o oznaczeniach SM0, SM1 określają tryb pracy układu transmisji szeregowej. Ich znaczenie pokazano w tabl. 2.11.

Tablica 2.11. Konfiguracja trybów pracy układu transmisji szeregowej.

SM0	SM1	Tryb pracy układu
0	0	tryb 0, transmisja synchroniczna, prędkość $f_{osc}/12$
0	1	tryb 1, transmisja asynchroniczna, 8 bit, prędkość Timer1
1	0	tryb 2, transmisja asynchroniczna, 9 bitów, prędkość $f_{osc}/64$ lub $f_{osc}/32$

1	1	tryb 3, transmisja asynchroniczna, 9 bitów, prędkość Timer1

Bit o nazwie SM2 steruje przejściem w tryb komunikacji wieloprocesorowej. W transmisji synchronicznej zaleca się aby SM2 = 0. W transmisji asynchronicznej wartość SM2 = 1 powoduje ustawienie wskaźnika przerwania RI tylko wtedy, jeżeli bit stopu (transmisja 8-bitowa) lub dziewiąty bit (transmisja 9-bitowa) jest równy jeden. Znaczenie pozostałych bitów jest następujące:

REN – bit włączający odbiornik: 1 – włączenie, 0 – wyłączenie;

TB8 – dziewiąty bit danych nadawanych podczas transmisji 9-bitowej;

RB8 – dziewiąty bit danych odbieranych podczas transmisji 9-bitowej;

TI – wskaźnik przerwania nadajnika; informuje o zakończeniu transmisji danej i gotowości nadajnika do przesyłania następnej; bit musi być zerowany programowo przez procedurę obsługi przerwania;

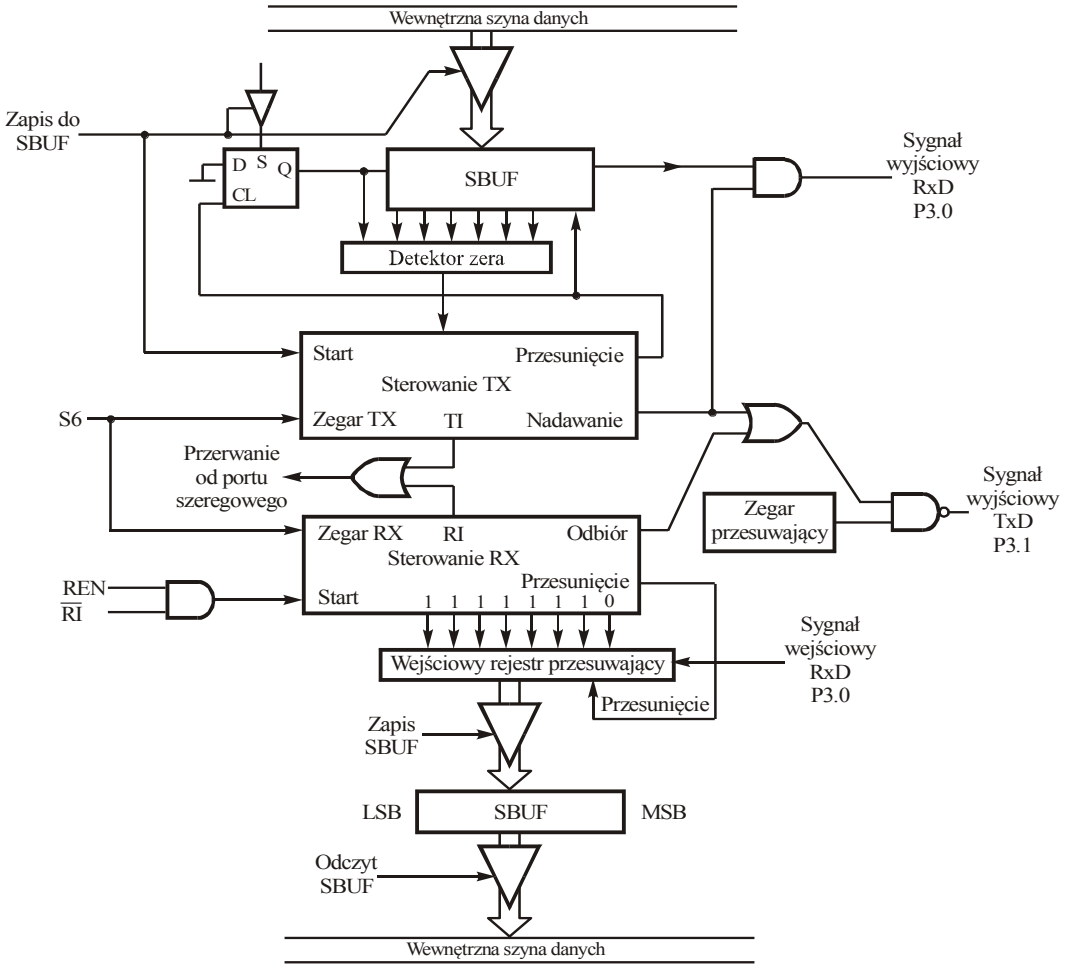
RI – wskaźnik przerwania odbiornika; informuje o właściwym odbiorze danej i jej gotowości do pobrania przez mikroprocesor; bit musi być zerowany programowo przez procedurę obsługi przerwania.

1.10.1. Tryb 0 pracy układu transmisji szeregowej

W trybie 0 układ transmisji szeregowej ma następujące cechy:

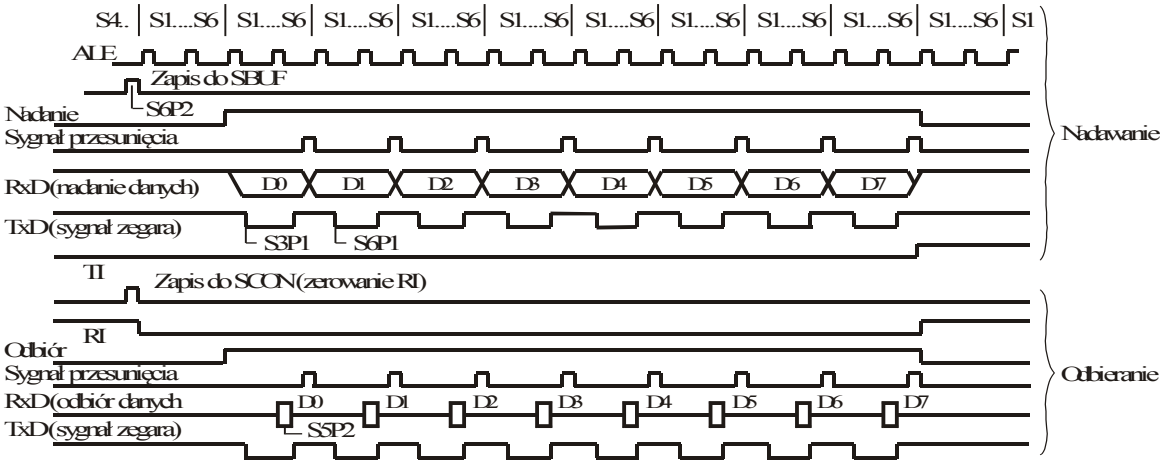
- transmisja synchroniczna,
- prędkość transmisji stała i równa 1/12 f_{osc},
- przesyłane 8 bitów danych.

Na rysunku 2.14 pokazano schemat układu portu szeregowego w trybie 0.



Rys. 2.14. Port szeregowy w trybie 0

Nadawanie danych ma miejsce w następnym cyklu maszynowym po wpisaniu danej przez mikroprocesor do rejestru SBUF. Dane nadawane są linią RxD począwszy od najmniej znaczącego bitu. Jednocześnie na linii TxD pojawiają się impulsy taktujące. Po nadaniu ośmiu bitów linie RxD i TxD powracają do stanu 1 i jest ustawiany bit TI. Odbiór informacji rozpoczyna się poprzez ustawienie bitu REN = 1 i RI = 0. W następnym cyklu maszynowym zostaje podany sygnał taktujący na linię TxD i następuje odczyt informacji bit po bicie (począwszy od najmniej znaczącego). Po odebraniu ośmiu bitów, cała informacja jest przepisywana do rejestru SBUF i ustawiany jest wskaźnik przerwania RI. Na rysunku 2.15 pokazano przebiegi czasowe sygnałów dla trybu 0.

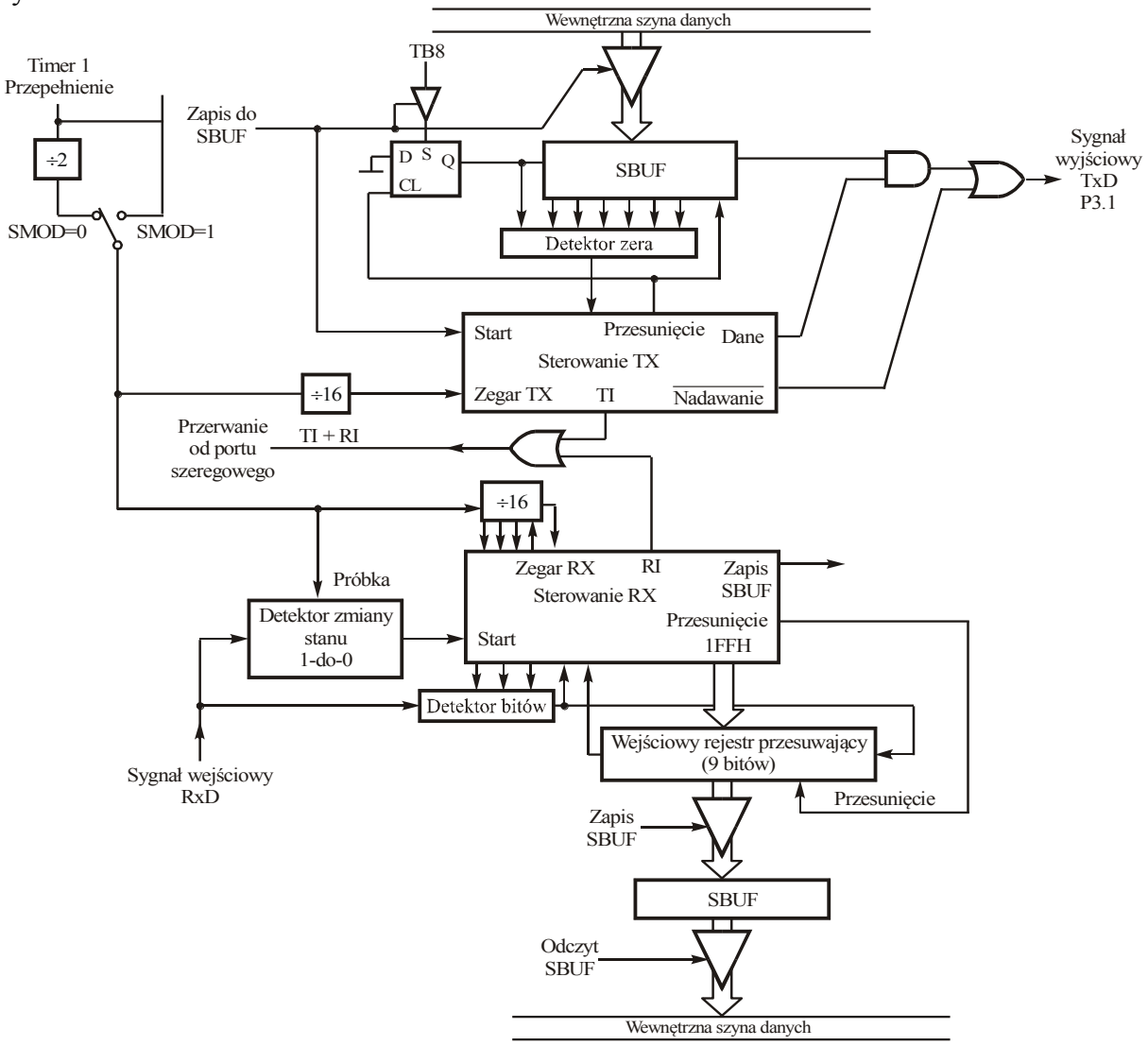


Rys. 2.15. Przebiegi czasowe sygnałów dla portu w trybie 0

1.10.2. Tryb 1 pracy układu transmisji szeregowej

Na rysunku 2.16 pokazano schemat układu portu szeregowego w t

rybie 1.



Rys. 2.16. Port szeregowy w trybie 1

Tryb 1 ma następujące cechy:

- transmisja asynchroniczna,
- płynnie regulowana prędkość transmisji,
- przesyłany bit startu równy 0,
- osiem bitów danych (od LSB),
- bit stopu równy 1.

Sygnałem taktującym transmisji są impulsy przepełnienia licznika T1 układu czasowego. Na prędkość transmisji ma również wpływ stan bitu SMOD rejestru PCON. Wyzerowanie bitu SMOD powoduje dwukrotne zmniejszenie prędkości transmisji. Prędkość transmisji możemy określić według wzoru:

$$V = (2^{\mathrm{SMOD}} * f_{\text{przepełnienia T1}})/32 \tag{2.3}$$

Licznik T1 może pracować w dowolnym trybie, jednak należy pamiętać o jego cyklicznym przeładowywaniu w razie konieczności. Często stosuje się tutaj tryb 2 licznika T1 (z automatycznym przeładowywaniem – odpada konieczność programowego przeładowywania licznika), wtedy wzór na prędkość transmisji ma następującą postać:

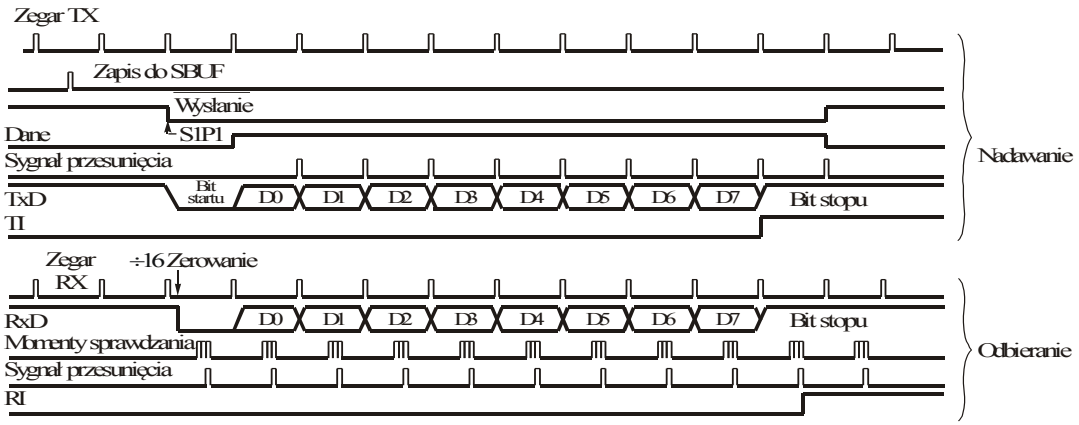
$$V = (f_{\mathrm{osc}} * 2^{\mathrm{SMOD}})/(32 * 12 * (256 - \mathrm{TH1})) \tag{2.4}$$

gdzie TH1 oznacza zawartość rejestru TH1.

Transmisję rozpoczyna zapis bajtu do rejestru SBUF. Wówczas w takt impulsów o częstotliwości $f_{\text{przepełnienia T1}}/16$ rozpoczyna się wysyłanie po linii TxD: bitu startu (0), 8 bitów danych, bitu stopu (1). Po wysłaniu bitu stopu zostaje ustawiony wskaźnik przerwania TI i linia TxD jest ustawiona w stan 1. Jeżeli odbiornik jest włączony (REN = 1), to bada on cały czas stan na wejściu RxD. Jeżeli wykryje opadające zbocze, to traktuje je jako początek bitu startu i następnie dokonuje kolejnego próbkowania w połowie czasu trwania bitu startu (wg ustalonej prędkości transmisji) i jeżeli na linii RxD nadal panuje stan niski, oznacza to, że został odebrany bit startu i linia RxD jest próbkowana w połowie czasu trwania każdego bitu danych. Sposób zakończenia odbioru zależy od bitów RI i SM2 oraz od wartości odebranego bitu stopu:

- Jeżeli RI = 1, to cały odebrany bajt jest ignorowany i odbiornik rozpoczyna na nowo nasłuchiwanie linii RxD; jeżeli RI = 0 i bit stopu ma wartość 1, to odbiór kończy się w sposób poprawny, czyli następuje przepisanie odebranego bajtu do rejestru SBUF, wartość bitu stopu jest zapisywana do bitu RB8 i następuje ustawienie wskaźnika przerwania RI;
- Jeżeli RI = 0 i bit stopu ma wartość 0, a SM2 = 1, to odebrany bajt jest ignorowany; jeżeli zaś SM2 = 0, to odbiór kończy się w sposób poprawny; sterowanie bitem SM2 umożliwia sprzętowe odrzucanie bajtów odebranych z błędem ramki.

Na rysunku 2.17 pokazano przebiegi czasowe sygnałów dla trybu 1.



Rys. 2.17. Przebiegi czasowe sygnałów dla portu w trybie 1

1.10.3. Tryb 2 pracy układu transmisji szeregowej

Tryb 2 ma następujące cechy:

- transmisja asynchroniczna;
- dwie, zależne od częstotliwości taktowania mikrokontrolera, prędkości transmisji;
- przesyłany bit startu równy 0;
- dziewięć bitów danych, bit stopu równy 1.

W trybie pracy są możliwe do uzyskania dwie prędkości transmisji – która z nich zostanie wybrana zależy od wartości bitu SMOD rejestru PCON.

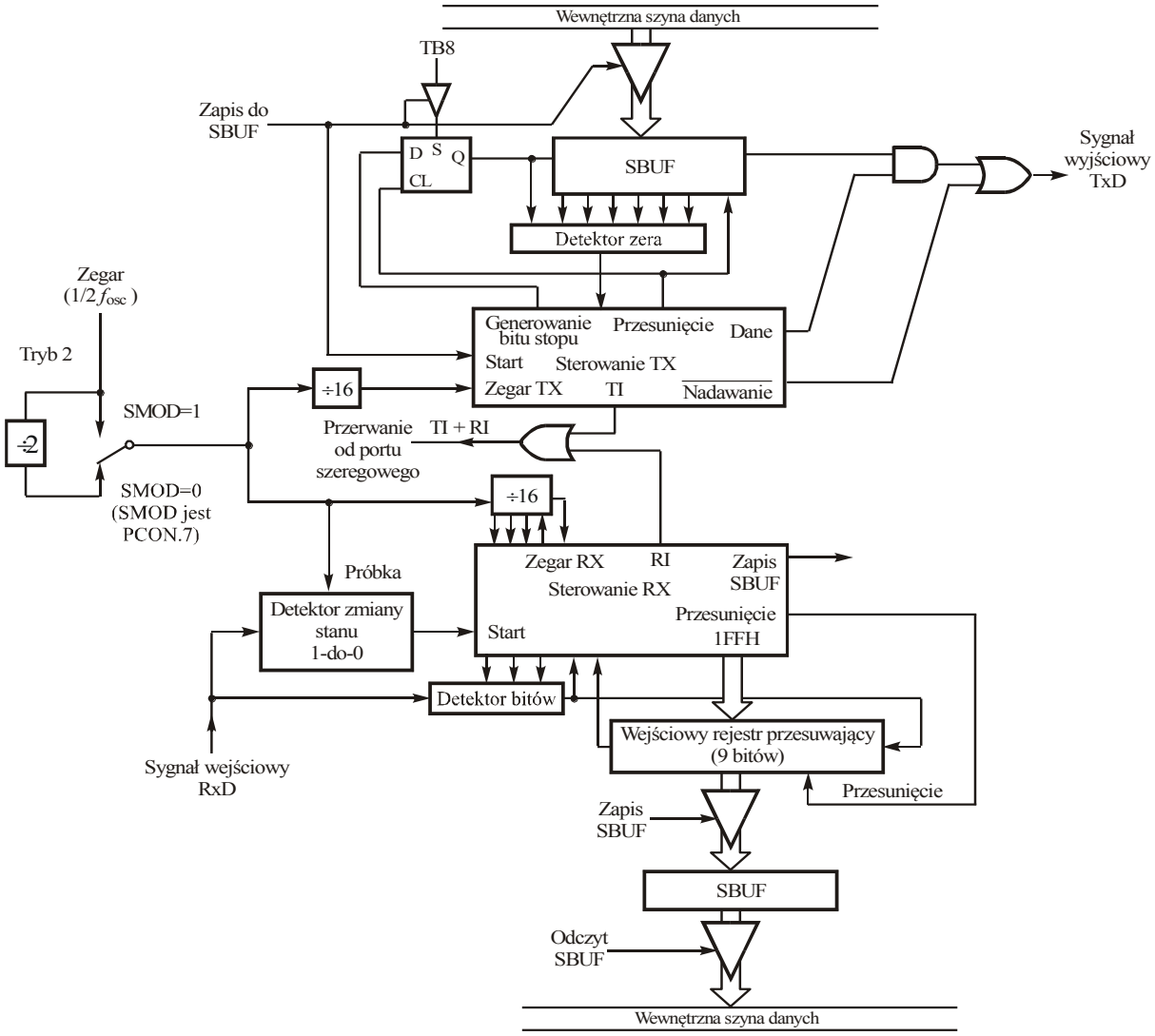
Prędkość można wyznaczyć z zależności:

$$V = f_{\text{osc}} * 2^{\text{SMOD}}/64 \tag{2.5}$$

Transmisja danych w trybie 2 przebiega tak samo jak w trybie 1, z tym, że jest transmitowany dziewiąty bit danych zapisany w bicie TB8, a odbierany dziewiąty bit jest zapisywany w bicie RB8. Warunki poprawnego odebrania informacji są identyczne z trybem 1, z tym że pod uwagę bierze się tu dziewiąty bit danych, a nie bit stopu (w trybie 2 bit stopu

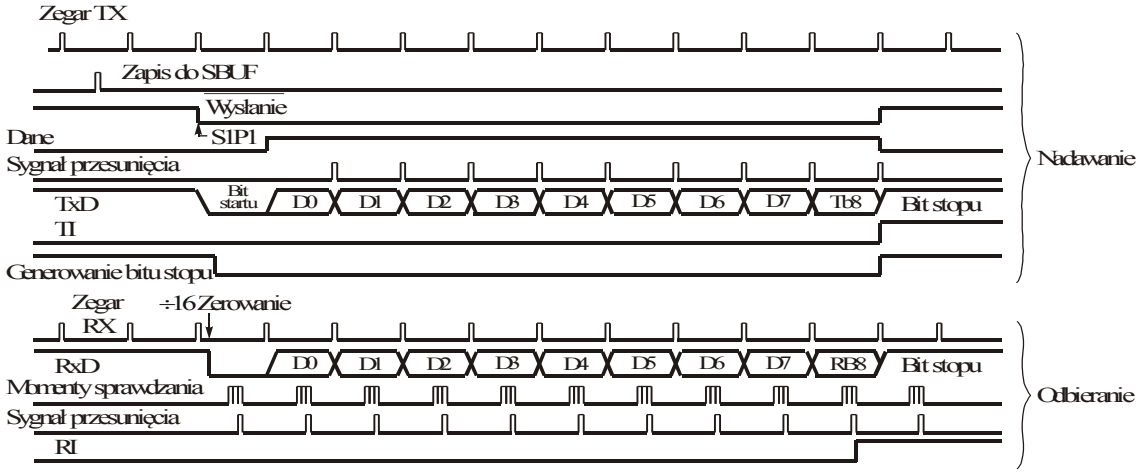
nie wpływa na proces odbioru). Transmitowany i odbierany dziewiąty bit może zostać wykorzystany do przesyłania bitu kontroli parzystości.

Na rysunku 2.18 pokazano schemat układu portu szeregowego w trybie 2.



Rys. 2.18. Port szeregowy w trybie 2

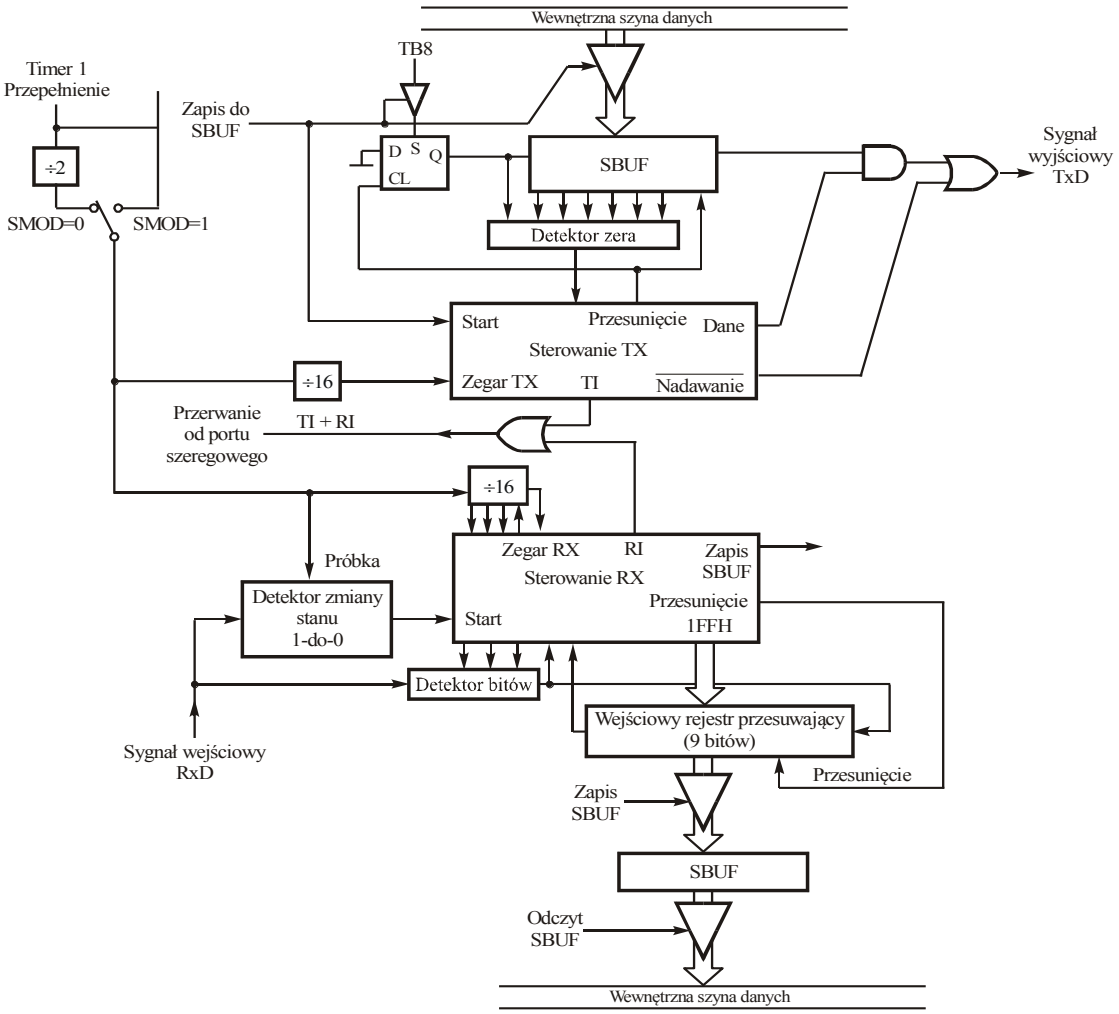
Na rysunku 2.19 pokazano przebiegi czasowe sygnałów dla trybu 2.



Rys. 2.19. Przebiegi czasowe sygnałów dla portu w trybie 2

1.10.4. Tryb 3 pracy układu transmisji szeregowej

Na rysunku 2.20 pokazano schemat układu portu szeregowego w trybie 3.

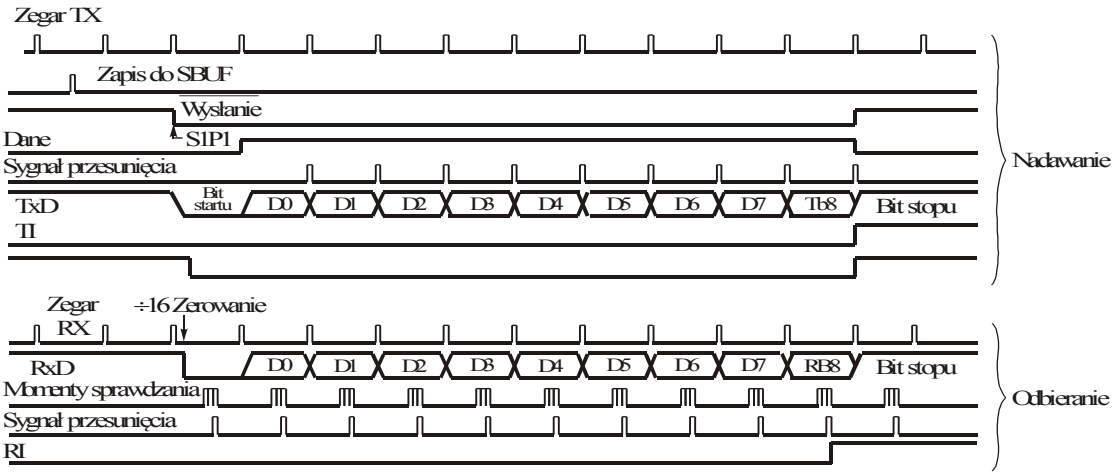


Rys. 2.20 Port szeregowy w trybie 3

Tryb 3 ma następujące cechy:

- transmisja asynchroniczna,
- płynnie regulowana prędkość transmisji,
- przesyłany jest bit startu równy 0,
- dziewięć bitów danych,
- bit stopu równy 1.

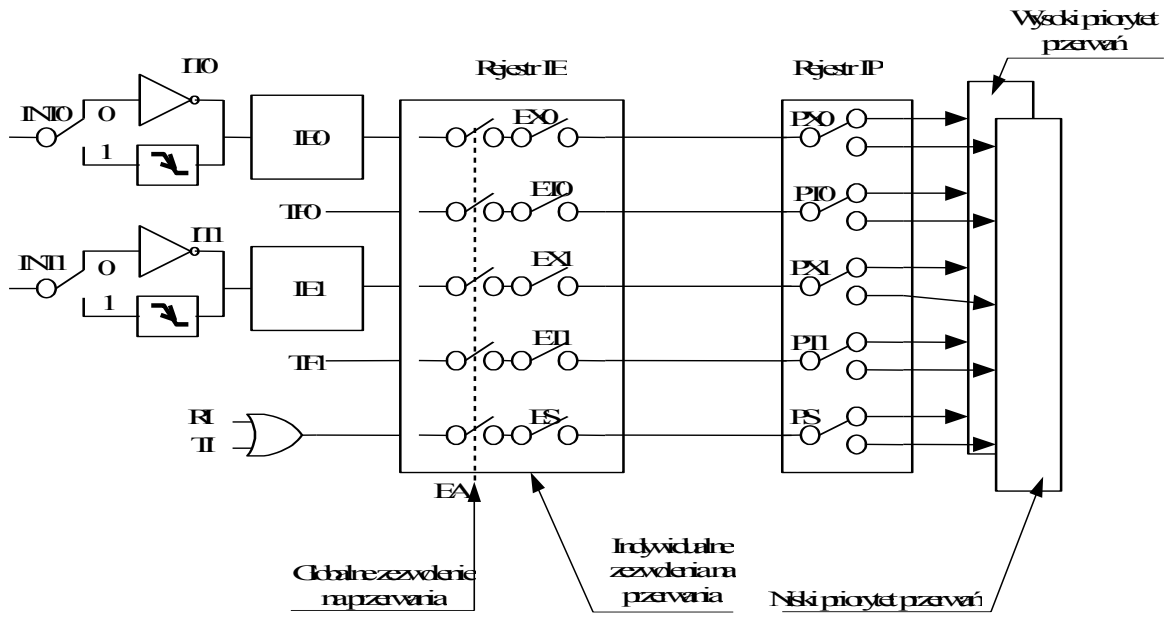
Tryb 3 stanowi połączenie właściwości trybów 1 i 2. W trybie tym układ transmisji szeregowej jest taktowany sygnałem przepełnienia licznika T1, tak samo określa się również prędkość transmisji. Logiczne cechy transmisji są identyczne z trybem 2. Na rysunku 2.21 pokazano przebiegi czasowe sygnałów dla trybu 3.



Rys. 2.21. Port szeregowy tryb pracy 3

1.11. Układ przerwań

Przerwaniem nazywamy reakcję mikroprocesora na zdarzenie zewnętrzne (np. sygnalizowane zmianą stanu na jednej z linii układu). Po wykryciu zdarzenia – żądania przerwania, mikroprocesor kończy wykonywanie bieżącej instrukcji, sprawdza czy obsłużenie danego przerwania jest dozwolone i jeżeli tak, wywołuje podprogram obsługi przerwania. Wywołanie to różni się od zwykłego wywołania podprogramu tym, że oprócz adresu powrotu zapamiętywany jest stan niektórych rejestrów mikroprocesora. Pociąga to za sobą także konieczność powrotu z takiego wywołania za pomocą specjalnej instrukcji o nazwie RETI. Każde przerwanie ma określony priorytet, dzięki czemu możliwy jest wybór kolejności obsługi przerwań, które nadeszły jednocześnie. Niektóre mikroprocesory umożliwiają programiście zmianę priorytetów wg własnego uznania. Adresy obsługi przerwań mogą być określone na stałe lub pobierane ze specjalnej tablicy wektorów przerwań. Pierwsze rozwiązanie zostało przyjęte w mikrokontrolerze 8051, a drugie jest znane z mikroprocesorów rodziny '86. Mikrokontroler 8051 jest wyposażony w priorytetowy, dwupoziomowy układ przerwań. Układ przerwań jest specjalną strukturą logiczną, której zadaniem jest monitorowanie stanu wskaźników przerwań i zgłaszanie faktu uaktywnienia określonego wskaźnika do układu sterowania. W mikrokontrolerze 8051 przerwanie może zostać wywołane przez jedno z pięciu źródeł zmiennych, jednym z pięciu wskaźników. Na rysunku 2.22 pokazano układ portu szeregowego mikrokontrolera 8051 wraz ze źródłami przerwań, a w tabl. 2.11 odpowiadające im bity wskaźników.



Rys. 2.22. Układ przerwań mikrokontrolera 8051

Cztery ze wskaźników są umieszczone w rejestrze TCON:

Tablica 2.11. Bity rejestru TCON

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0

Znaczenie poszczególnych bitów jest następujące:

TF1 – wskaźnik przerwania od licznika T1,

TF0 – wskaźnik przerwania od licznika T0,

TR1, TR0 – sterowanie licznikami T1 i T0 (nieistotne dla układu przerwań),

IE1 – wskaźnik przerwania zewnętrznego INT1' związany ze stanem logicznym lub zmianą stanu linii INT1',

IE0 – wskaźnik przerwania zewnętrznego INT0' związany ze stanem logicznym lub zmianą stanu linii INT0',

IT0, IT1 – sposób zgłaszania odpowiednich przerwań od linii zewnętrznych INT0' i INT1': 0 – zgłaszanie niskim poziomem napięcia, 1 – zgłaszanie zboczem opadającym.

Piątym źródłem przerwania jest układ transmisji szeregowej. Przerwanie to jest zgłaszane przez ustawienie dowolnego z bitów RI lub TI rejestru SCON.

W przypadku przerwań zewnętrznych i pochodzących od układów czasowych, wskaźniki przerwania są sprzętowo zerowane po przyjęciu zgłoszenia przerwania (za wyjątkiem sytuacji, gdy przerwanie zewnętrzne jest zgłaszane niskim poziomem). Wskaźniki przerwania z układu transmisji szeregowej muszą być zerowane programowo przez procedurę obsługi przerwania, gdyż sprzętowe zerowanie uniemożliwiłoby określenie, który ze wskaźników (RI czy TI) przerwanie wywołał.

Do uaktywniania poszczególnych przerwań i określania ich priorytetów są przeznaczone rejestry sterujące IE i IP, których budowę pokazano w tabl. 2.12.

Tablica 2.12. Rejestry sterujące układu przerwań

Rejestr	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
IE	EA	–	–	ES	ET1	EX1	ET0	EX0
IP	–	–	–	PS	PT1	PX1	PT0	PX0

Znaczenie bitów rejestru IE jest następujące:

EA – ustawienie bitu włącza układ przerwań, wyzerowanie wyłącza układ przerwań (blokuje wszystkie przerwania);

ES – ustawienie bitu powoduje włączenie obsługi przerwania z układu transmisji szeregowej;

ET1, ET0 – ustawienie bitów powoduje włączenie obsługi przerwań z odpowiednich liczników (T1 i T0);

EX1, EX0 – ustawienie bitów powoduje włączenie obsługi odpowiednich przerwań zewnętrznych.

Rejestr IP służy do określenia poziomu poszczególnych przerwań. 0 lub 1 na poszczególnych pozycjach przyporządkowują dane przerwanie do poziomu, odpowiednio:

0 — dla niskiego poziomu,

1 — dla wysokiego poziomu.

Znaczenie bitów rejestru IP jest następujące:

PS — ustalanie poziomu priorytetu przerwania z układu transmisji szeregowej,

PT1, PT0 — poziomy priorytetów przerwań z odpowiednich liczników,

PX1, PX0 — poziomy priorytetów odpowiednich przerwań zewnętrznych.

Podczas realizacji procedury obsługi przerwania poziomu 0 może nastąpić jej przerwanie przez procedurę obsługi przerwania o poziomie 1, nie może jednak wystąpić sytuacja odwrotna. Nie może również wystąpić wzajemne przerywanie procedur obsługi przerwań z tego samego poziomu. Dodatkowo podczas realizacji programu może wystąpić jednoczesne zgłoszenie dwóch lub więcej przerwań o tym samym poziomie. Powoduje to wybranie do wykonania przez układ przerwań obsługi przerwania o najwyższym priorytecie wg kolejności: $\overline{\text{INT0}}$ (priorytet najwyższy), TF0, $\overline{\text{INT1}}$, TF1, RI + TI (priorytet najniższy).

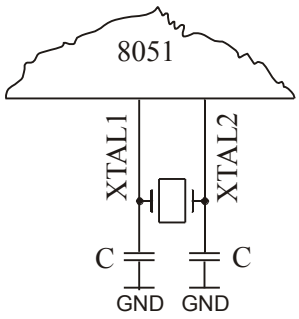
Przyjęcie przerwania powoduje sprzętową generacje rozkazu LCALL z adresem procedury obsługi przerwania, właściwym dla każdego przerwania (patrz podrozdział 2.7. „Pamięć programu"). Przyjęcie przerwania jest możliwe jednak tylko wtedy, gdy nie jest wykonywane przerwanie o równym lub wyższym priorytecie, nie jest wykonywany rozkaz (układ obsługi przerwania musi poczekać do zakończenia wykonywania tego rozkazu) lub jeżeli nie jest wykonywany rozkaz powrotu z procedury obsługi przerwania RETI, rozkaz dostępu do rejestrów IE i IP lub jakikolwiek rozkaz bezpośrednio wykonywany po nich.

1.12. Generator sygnału taktującego

Mikroprocesory mają wejście zegarowe, przez które jest podawany sygnał taktujący pracę układu lub mają wewnętrzny generator sygnału taktującego. Sygnał ten jest zazwyczaj dzielony za pomocą dzielników częstotliwości na szereg sygnałów o mniejszej częstotliwości, przesuniętych względem siebie w fazie. Okres takiego sygnału o mniejszej częstotliwości wyznacza cykl maszynowy, podczas którego mikroprocesor jest w stanie wykonać operację elementarną, np. pobrać rozkaz z pamięci i zdekodować go lub wykonać wcześniej zdekodowany rozkaz. W zależności od stopnia skomplikowania rozkazu i liczby argumentów, wykonanie rozkazu może trwać 1 lub więcej cykli maszynowych. Grupę cykli maszynowych, podczas których jest wykonywany jeden rozkaz, nazywa się cyklem rozkazowym.

Mikrokontroler 8051 ma wbudowany generator sygnału zegarowego, mogący współpracować z rezonatorem kwarcowym lub ceramicznym. Generator wytwarza sygnał taktujący mikrokontroler o częstotliwości równej częstotliwości zastosowanego rezonatora (układ powoduje wzbudzenie rezonatora na częstotliwości podstawowej). Przygotowanie generatora (a zarazem całego mikrokontrolera) do pracy polega na przyłączeniu do

wyprowadzeń XTAL1 i XTAL2 rezonatora (kwarcowego lub ceramicznego) oraz dwóch kondensatorów wg schematu pokazanego na rys. 2.23.



Rys. 2.23. Schemat przyłączenia rezonatora do mikrokontrolera

Pojemność kondensatorów nie jest zbyt istotna: dla rezonatora kwarcowego jest to zwykle wartość w granicach 20 ÷ 40 pF, dla rezonatora ceramicznego 45 ÷ 60 pF. Możliwe jest również taktowanie mikrokontrolera sygnałem z zewnętrznego generatora. Częstotliwość sygnału powinna zawierać się w przedziale zalecanym przez producenta (dla układu 8051 w granicach 1,2 ÷ 12 MHz), a wypełnienie powinno być równe 50% (choć niektórzy producenci dopuszczają taktowanie sygnałem o innym wypełnieniu). Sposób przyłączenia generatora do mikrokontrolera jest różny, w zależności od technologii wykonania układu. Dla mikrokontrolerów wykonanych w technologii NMOS sygnał z generatora należy doprowadzić do wyprowadzenia XTAL2, a wyprowadzenie XTAL1 połączyć z masą układu. W przypadku mikrokontrolerów wykonanych w technologii CMOS (z literą C w nazwie, np. 80C51) sygnał generatora doprowadza się do wyprowadzenia XTAL1, natomiast wyprowadzenie XTAL2 pozostawia się nie przyłączone.

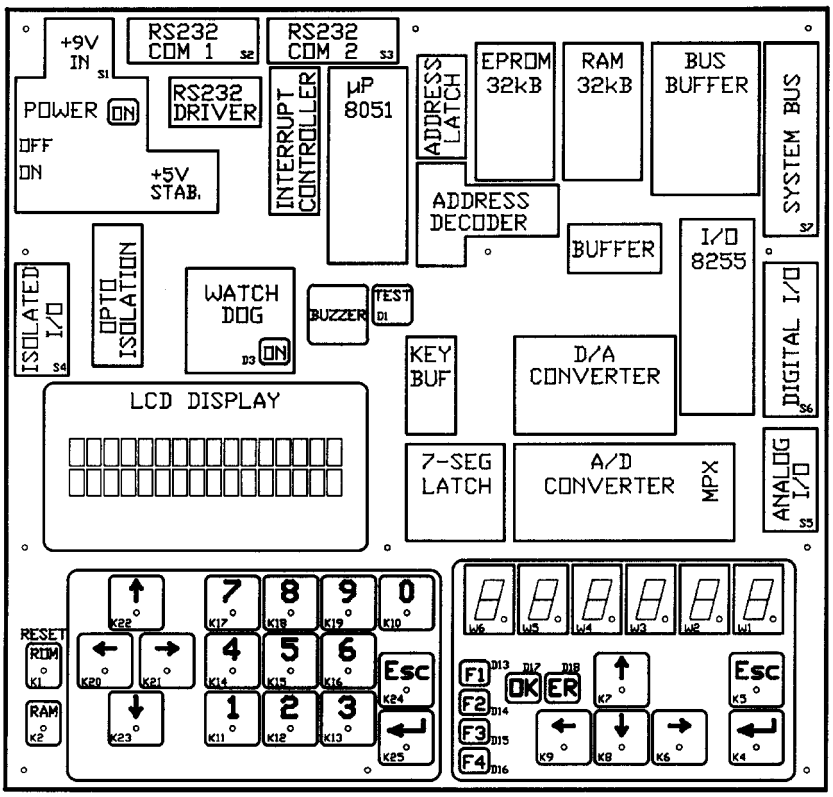
Producenci mikrokontrolerów stawiają również wymagania co do parametrów elektrycznych sygnału taktującego. Najczęściej od sygnału generatora oczekuje się następujących parametrów: w stanie niskim napięcie nie powinno przekraczać 0,75 V, natomiast w stanie wysokim nie powinno być niższe niż 2,5 V w przypadku układów NMOS lub 3,5 V w przypadku mikrokontrolerów CMOS.

Powyższe cechy generatora taktującego mikrokontrolera 8051 są w większości przypadków identyczne z innymi mikrokontrolerami rodziny '51. Zazwyczaj najbardziej znaczącą różnicą jest maksymalna częstotliwość rezonatora (zewnętrznego generatora) taktującego mikrokontroler.

2. Opis systemu DSM-51

2.1. Wprowadzenie

Z programowaniem mikroprocesorów przy wykorzystaniu języków asembler czy też C można się spotkać podczas budowy różnego rodzaju sterowników, stosowanych zarówno w przemyśle, jak i sprzęcie powszechnego użytku. Ale nauka programowania mikrokontrolerów bez możliwości faktycznego uruchomienia napisanych programów i obserwacji ich działania jest niezwykle trudna. Obserwując działanie nawet najprostszego programu na programowym symulatorze mikroprocesora, nie jesteśmy w stanie w pełni zaobserwować i zrozumieć działania tego programu, uruchamiając go w prawdziwym systemie. Aby móc w pełni ocenić działanie oraz pojąć możliwości drzemiące w mikrokontrolerze i programie, konieczny jest kontakt z realnie działającym systemem. Aby ułatwić zapoznanie się z funkcjonowaniem mikrokontrolera i jego programowaniem, firma MICROMADE opracowała na bazie mikrokontrolera 8051 dydaktyczny system mikroprocesorowy DSM-51. Jako materiał dydaktyczny umożliwiający zapoznanie się z tym systemem wydawnictwo MIKOM wydało podręcznik „Podstawy programowania mikrokontrolera 8051. Pracownia systemów mikroprocesorowych na bazie DSM-51" autorstwa Piotra i Pawła Gałków. Podręcznik ten zawiera opis systemu DSM-51 oraz 20 lekcji dotyczących ćwiczeń z jego użyciem, które wprowadzają Czytelnika w podstawy programowania w języku asembler. Rozwój techniki mikroprocesorowej oraz coraz powszechniejsze wykorzystywanie mikrokontrolerów w sprzęcie powszechnego użytku, wymusiły przyśpieszenie procesu zarówno powstawania urządzenia, jak i tworzenia oprogramowania do jego obsługi.



Rys. 1.1. Płyta czołowa systemu mikroprocesorowego DSM-51

Jedną z dróg przyśpieszenia procesu tworzenia programu była zmiana poziomu języka na wyższy, prostszy i łatwiejszy do przyswojenia. Można stwierdzić, że programowanie mikrokontrolerów w języku C częściowo wyparło programowanie w języku asembler i obecnie jest najpopularniejszym sposobem na tworzenie oprogramowania dla mikrokontrolerów. Dlatego konieczne stało się bliższe zapoznanie się z programowaniem w języku C i umożliwienie prostego i szybkiego implementowania tego oprogramowania w rzeczywistym urządzeniu. Po przeanalizowaniu budowy i oprogramowania systemu DSM-51 okazało się, że może on być z powodzeniem stosowany również jako narzędzie do nauki programowania w języku C.

Na rysunku 1.1 przedstawiono widok płyty czołowej DSM-51, na którą naniesiono oznaczenia poszczególnych bloków systemu, takich jak:

POWER — blok zasilania urządzenia wyposażony w stabilizator napięcia 5V, włącznik napięcia, diodę sygnalizacyjną oraz złącze zasilające;

μP 8051 — mikrokontroler 8051;

EPROM 32 kB, RAM 32 kB — bloki pamięci stałej oraz ulotnej;

BUS BUFFER — blok buforowanej szyny systemowej składający się z sterownika BUS BUFFER oraz złącza SYSTEM BUS (S7);

INTERRUPOT CONTROLER — sterownik przerwań;

ADDRESS LATCH — bufor adresów;

ADDRES DECODER — dekoder adresów;

I/O 8255 — blok dwukierunkowych we/wy cyfrowych składający się z układu we/wy cyfrowych (I/O 8255) oraz złącza (DIGITAL I/O- S6);

RS-232 — blok łącza RS-232 składający się ze sterownika (RS−232 DRIVER) oraz dwu złączy (RS-232 COM S2 i S3);

LCD DISPLAY — blok wyświetlacza LCD (2 linie po 16 znaków);

7-SEG LATCH — blok wyświetlacza 7-segmentowego składający się z 6 cyfr (W1−W6), 6 diod (D13−D18) oraz buforu wskaźnika 7-SEG LATCH;

Klawiatura przeglądana sekwencyjnie składająca się z 6 klawiszy (K4−K9);

Klawiatura matrycowa składająca się z 16 klawiszy (K10−K25) oraz buforu klawiatury KEY BUF;

RESET ROM i RAM — klawisze reset (K1 i K2);

TEST — sygnalizator optyczny w postaci diody D1;

BUZZER — sygnalizator akustyczny w postaci brzęczyka;

A/D CONVERTER — blok konwertera analogowo-cyfrowego z multiplexerem MPX i złączem ANALOG I/O (S5);

D/A CONVERTER — blok konwertera cyfrowo-analogowego ze złączem ANALOG I/O (S5) i buforem danych BUFFER;

WATCHDOG — blok kontroli pracy systemu;

OPTO ISOLATION — blok izolacji galwanicznej ze złączem ISOLATED I/O (S4).

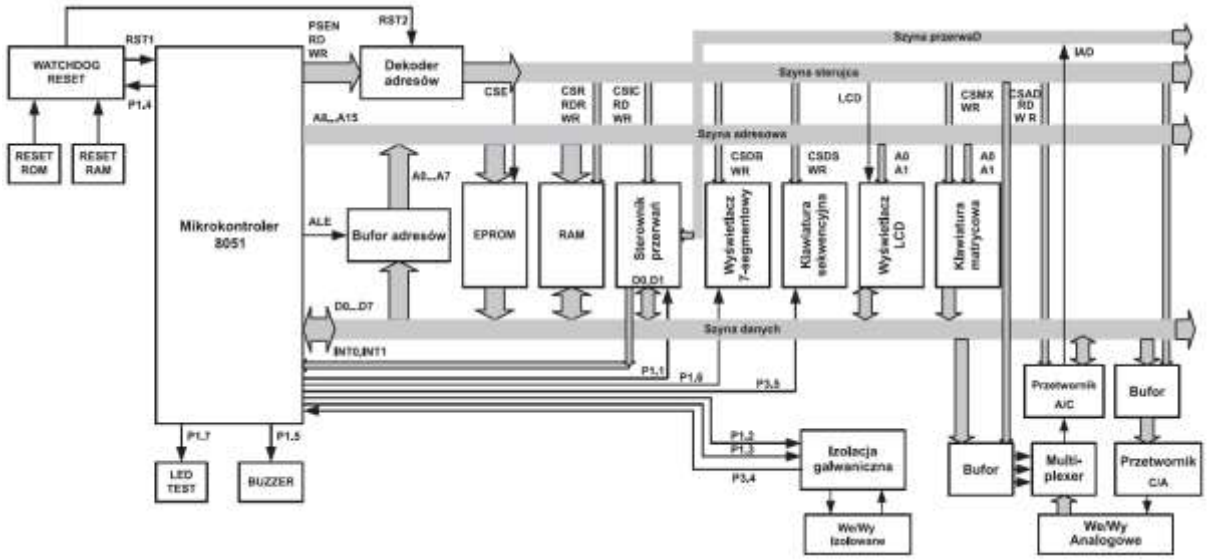
Oznaczenia są naniesione na płytce przezroczystej, będącej zabezpieczeniem przed przypadkowymi uszkodzeniami, a która jednocześnie umożliwia obserwację elementów urządzenia. Wszystkie układy scalone umieszczono na podstawkach, co ułatwia ich wymianę w przypadku uszkodzenia. System zawiera oprogramowanie zapisane w pamięci EPROM, umożliwiające uruchamianie programów przesłanych do pamięci RAM urządzenia łączem szeregowym. System DSM-51 może tak otrzymany program wykonywać krokowo pod nadzorem komputera PC lub samodzielnie z pełną wydajnością.

2.2. Wyposażenie systemu mikroprocesorowego DSM-51

System DSM-51 jest uniwersalnym sterownikiem mikroprocesorowym zawierającym wiele elementów zewnętrznych, które są przykładami zarówno układów komunikacji z użytkownikiem, jak i układów pomiarowych oraz sterowania. Sterownik umożliwia testowanie programów o różnym poziomie trudności: od krótkich programów obsługujących diodę świecącą, poprzez obsługę wyświetlaczy i klawiatur, aż do złożonych programów sterujących modelami rzeczywistych urządzeń. Ponieważ w praktyce spotyka się różne układy spełniające podobne funkcje, więc system DSM-51 został wyposażony w alternatywne układy, umożliwiające naukę programowania elementów, takich jak:

- wyświetlacz 7-segmentowy wymagający sekwencyjnego sterowania oraz wyświetlacz ciekłokrystaliczny sterowany za pomocą wbudowanego sterownika;
- klawiatura przeglądana sekwencyjnie oraz klawiatura matrycowa odczytywana równolegle;
- we/wy cyfrowe sterowane bezpośrednio za pomocą portów mikrokontrolera oraz linie we/wy obsługiwane przez układ 8255;
- łącze szeregowe RS-232 sterowane sprzętowo za pomocą mikrokontrolera 8051 oraz łącze szeregowe RS-232 obsługiwane programowo.

Schemat blokowy systemu mikroprocesorowego DSM-51 pokazano na rys. 1.2.



Rys. 1.2. Schemat blokowy systemu mikroprocesorowego DSM-51

W schemacie sterownika możemy wyróżnić następujące elementy:

- mikrokontroler 80C31 (zegar 11,059 MHz),
- watchdog,
- RAM 32 kB,
- dekoder adresów — GAL16V8,
- sterownik przerwań — GAL16V8,
- sygnalizatory: LED, buzzer,
- wyświetlacz LED (6 elementów 7-segmentowych),
- wyświetlacz LCD (2 wiersze po 16 znaków),
- klawiatura przeglądana sekwencyjnie (6 klawiszy),
- klawiatura matrycowa (2 grupy po 8 klawiszy),
- 2 kanały RS-232,
- 24 linie we/wy cyfrowych układ 8255,
- 2 linie wejść cyfrowych izolowanych galwanicznie,

- 2 linie wyjść cyfrowych izolowanych galwanicznie,
- przetwornik analogowo-cyfrowy,
- 1 linia wyjścia analogowego,
- 8 linii wejść analogowych,
- przetwornik cyfrowo-analogowy,
- buforowana szyna systemowa.

System został wyposażony w szereg złączy:

- we/wy cyfrowych,
- we/wy analogowych,
- we/wy cyfrowych izolowanych galwanicznie,
- dwu kanałów RS-232,
- szyny systemowej.

Złącza te umożliwiają sterowanie różnymi układami lub urządzeniami zewnętrznymi, komunikowanie się z innymi systemami mikroprocesorowymi i komputerami, rozbudowę systemu (złącze szyny systemowej).

2.3. Szyna systemowa, pamięć RAM, pamięć EPROM, dekoder adresów

Schemat elektryczny sterownika DSM-51 pokazano na rys. 1.3. Jako główny element sterujący w systemie mikroprocesorowym DSM-51 został zastosowany mikrokontroler rodziny ’51 w wersji pozbawionej pamięci ROM (80C31). Budowa oraz działanie mikrokontrolera są omówione w rozdz. 2 niniejszej książki. Mikrokontrolery z rodziny ‘51 komunikują się z elementami zewnętrznymi za pomocą szyny systemowej, w której można wyodrębnić następujące części składowe (rys. 1.2):

- szyna adresowa — 16 linii oznaczonych od A0 do A15; na tę szynę jest podawany adres; pod każdym adresem może znajdować się komórka pamięci lub jeden 8-bitowy port we/wy; ustawienie adresu polega na ustawieniu 0 lub 1 na każdej linii adresowej; maksymalnie można w ten sposób zaadresować 65 536 komórek (64 kB);
- szyna danych — 8 linii oznaczonych od D0 do D7; po szynie danych jest przesyłana zawartość komórki pamięci lub portu; szyna danych jest dwukierunkowa i umożliwia przesyłanie danych z i do mikrokontrolera; przesyłanie danych to ustawianie poszczególnych linii danych w stan 0 lub 1, przez urządzenie zewnętrzne (odczyt) lub przez mikrokontroler (zapis);
- szyna sterująca — składa się z trzech następujących linii sterujących przesyłaniem danych (w mikrokontrolerze 8051):

  PSEN — odczyt pamięci programu,
  RD — odczyt pamięci danych,
  WR — zapis pamięci danych.

Do szyny systemowej (rys. 1.2) przyłącza się urządzenia we/wy, pamięć programu i pamięć danych. W pamięci programu jest umieszczony program główny systemu. Jest to pamięć EPROM 27C256 (element U3 na rys. 1.3). W czasie pracy mikrokontroler 8051 pobiera z pamięci programu kolejne rozkazy i je realizuje. Ponieważ pamięć ta jest tylko odczytywana, więc do jej obsługi przewidziano tylko jedną linię PSEN. Pamięć danych służy do przechowywania informacji niezbędnych w trakcie wykonywania programu. Mikrokontroler 8051 ma wewnętrzną pamięć RAM. W przypadkach gdy jest konieczne zwiększenie pamięci RAM, można przyłączyć do szyny systemowej zewnętrzną pamięć RAM 62 256 (element U4 na rys.1.3). Do obsługi pamięci RAM, która jest zapisywana i odczytywana, przewidziano dwie linie sterujące: RD (odczyt pamięci danych) i WR (zapis pamięci danych). Ponieważ do szyny systemowej przyłączonych może być kilka układów pamięci i wiele portów we/wy, więc układy te są wyposażone w wejście wyboru układu.

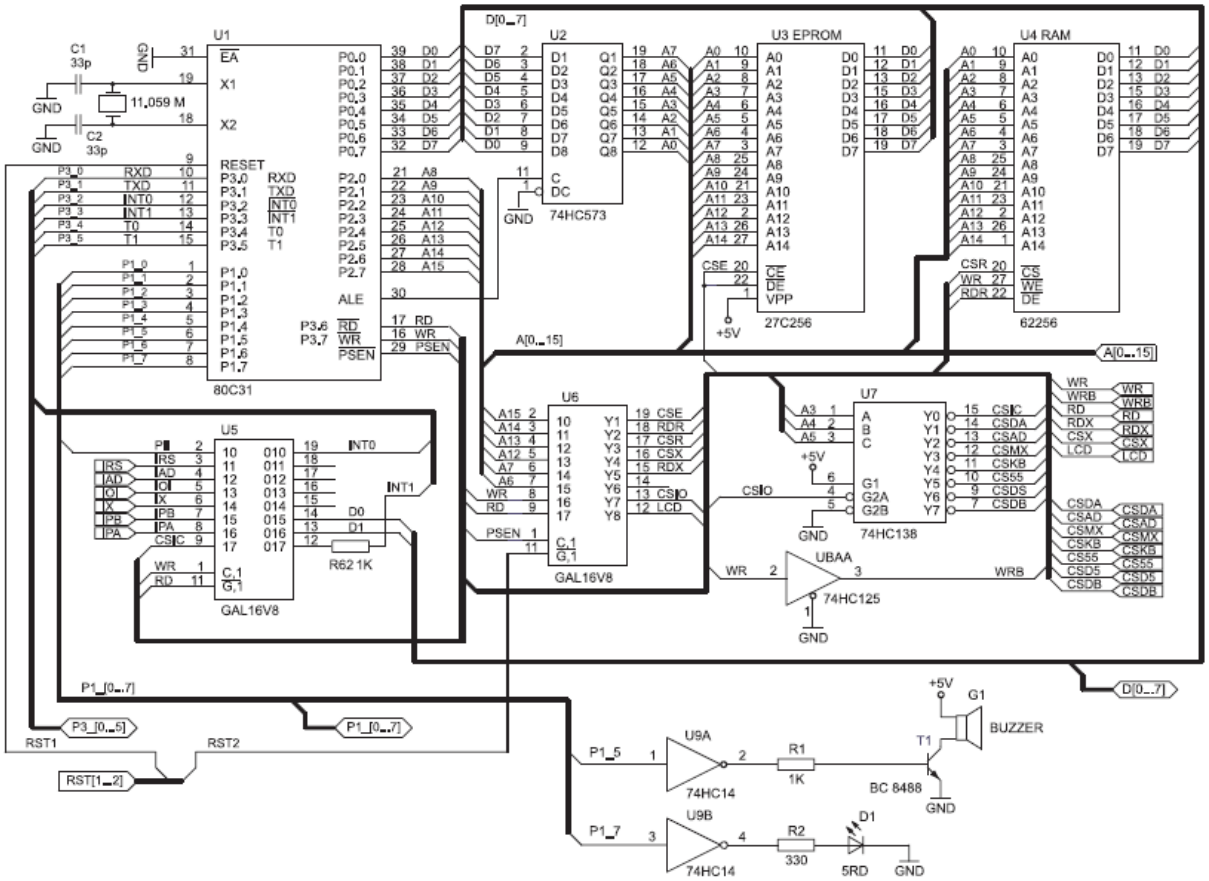
Wejście to jest oznaczone jako CS (ang. *Chip Select*). Zapis do pamięci następuje przy jednoczesnym wystąpieniu sygnałów CS i WR, a odczyt przy sygnałach CS i RD.

Do rozdzielenia sygnałów CS służą układy dekoderów adresów; mają one za zadanie wygenerować sygnał CS tylko do jednego konkretnego układu, uzależnionego od wysłanego przez mikrokontroler adresu. W systemie mikroprocesorowym DSM-51 konieczne było umożliwienie zapisu do pamięci programu przesłanego z komputera kodu (programu) i uruchomienie go. Posłużył do tego dekoder adresów.

Dekoder adresów został zbudowany na podstawie układu GAL 16V8 (element U6 na rys. 1.3), który jest matrycą logiczną, składającą się z wielu bramek. Wzajemne połączenia wewnętrzne między bramkami są dokonywane na drodze programowania układu GAL. Dekoder adresów w systemie DSM 51 może pracować w jednym z dwu trybów:

TRYB0 — w zewnętrznym obszarze pamięci programu znajduje się układ pamięci EPROM; kod programu jest czytany z pamięci EPROM; w obszarze zewnętrznej pamięci danych znajduje się układ pamięci RAM (32 kB); zapis danych do obszaru zewnętrznej pamięci danych powoduje ich wpisanie do pamięci RAM;

TRYB1 — w tym trybie, w obszarze zewnętrznej pamięci programu, znajduje się pamięć RAM; kod programu jest czytany w tym trybie z pamięci RAM; jedynie obszar zewnętrznej pamięci programu o adresie powyżej 0x8000 jest czytany z pamięci EPROM.



Rys. 1.3. Schemat systemu mikroprocesorowego DSM-51

Dzięki takiej pracy dekodera adresów, w trybie 0 jest możliwy zapis kodu programu z komputera PC przez łącze szeregowe RS-232 do mikrokontrolera i dalej do pamięci RAM (w tym przypadku program obsługi łącza szeregowego, zapisu do pamięci RAM itp. znajduje się w obszarze 0x0000...0x8000 układu EPROM). Pozostałe układy systemu mikroprocesorowego DSM-51 są, bez względu na tryb pracy dekodera adresów, umieszczone pod tymi samymi adresami w obszarze zewnętrznej pamięci danych. Zostało to zrealizowane

przez doprowadzenie do dekodera adresów sygnałów RD i PSEN. Na ich podstawie oraz na podstawie linii adresowych wygenerowane zostały sygnały sterujących układami pamięci:

CSE — wybór i odczyt pamięci EPROM,

CSR — wybór pamięci RAM,

RDR — odczyt pamięci RAM.

Sygnał PSEN (odczyt pamięci programu) trafia do pamięci EPROM lub RAM, zależnie od adresu i trybu pracy dekodera.

W tablicy 1.1 pokazano wszystkie wcześniej wymienione układy wraz z ich adresami, opisem oraz możliwymi do wykonania operacjami (WR zapis /RD odczyt).

Tablica 1.1. Urządzenia we/wy systemu mikroprocesorowego DSM-51 oraz ich adresy w pamięci

Dolna część adresu (Bin)	Pełny adres (Hex)	Symbol	Możliwe operacje	Urządzenie
0000 0xxx B	FF 00 H	CSIC	WR	sterownik przerwań
0000 1xxx B	FF 08 H	CSDA	WR	przetwornik C/A
0001 0xxx B	FF 10 H	CSAD	RD/WR	przetwornik A/C
0001 1xxx B	FF 18 H	CSMX	RD	multiplekser analogowy
0010 0x01 B	FF 21 H	CSKB0	WR	klawiatura matrycowa: klawisze od 0 do 7
0010 0x10 B	FF 22 H	CSKB1	WR	klawiatura matrycowa: klawisze od 8 do ...
0010 1x00 B	FF 28 H	CS55A	RD/WR	układ 8255 – rejestr portu A
0010 1x01 B	FF 29 H	CS55B	RD/WR	układ 8255 – rejestr portu B
0010 1x10 B	FF 2A H	CS55C	RD/WR	układ 8255 – rejestr portu C
0010 1x11 B	FF 2B H	CS55D	RD/WR	układ 8255 – rejestr sterujący
0011 0xxx B	FF 30 H	CSDS	WR	bufor wyboru wskaźnika 7-segmentowego
0011 1xxx B	FF 38 H	CSDB	WR	bufor danych wskaźnika 7-segmentowego
01xx xxxx B	FF 40 H	CSMOD	RD/WR	tryb dekodera adresów
10xx xx00 B	FF 80 H	LCDWC	WR	HD44780 – wpis rozkazów do LCD
10xx xx01 B	FF 81 H	LCDWD	WR	HD44780 – wpis danych do LCD
10xx xx10 B	FF 82 H	LCDRC	RD	HD44780 – odczyt stanu do LCD
10xx xx11 B	FF 83 H	LCDRD	RD	HD44780 – odczyt danych do LCD
11xx xxxx B	FF C0 H	CSX	RD\WR	zewnętrzna magistrala systemowa do LCD

2.4. Sterownik przerwań

W systemie DSM-51 rozbudowano system przerwań o dodatkowe źródła zewnętrzne. Sterownik przerwań został zbudowany na podstawie układu GAL16V8 (element U5). Umiejscowienie oraz przyłączenie układu GAL 16V8 jest pokazane na rys. 1.3.

Do sterownika doprowadzono sześć sygnałów przerwań:

IAD — z przetwornika analogowo-cyfrowego,

IOI — z wejścia izolowanego galwanicznie,

IPA — z układu 8255,

IPB – z układu 8255,

IX — ze złącza szyny systemowej,

IRS — kanału COM2.

Sterownik sygnalizuje przyjście przerwania przez obydwa zewnętrzne wejścia przerwań mikrokontrolera 8051 – INT0 i INT1. Oprócz tego sterownik jest przyłączony do szyny systemowej DSM-51 jako urządzenie wybierane sygnałem CSIC. Doprowadzone są do niego sygnały sterujące oraz dwie linie danych (D0 i D1). Dzięki temu jest możliwy odczyt numeru zgłoszonego przerwania oraz zapis kasujący zgłoszone przerwanie. Przy wykorzystaniu dwóch linii możliwe jest przesłanie 4 różnych numerów przerwań, dokładnie tylu, ile potrafi obsłużyć sterownik. Ponieważ do sterownika doprowadzonych jest więcej sygnałów przerwań, więc w ich wyróżnieniu pomaga linia P1_1 mikrokontrolera przyłączona do

sterownika. Linia P1_1 wyróżnia dwa tryby pracy sterownika przerwań zależne od stanu linii P1_1. Tryby pracy zostały przedstawione w tabl. 1.2.

Tablica 1.2. Tryby pracy sterownika przerwań

Nr przerwania	Sygnał przerwania		Priorytet przerwania
	P1.1 = 1	P1.1 = 0	
0	IAD	IAD	najwyższy
1	IOI	IOI	
2	---	IPA	
3	IX	IPB	najniższy

2.5. Sygnalizatory Led i Buzzer

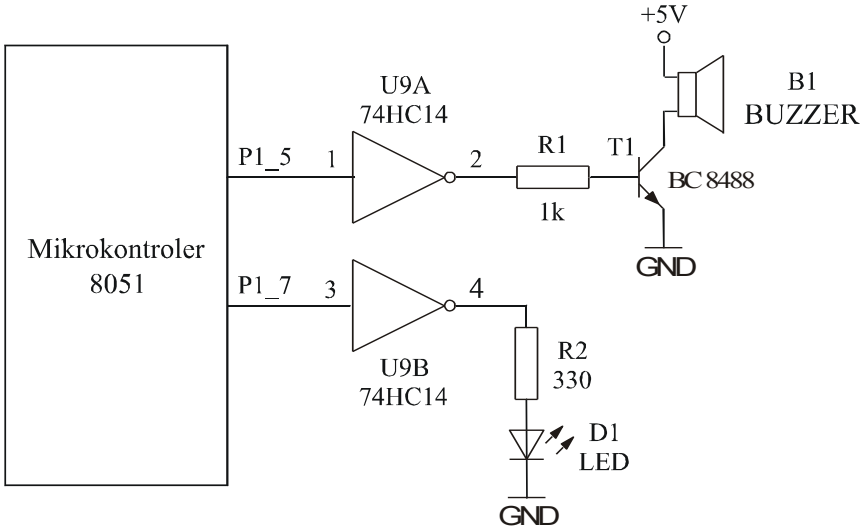
Najprostszymi elementami, którymi może sterować mikrokontroler, są sygnalizatory dwustanowe (stan niski: spoczynek, stan wysoki: praca). Bezpośrednio pokazują one stan linii wyjściowej mikrokontrolera. System mikroprocesorowy DSM-51 został wyposażony w dwa sygnalizatory:

optyczny — dioda led przyłączona do linii 7 portu P1 (sterowana stanem bitu 7 rejestru P1),

dźwiękowy — brzęczyk (buzzer) przyłączony do linii 5 portu P1 (sterowany stanem bitu 5 rejestru P1).

Sposób przyłączenia obu elementów został pokazany na rys. 1.4. Włączenie sygnału dźwiękowego następuje z chwilą podania na dolne wyprowadzenie brzęczyka B1 sygnału masy przez tranzystor T1, co jest wynikiem podania dodatniego napięcia między bazę a emiter tranzystora T1. Napięcie to powstaje ze względu na podanie na rezystor R1 napięcia 5 V z wyjścia bramki U9A będącej negatorem, sterowanej z wyprowadzenia o nazwie P1_5 procesora 8051. Z powyższego opisu wynika, że aby brzęczyk emitował dźwięk, linia P1_5 musi zostać ustawiona w stan logicznego 0.

Kolejny element sygnalizacyjny, dioda LED, jest sterowany w podobny sposób, przy czym w tym przypadku po podaniu logicznego zera z wyprowadzenia P1_7 na wejście bramki U9B, bezpośrednio z wyjścia tego układu , jest podawane napięcie 5 V na szeregowy układ rezystora R2 i diody LED D1, powodując świecenie diody wymuszone przepływem prądu o wartości ograniczonej opornością rezystora.



Rys. 1.4. Schemat przyłączenia sygnalizatorów LED i BUZZER do mikrokontrolera 8051

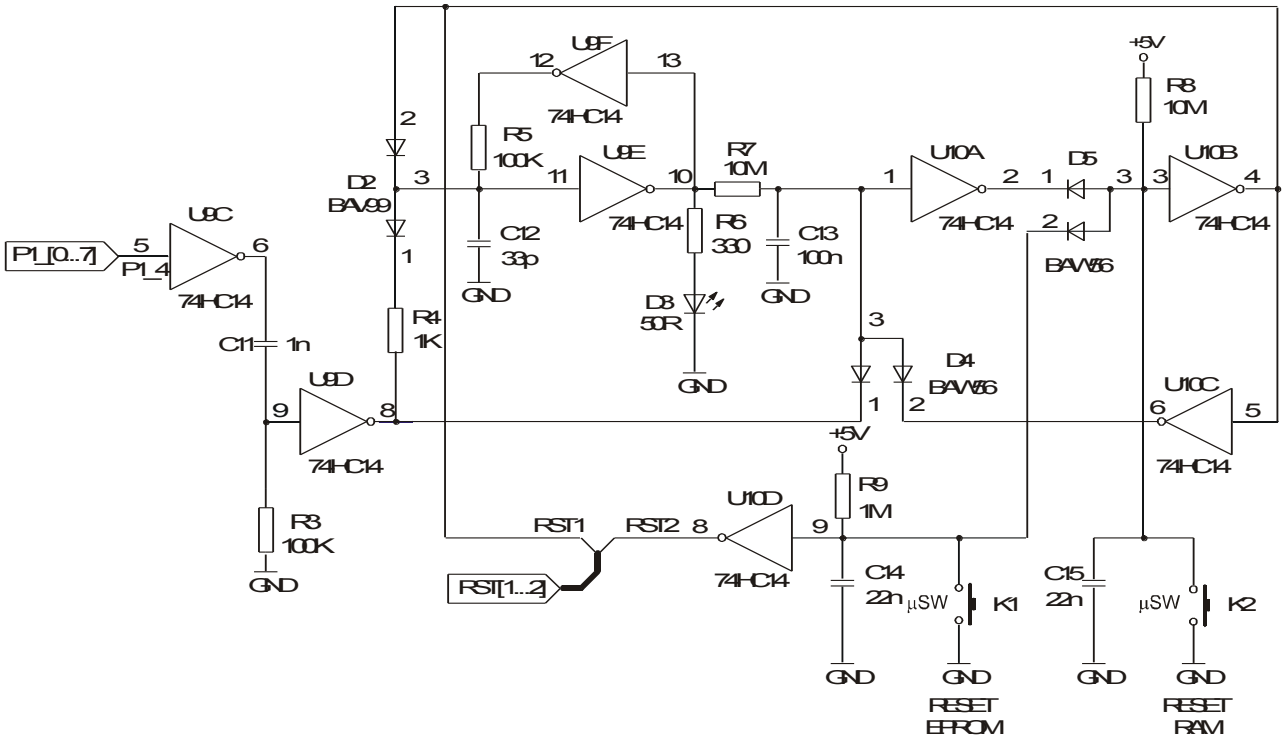
2.6. Układ *Watchdog* — budowa i działanie

Właściwa praca sytemu mikroprocesorowego opiera się na poprawnym wykonywaniu zadanego programu. Każdy program składa się z wielu operacji, a każda operacja to wiele sygnałów elektrycznych przekazywanych przez wiele linii. Najmniejszy błąd, polegający np. na złej interpretacji jednego z bitów kodu rozkazu przekazywanego przez linie danych z pamięci do procesora, może zakłócić działanie systemu. W pewnych środowiskach pracy zakłócenia elektryczne pojawiają się bardzo często, a w innych dość rzadko, jednak w świecie zdominowanym przez elektryczność i urządzenia elektryczne zakłócenia są zjawiskiem normalnym i należy się zastanawiać nie nad tym, czy się pojawią, ale kiedy? Ponieważ nawet najdrobniejsze zakłócenie może spowodować zawieszenie lub wadliwe działanie systemu, wymyślono urządzenie, które ma przywrócić normalne jego działanie. Takim układem jest *watchdog*, czyli specjalny układ czasowy. Odmierza on czas między kolejnymi wysyłanymi do niego sygnałami zerującymi. Jeżeli w wyniku zakłócenia pracy programu od wysłania ostatniego sygnału minie czas dłuższy niż zadany, to wtedy układ wysyła do mikroprocesora sygnał *Reset*. W ten sposób mikroprocesor zaczyna wykonywać program od początku, dzięki czemu system powraca do normalnej pracy. Aby układ *watchdog* działał właściwie, czyli spowodował podanie sygnału *Reset* w przypadku zaburzenia działania układu mikroprocesorowego, ale nie spowodował podania tego sygnału w trakcie poprawnej pracy, należy odpowiednio napisać program jego obsługi. Polega to na wpleceniu w program instrukcji podających sygnał zerowania czasu układu *watchdog*, informujących układ o właściwym przebiegu programu.

W praktyce można się spotkać z wieloma odmianami układów *watchdog*, ale ogólna zasada ich pracy jest zawsze podobna do opisanej powyżej.

W systemie mikroprocesorowym DSM-51 sygnałem sterującym układu *watchdog* jest sygnał linii P1_4. Rozpoczęcie pracy układu oraz jego zerowanie jest wymuszane pojawieniem się ujemnego zbocza na tej linii, a maksymalny czas między kolejnymi zerowaniami układu, określony stałą czasową, wynosi 250 ms.

Sygnał *Reset* wytworzony przez układ *watchdog* jest jednoznaczny z użyciem klawisza Reset_RAM, zatem po podaniu tego sygnału mikrokontroler rozpocznie ponowne wykonywanie programu użytkownika, umieszczonego w pamięci RAM. Schemat urządzenia *watchdog,* wraz z układami wytwarzającymi sygnały RST2 (Reset_ROM) i RST1 (Reset_RAM) oraz współpracującymi z nimi mikroprzełącznikami o nazwach Reset To EPROM i Reset To RAM, pokazano na rys. 1.5.



Rys. 1.5. Schemat układu *Watchdog* oraz układu generacji sygnałów Reset_ROM i Reset_RAM

Stanem aktywnym sygnału RST2, powodującym ustawienie dekodera adresu w stan początkowy, w którym kod programu pobieranego przez procesor będzie odczytywany z pamięci EPROM, jest jedynka logiczna, pojawiająca się na wyjściu bramki U10D. Stanem aktywnym sygnału RST1, powodującym ustawienie procesora w stan RESET, jest również jedynka logiczna, pojawiająca się na wyjściu bramki U10B. Stan aktywny sygnału RST2 może być spowodowany:

- podaniem 0 logicznego na wejście tej bramki przez zwarcie wyprowadzeń mikroprzełącznika K1;
- stanem rozładowania kondensatora C14, który będzie się ładował do napięcia 5 V (stanu jedynki logicznej) przez rezystor R9, począwszy od chwili podania zasilania; należy zwrócić uwagę, że z tego względu, po podaniu zasilania, przez czas ładowania się kondensatora jest wytwarzany nie tylko aktywny stan sygnału RST2, ale również poprzez diodę D5 sygnał RST1 powodujący RESET mikroprocesora.

  Aktywny stan sygnału RST1 może pochodzić nie tylko od wspomnianego układu R9, C14 i K1, ale również od dwu innych układów:
- mikroprzycisku K2, podającego zero logiczne na wejście bramki U10B poprzez rozładowywanie kondensatora C15, ładowanego w normalnym stanie pracy do napięcia 5 V (stanu jedynki logicznej) przez rezystor R8;
- układu *watchdog* zbudowanego na bramkach U9C, U9D, U9E, U9F oraz związanych z nimi układów RC wykorzystywanych jako elementy opóźniające, charakteryzujące się określonymi stałymi czasowymi.

  Należy zwrócić uwagę, że po podaniu zasilania układ *watchdog* jest w stanie nieaktywnym, co powoduje, że będące jego wyjściem wyjście bramki U10A jest w stanie wysokim i nie powoduje polaryzacji wejścia bramki U10B przez diodę D5 w stan aktywny 0 logicznego. W tym stanie układ wzajemnie polaryzujących się bramek U9F i U9E jest w stanie stabilnym, utrzymując stan 0 logicznego na wyjściu bramki U9E.

  Uaktywnienie układu *watchdog* następuje po zmianie stanu sygnału P1_4 na 0 logiczne, co powoduje, że na wyjściu bramki U9C pojawia się sygnał 1 logicznej, który po

przejściu przez kondensator C11 wytwarza na wejściu bramki U9D dodatni impuls o szerokości ustalanej stałą czasową wspomnianego kondensatora i rezystora R3 oraz poziomem przełączania bramki U9D. Impuls formujący się na wejściu tej bramki jest przez nią negowany i podawany na rezystor R4 jako krótkotrwały stan 0 logicznego, a przez niego na diodę D2, rozładowując kondensator C12, co powoduje:

- zmianę stanu układu bramek U9F i U9E ;
- rozpoczęcie ładowania się układu R7 i C13, stanowiącego stałą czasową układu *watchdog*.

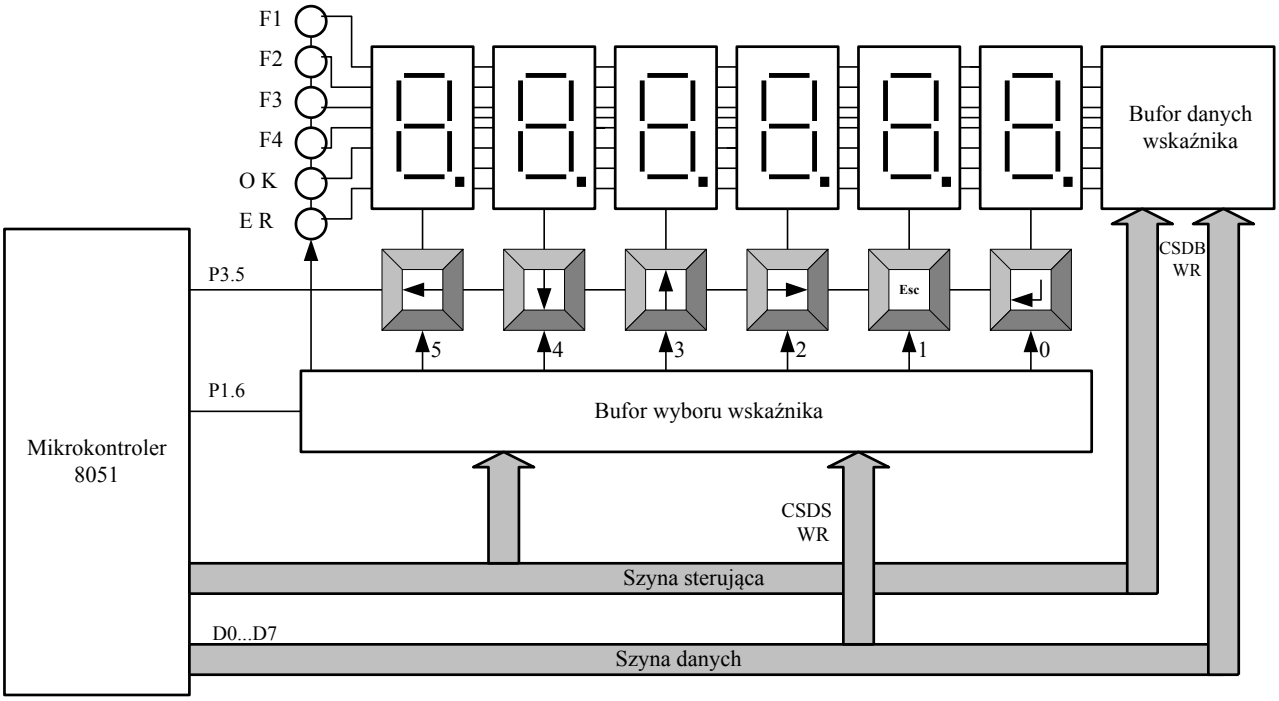
Każde kolejne pojawienie się ujemnego zbocza na linii P1_4 będzie od tej chwili powodowało wytwarzanie się krótkotrwałego stanu 0 logicznego. Stan ten przez diodę D4 będzie rozładowywał kondensator C13, nie dopuszczając, aby napięcie na nim wzrosło powyżej progu przełączenia bramki U10A.W konsekwencji podania 0 logicznego z wyjścia tej bramki poprzez diodę D5 na wejście bramki U10B, ta ostatnia wytwarza sygnał RST1 powodujący RESET procesora (RESET TO RAM.

Należy zwrócić uwagę, że pojawienie się stanu aktywnego (1 logicznej) na wyjściu RST1 spowoduje pełne naładowanie kondensatora C12 przez diodę D2 oraz rozładowanie kondensatora C13 przez bramkę U10C oraz diodę D4, co w efekcie zablokuje układ *watchdog* do chwili podania kolejnego sygnału 0 logicznego na linię P1_4.

2.7. Wyświetlacz LED i klawiatura przeglądana sekwencyjnie

System mikroprocesorowy DSM-51 został wyposażony w 7-segmentowy wyświetlacz LED, grupę 6 diod LED oraz klawiaturę przeglądaną sekwencyjnie. Schemat blokowy przyłączenia tych elementów pokazano na rys. 1.6, a schemat elektryczny na rys. 1.7.

Wyświetlacz 7-segmentowy LED składa się 6 modułów wskaźnikowych D350PA (elementy W1 do W6). Każdy z tych modułów to 8 diod LED, za pomocą których można przedstawić dowolną cyfrę. Wyświetlacz 7-segmentowy, jak pokazano na rys. 1.6, jest sterowany przez dwa bufory: wyboru wskaźnika CSDS oraz danych wskaźnika CSDB.

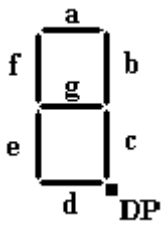


Rys. 1.6. Sterowanie wyświetlaczem 7-segmentowym, klawiatura przeglądana sekwencyjnie

- Bufor wyboru wskaźnika CSDS – dane wpisane do tego bufora określają, który wskaźnik w danej chwili ma być aktywny. Do wyboru jest 7 wskaźników oraz diody LED. Każdy bit ustawiony na 1 decyduje o świeceniu jednego wskaźnika.
- Bufor danych wskaźnika CSDB – dane umieszczone w tym buforze decydują, które segmenty wskaźnika będą aktywne w wybranym wskaźniku.

Zamiast siódmego wskaźnika do bufora wyboru wskaźnika CSDS zostały przyłączone diody led Poprzednio była pisownia LED, proszę ujednolicić w całej książce (D13 do D18). Dane umieszczone w buforze CSDB, przy wybranych w buforze wskaźnika diodach LED decydują, które diody będą świeciły. Ułożenie diod LED dla pojedynczego wskaźnika podano w tabl. 1.3.

Tablica 1.3. Opis segmentów wskaźnika i diod LED

Bufor danych wskaźnika				Bufor wyboru wskaźnika	
	bit danych **wskaźnika**	segment	led	bit wyboru **wskaźnika**	wskaźnik
	0	A	F1	0	W1
	1	B	F2	1	W2
	2	C	F3	2	W3
	3	D	F4	3	W4
	4	E	OK	4	W5
	5	F	ER	5	W6
	6	G		6	LED

I tak wysłanie do bufora wyboru wskaźnika (CSDS) bajtu o wartości binarnej 0000 1000, a do bufora danych wskaźnika (CSDB) bajtu 0000 0110 spowoduje wyświetlenie na wskaźniku W4 cyfry 1. W systemie DSM-51 do portu P3.5 przyłączono klawiaturę przeglądaną sekwencyjnie. Jest ona zbudowana z 6 klawiszy: ↵, Esc, ←, →, ↑, ↓. O tym, który klawisz aktualnie można odczytać, decyduje bufor wyboru wskaźnika. Jeżeli jest ustawiony wskaźnik nr 1, to można odczytać stan klawisza 1 (↵). Sposób przyporządkowania poszczególnych klawiszy odpowiednim bitom wpisanym do bufora wyboru wskaźnika przedstawiono w tabl. 1.4.

Tablica 1.4. Sposób wyboru klawisza

Bit wyboru wskaźnika	Odczytany klawisz
0	[↵]
1	[Esc]
2	[→]
3	[↑]
4	[↓]
5	[←]

Rys. 1.7. Schemat wyświetlacza 7-segmentowego oraz klawiatury przeglądanej sekwencyjnie

Sterowanie wskaźnikami, składającymi się z diod LED połączonych w modułach 7-segmentowych oraz w przypadku grupy diod od D13 do D18 anodami, odbywa się przez zmianę polaryzacji odpowiednich tranzystorów sterujących. Sterowanie zostało podzielone na sterowanie zasilaniem oraz sterowanie sygnałem masy.

Zasilanie każdego z modułów oraz grupy diod LED jest wykonywane przez grupę dwu tranzystorów, połączonych w układ Darlingtona. Każda z grup, np. składająca się z tranzystorów T16 i T17, podaje napięcie +5 V na połączone anody diod LED D13...D18. W tym przypadku podanie napięcia następuje po spolaryzowaniu tranzystora T17 w kierunku przewodzenia, co ma miejsce po podaniu na jego bazę potencjału niższego niż podany na jego emiter (tranzystor PNP). Taki stan wystąpi w przypadku jednoczesnego:

- spolaryzowania tranzystora T16 w kierunku przewodzenia, co nastąpi po podaniu na jego bazę potencjału wyższego niż podawany na jego emiter (tranzystor NPN), czyli po podaniu 1 logicznej na linię C6, poprzez wpisanie do zatrzasku U23 na pozycję 6 bitu wartości 1;
- podania na emiter tranzystora T16, przez rezystor R39 potencjału ok. 0 V z emitera tranzystora T18 spolaryzowanego w kierunku przewodzenia.

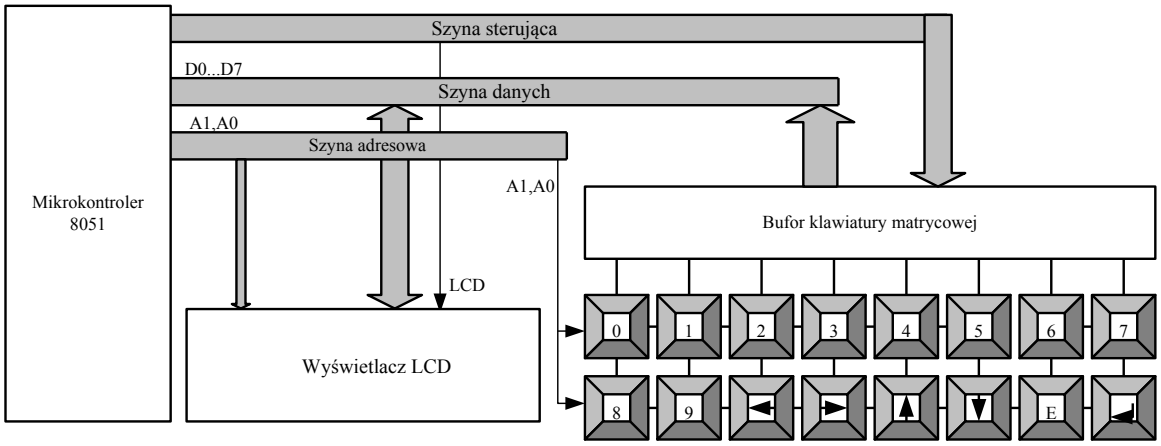
  Należy zauważyć, że spolaryzowanie tranzystora T18 w kierunku przewodzenia następuje w wyniku podania na jego bazę potencjału niższego niż podawany na jego emiter (tranzystor PNP), czyli po ustawieniu na linii P1_6 stanu 0 logicznego.

  Świecenie pojedynczej diody LED związane jest z przepływem przez nią prądu. Będzie to miało miejscew wyniku zamknięcia związanego z nią obwodu, w którym napięcie zasilania jest doprowadzane w sposób opisany powyżej. W celu zamknięcia obwodu, przez tranzystory T19...T26 oraz przyłączone do ich emiterów rezystory ograniczające, na katody wybranych diod jest podawany potencjał masy. Wybór odpowiedniego obwodu oraz związanej z nim diody następuje po podaniu 1 logicznej na wybraną linię poprzez wpisanie 1 logicznej na wybraną pozycję bitową zatrzasku U22, np. w przypadku diody D13, poprzez wpisanie do zatrzasku na pozycję 0 wartości 1. We wspomnianym przypadku, na bazę tranzystora T19 przez linię K0 zostaje podany potencjał wyższy niż podawany jest na emiter (tranzystor NPN), co powoduje jego spolaryzowanie w kierunku przewodzenia i podanie poprzez układ dwu równolegle połączonych rezystorów i złącze kolektor-emiter potencjału masy na przyłączoną do kolektora tego tranzystora katodę diody D13.

  Sterowanie klawiaturą przeglądaną sekwencyjnie polega na sprawdzaniu stanu linii P3_5 po odpowiednim ustawieniu stanu linii C0...C5, których stan jest ustawiany w sposób wyjaśniony powyżej, przy okazji opisywania sterowania zasilaniem układu diod LED. Ze względu na spolaryzowanie przez rezystor R56 na 0 logiczne linii P3_5, jej stan może być zmieniony na jedynkę jedynie w przypadku ustawienia odpowiedniego sygnału linii C0...C5 w stan 1 logicznej oraz naciśnięcia wybranego przycisku µSw_K...µSw_K9. W takim przypadku, np. po ustawieniu sygnału C0 w stan 1 logicznej i naciśnięciu przycisku µSw_K4 sygnał ten przejdzie przez diodę D19, powodując zmianę stanu na linii P3_5 na 1 logiczną.

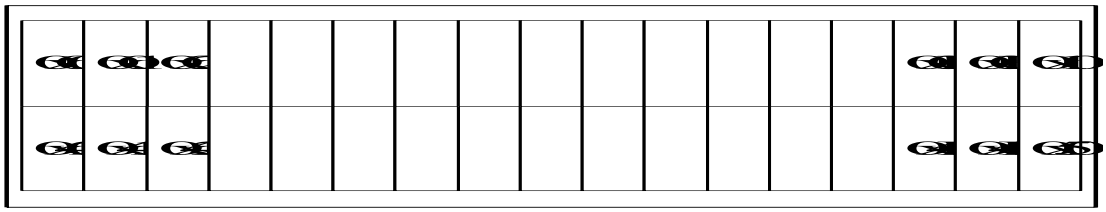
2.8. Wyświetlacz LCD

Oprócz wyświetlacza LED oraz klawiatury przeglądanej sekwencyjnie, do systemu mikroprocesorowego DSM-51 przyłączono również wyświetlacz LCD oraz klawiaturę matrycową. Blokowy schemat przyłączenia pokazano na rys. 1.8.



Rys. 1.8. Przyłączenie wyświetlacza LCD oraz klawiatury matrycowej

Wyświetlacz dostępny w systemie DSM-51 jest wyświetlaczem uniwersalnym, wyświetlającym 2 linie po 16 znaków. Wyświetlacz LCD jest wyposażony w sterownik HD44780, który zawiera dwa rodzaje pamięci kontrolowanych przez użytkownika: pamięć DD (*display data*) i pamięć CG (*character generator*). Pamięć CG składa się z dwu części: pamięci CG ROM zawierającej przygotowane przez producenta kody znaków do ekspozycji oraz pamięci CG RAM przeznaczonej na kody znaków definiowanych przez użytkownika. Obecność pamięci CG RAM jest opcjonalna. Pamięć DD RAM ma zawsze, niezależnie od modelu, jednakową wielkość, pozwalającą na wpisanie 128 znaków (2 wiersze po 64 znaki). Jeśli wyświetlacz jest krótszy i nie zawiera wszystkich znaków, to pewne pozycje w pamięci DD RAM nie mają odpowiedników na wyświetlaczu (są one pomijane przy wyświetlaniu). Sposób adresowania wyświetlanych znaków w pamięci DD RAM przez wyświetlacz LCD jest pokazany na rys. 1.9.



Rys. 1.9. Wygląd wyświetlacza LCD zastosowanego w systemie DSM-51 wraz z adresami wyświetlanych danych

We wnętrzu każdej komórki pamięci DD RAM można umieścić dowolny kod znaku zgodnego w zakresie 0x20 do 0x7f z kodami ASCII. Kody, które mogą być uwidocznione na wyświetlaczu LCD, przedstawiono w tabl. 1.5.

Tablica 1.5. Tabela znaków wyświetlacza LCD

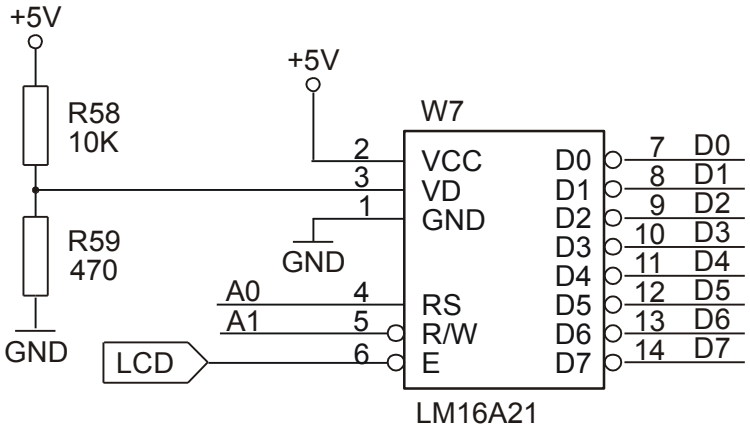
LCD	x0	x1	x2	x3	x4	x5	x6	x7	x8	x9	xA	xB	xC	xD	xE	xF
0x	(0)	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(0)	(1)	(2)	(3)	(4)	(5)	(6)	(7)
1x																
2x	∇	!	”	#	$	%	&	‘	(	)	*	+	,	-	.	/
3x	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
4x	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
5x	P	Q	R	S	T	U	V	W	X	Y	Z	[	ψ	]	^	-
6x	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
7x	p	q	r	s	t	u	v	w	x	y	z	{	|	}	←	→
8x	Znaki alfabetu japońskiego															
...																
Ex																
Fx																

Jak już wspomniano, pamięć CG RAM zawiera znaki definiowane przez użytkownika. Jest ona zorganizowana w taki sposób, że poczynając od adresu 0x00 osiem kolejnych komórek zawiera definicje bitów, które mają być uwidocznione po to, aby uzyskać definiowany kształt znaku. Znaki są wyświetlane w postaci matrycy punktów o rozmiarach 5×8.

Wyświetlacz LCD zajmuje w przestrzeni adresowej cztery kolejne komórki pamięci:

0x02ff80 – LCDWC zapis instrukcji,
0x02ff81 – LCDWD zapis danych,
0x02ff82 – LCDRC odczyt stanu,
0x02ff83 – LCDRD odczyt danych.

Po wysłaniu do wyświetlacza LCD instrukcji lub danej, musi upłynąć pewien czas, aby sterownik wykonał rozkaz lub umieścił daną pod wskazanym adresem. W tym czasie sterownik jest zajęty i nie przyjmuje kolejnych poleceń. Możliwe jest tylko odczytanie jego stanu. Przed wysłaniem kolejnego polecenia trzeba więc każdorazowo sprawdzić, czy sterownik nie jest zajęty. Schemat przyłączenia wyświetlacza LCD jest pokazany na rys. 1.10.



Rys. 1.10 Schemat przyłączenia wyświetlacza LCD

Sterownik HD44780 współpracuje z mikroporcesorem 8051 wymieniając z nim dane przez linie danych oznaczone na rys. 1.10 jako D0, D1...D7. W przypadku wymiany danych ze sterownikiem, mikroprocesor wystawia na szynę adresową jeden z 4 adresów wyświetlacza, co powoduje, że dekoder adresów ustawia sygnał LCD w stan aktywny zera logicznego. Dwa najmłodsze sygnały adresowe A0 i A1, doprowadzone do wyprowadzeń RS i R/W, służą do określenia rodzaju czynności, którą mikroprocesor chce wykonać. Stan sygnału RS określa rodzaj przesyłanych danych, którymi mogą być sygnały sterujące (instrukcje lub stan) w wypadku podania 0 logicznego lub dane w w przeciwnym wypadku. Wyprowadzenie R/W określa rodzaj operacji. W przypadku polaryzacji na 0 logiczne oznacza natomiast, że jest wykonywana operacja odczytu, dla stanu 1 logicznej oznacza, że mikroprocesor dokonuje zapisu. Na wyprowadzenie VD jest podawany sygnał napięciowy ustalający poziom kontrastu wyświetlacza.

W tablicy 1.6 pokazano zestawienie rozkazów sterujących sterownikiem HD44780 wyświetlacza LCD.

Tablica 1.6. Rozkazy wyświetlacza LCD

Rozkaz	Opis
0000 000 1	kasuj dane wyświetlacza i ustaw kursor pod adres 0
0000 00 1 X[1)]	ustaw kursor pod adres 0; ustaw dane o adresie 0 na pozycji 1 wyświetlacza
0000 0 1 I/D S	ustaw kierunek przesunięcia kursora oraz włącz lub wyłącz przesuwanie danych wyświetlacza I/D = 1 zwiększaj o 1 adres kursora po każdym wpisie lub odczycie danych I/D = 0 zmniejszaj o 1 adres kursora po każdym wpisie lub odczycie danych S = 1 jednoczesny obrót danych wyświetlacza
0000 1 D C B	włączenie i wyłączenie wyświetlacza. 1 = włączony 0 = wyłączony D - cały wyświetlacz C - tylko kursor B - mruganie znaku w pozycji kursora
000 1 S/C R/L XX	S/C = 1 obrót danych wyświetlacza S/C = 0 przesunięcie kursora R/L = 1 obrót/ przesuniecie w prawo R/L = 0 obrót/ przesuniecie w lewo

[1)] X – oznacza dowolny stan (0 lub 1)

00 1 DL N F XX	ustawienia: DL = 1 8 bitowa szyna sterująca DL = 0 4 bitowa szyna sterująca N = 1 2 linie wyświetlacza N = 0 1 linia wyświetlacza F = 1 znaki 5x10 punktów F = 0 znaki 5x7 punktów Dla DSM-51 należy ustawić 0011 10XX
0 1 A5 A4 A3 A2 A1 A0	ustawienie adresu generatora znaków(A5...A0) w liczniku adresów; po tej operacji można pisać lub czytać generator znaków z pamięci RAM
1 A6 A5 A4 A3 A2 A1 A0	ustawienie adresu danych (A6...A0) w liczniku adresów

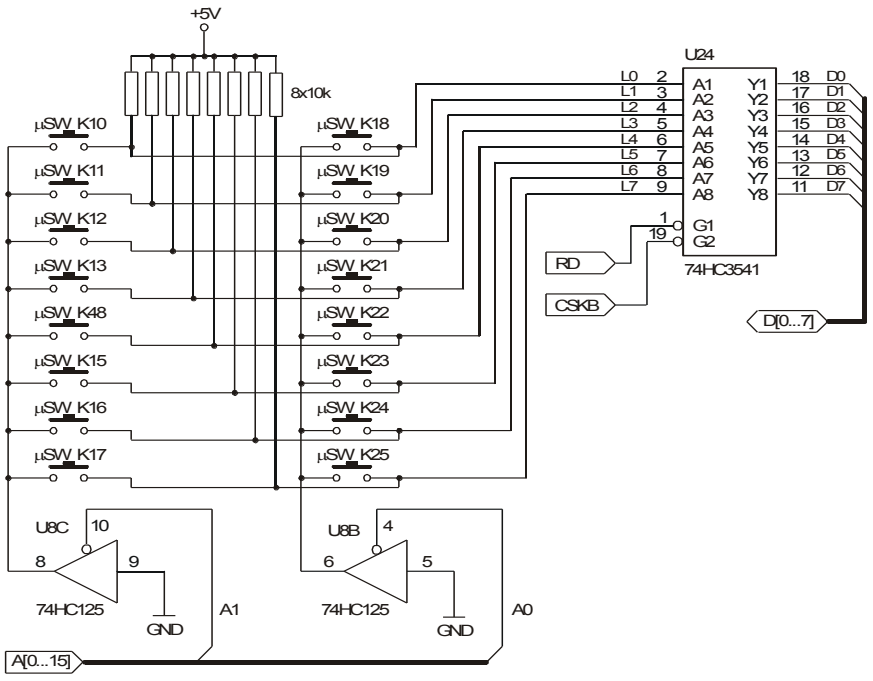
2.9. Klawiatura matrycowa

System mikroprocesorowy DSM-51 jest wyposażony w klawiaturę matrycową, składającą się z 16 klawiszy przyłączonych do systemu w sposób pokazany na rys. 1.11.

Stan klawiatury można odczytać z dwu adresów oznaczonych w tabl. 1.1. jako CSKB0 i CSKB1. Pod każdym z nich można odczytać stan ośmiu klawiszy, co daje w sumie 16 klawiszy. W tablicy 1.7 przedstawiono układ klawiszy w rejestrach CSKB0 i CSKB1. Wyboru odczytywanego zestawu klawiszy dokonuje się za pomocą dwu linii adresowych A0 i A1 (stan 0 uaktywnia związany z linią zestaw klawiszy).

Tablica 1.7. Odzwierciedlenie klawiatury matrycowej w buforach CSKB0 i CSKB1

Bit wskaźnika	Klawisz	
	odzwierciedlenie dla CSKB0 = 21H	odzwierciedlenie dla CSKB1 = 22H
0	0	8
1	1	9
2	2	←
3	3	→
4	4	↑
5	5	↓
6	6	Esc
7	7	Enter



Rys. 1.11 Schemat przyłączenia klawiatury matrycowej

Sterowanie układem polega na odczytywaniu danych z bufora trójstanowego oznaczonego jako U24, który jest uaktywniany w przypadku odwoływania się mikroprocesora do jednego ze wspomnianych adresów klawiatury matrycowej (aktywny sygnał CSKB – 0 logiczne) w cyklu odczytu sygnalizowanym przez aktywny stan sygnału RD (0 logiczne). Dane wyprowadzane przez bufor na magistralę danych pochodzą z dwu zestawów 8 klawiszy polaryzowanych z jednej strony przez rezystory napięciem zasilania. Drugie strony przycisków mają polaryzację 0 logicznego przez bramki trójstanowe U8C i U8B w przypadku, gdy na ich wyprowadzenia sterujące przyłączone do sygnałów adresowych A0 i A1 jest podawane 0 logiczne.Czy na pewno używa się sformułowania „polaryzacja przez" – wydaje mi się że tak (RK)

2.10. Kanały RS-232

W systemie mikroprocesorowym DSM-51 wbudowano dwa kanały transmisji szeregowej oznaczone na obudowie jako:

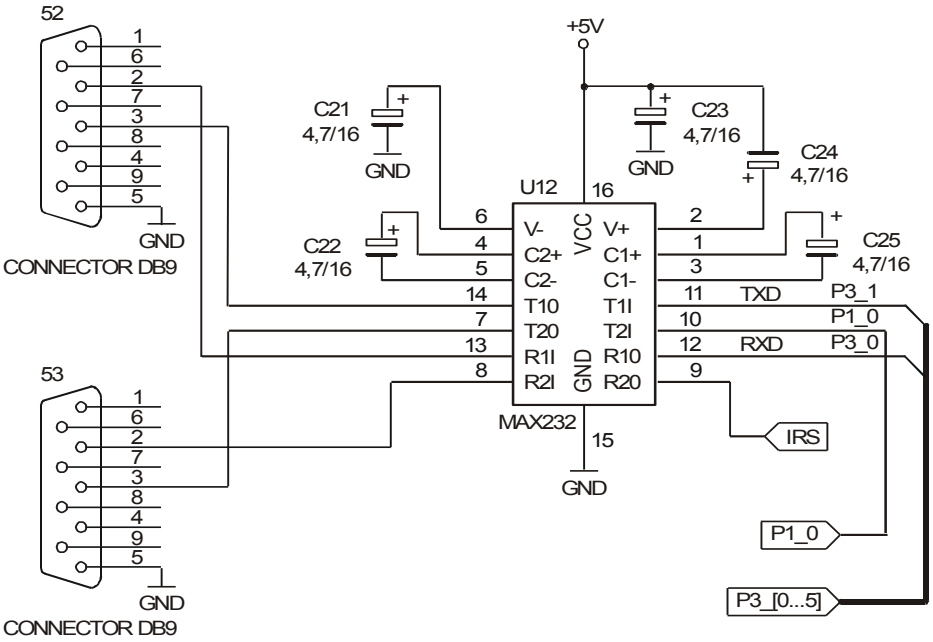
COM1 — łącze obsługiwane przez sterownik transmisji szeregowej mikrokontrolera,

COM2 — łącze obsługiwane programowo przez mikrokontroler.

COM1 jest łączem szeregowym mikrokontrolera 8051, które umożliwia prowadzenie synchronicznej lub asynchronicznej transmisji danych. Podczas transmisji asynchronicznej dane są wysyłane linią TxD (wyprowadzenie P3_1 mikroprocesora), zaś odbierane poprzez linię RxD (wyprowadzenie P3_0). Podczas transmisji synchronicznej dane są odbierane i nadawane poprzez linię RxD, a na linię TxD jest wysyłany sygnał taktujący. Podczas transmisji szeregowej za pomocą tego łącza mikrokontroler wykorzystuje wewnętrzny moduł transmisji szeregowej, nazywany blokiem lub układem UART, którego działanie jest ustalane przez ustawienie parametrów odpowiednich rejestrów wewnętrznych procesora.

COM2 jest łączem szeregowym, w którym dane wyjściowe są wysyłane z mikrokontrolera poprzez linię P1_0, sterowaną procesorem z pominięciem wewnętrznego układu UART. Z tego względu wysyłanie danych – np. bajtu, polega na ustawianiu stanu linii P1_0 zgodnie z wartością jego kolejnych bitów, a każdy bit powinien być podany na linię P1_0 w jednakowym czasie, zgodnym z przyjętą prędkością transmisji. W tym przypadku podczas programowania należy pamiętać o wysyłaniu bitu startu na początku transmisji oraz odczekaniu odpowiedniego czasu (czasu bitu stopu) po przesłaniu wszystkich bitów.

Linie mikrokontrolera nie są przyłączone bezpośrednio do złączy COMx, gdyż między mikrokontrolerem a złączem musi istnieć układ dopasowania napięć. Mikrokontroler może podać napięcia: 0 V (stan niski) lub 5 V (stan wysoki), natomiast transmisja szeregowa odbywa się na poziomie napięć standardu RS-232 wynoszących ok. +9 V (stan niski) lub –9 V (stan wysoki). Dopasowaniem poziomów napięć zajmuje się układ MAX232 oznaczony na rys. 1.12 jako U12. Układ zawiera po dwa konwertery poziomów TTL na RS-232 (wejścia TTL oznaczone jako T1I i T2I, wyjścia RS-232 oznaczone jako T1O i T2O) oraz RS-232 na TTL (wejścia RS-232 oznaczone jako R1I i R2I, wyjścia TTL oznaczone jako R1O i R2O). Kondensatory przyłączone do dodatkowych wyprowadzeń układu są wykorzystywane w przetwarzaniu napięcia zasilającego 5 V na napięcia RS-232 (+9 V i –9 V). Linie łącza szeregowego mikrokontrolera 8051 oznaczone jako TXD (P3_1) i RXD(P3_0) są przyłączone do wyprowadzeń T1I oraz R1O układu U12. Do drugiej pary konwerterów układu MAX 232 są przyłączone: sygnał mikroprocesora P1_0 oraz sygnał IRS podawany do wejścia I1 układu kontrolera przerwań (U5 na rys. 1.3).



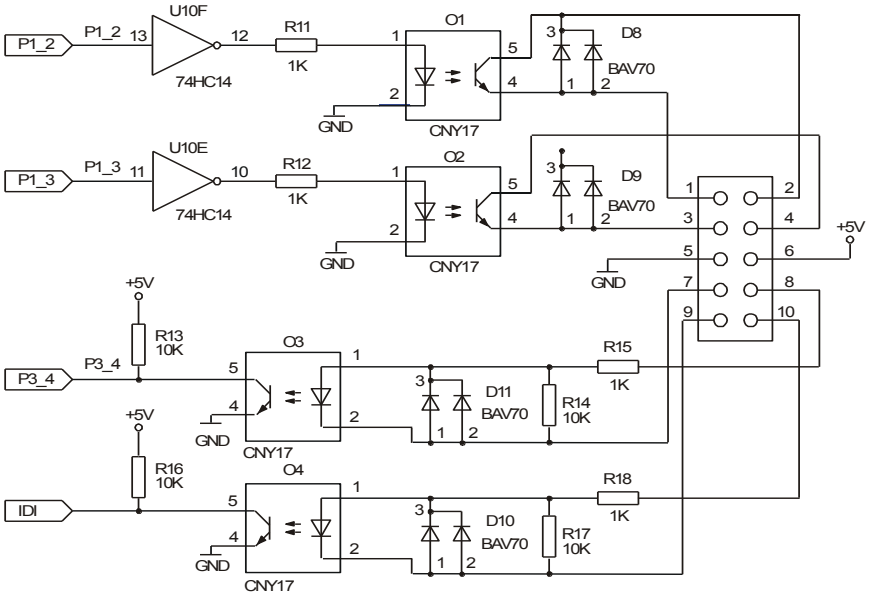
Rys. 1.12 Schemat 2 linii RS-232

2.11. Linie wejść i wyjść cyfrowych izolowane galwanicznie

W przypadku łączeniu ze sobą kilku sterowników czy urządzeń zewnętrznych, nie można wykluczyć istnienia różnicy potencjałów miedzy ich masami. Nie chodzi tu wyłącznie o istnienie składowej stałej, ale również o różne impulsy, powstające np. podczas pracy zasilaczy impulsowych. Linia izolowana galwanicznie pozwala na przesyłanie sygnałów między dwoma urządzeniami, których masy mają różne potencjały. System DSM-51 jest wyposażony w złącza, do których można przyłączyć urządzenia zewnętrzne bez łączenia mas tych urządzeń. Izolacja galwaniczna jest wykonana za pomocą 4 transoptorów CNY17 (elementy O1 do O4), przyłączonych w sposób pokazany na rys. 1.13.

Tworzą one 4 linie:

- 2 linie wyjść cyfrowych izolowanych galwanicznie (O1 i O2),
- 2 linie wejść cyfrowych izolowanych galwanicznie (O3 i O4).



Rys. 1.13 Schemat wyjść izolowanych galwanicznie

Sygnały wejść cyfrowych podawane na łączówkę przechodzą przez dzielniki napięciowe złożone z dwóch rezystorów (np. R15 i R14), które ograniczają prąd płynący przez diody transoptorów (np. transoptora O3). Diody te są zabezpieczone przed podaniem napięcia o odwrotnej polaryzacji diodami, (np. D11) umieszczonymi równolegle do diod transoptorów. Po drugiej stronie transoptorów wejściowych, w kolektorach tranzystorów, są umieszczone rezystory polaryzujące (R13 i R10). Ustalają one napięcie na wyprowadzeniu P3_4 mikroprocesora oraz IOI układu kontrolera przerwań na wartość ok. 5V, w przypadku braku oświetlenia tranzystorów, czyli braku przepływu prądu przez diody. Napięcie o tej wartości jest interpretowane jako stan 1 logicznej. W przypadku doprowadzenia napięcia wymuszającego przepływ przez diodę transoptora prądu o wartości rzędu kilku miliamperów, dioda ta zaczyna świecić powodując spolaryzowanie w kierunku przewodzenia współpracującego z nią tranzystora. W takim przypadku potencjał emitera tranzystora (potencjał masy) jest przenoszony na kolektor, powodując spolaryzowanie odpowiedniej linii na ok. 0 V, co jest jednoznaczne z ustaleniem na tej linii stanu 0 logicznego.

Działanie i konstrukcja wyjść cyfrowych izolowanych galwanicznie jest podobna do opisanej. W tym przypadku sygnał TTL pochodzący z linii P1_2 lub P1_3 mikroprocesora jest wzmacniany przez odpowiednią bramkę negującą (U10F lub U10E). Wyjście każdej z bramek jest przyłączone do rezystora połączonego szeregowo z diodą transoptora. Takie przyłączenie powoduje, że napięcie stanu 1 logicznej (5V) podawane przez bramkę, wymusza w obwodzie przepływ prądu o wartości ograniczonej rezystorem. Prąd płynąc przez diodę powoduje jej świecenie i w efekcie spolaryzowanie w kierunku przewodzenia współpracującego z nią tranzystora. W efekcie, w przypadku polaryzowania przez mikroprocesor linii P1_2 lub P1_3 na 0 logiczne, zwierane są wyprowadzenia złącza połączone z emiterem i kolektorem odpowiedniego tranzystora. Diody D8 i D9, łączące wspomniane wyprowadzenia tranzystorów, spełniają funkcję zabezpieczającą.

2.12. Przetwornik analogowo-cyfrowy A/C

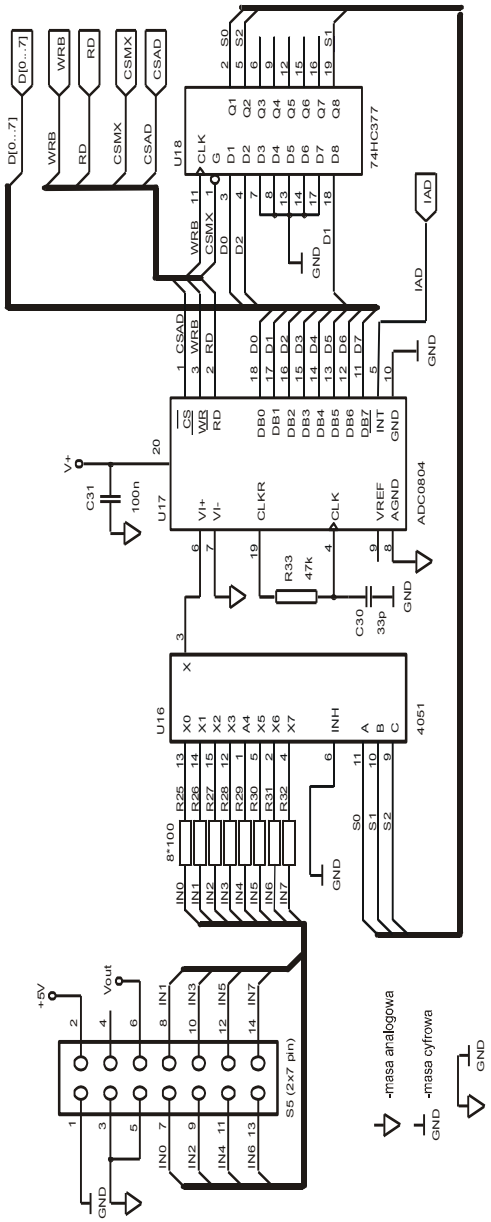
Przetwornik analogowo-cyfrowy A/C należy do grupy układów elektronicznych, które przetwarzają sygnały analogowe, np. napięcie lub prąd w odpowiednich zakresach na odpowiadające im sygnały cyfrowe np. 8 bitowe, co daje możliwość podzielenia całego zakresu na $2^8 = 256$ próbek.

Najważniejszymi parametrami przetworników A/C są:

- długość słowa: liczba bitów, na których jest podawany wynik np. 8 bitów;
- czas przetwarzania: np. 150 μs;
- zakres napięcia wejściowego, np.: 0 ÷ 5V.

Schemat przyłączenia przetwornika A/C typu ADC0804 (element U17) w systemie mikroprocesorowym DSM-51 pokazano na rys. 1.14. Sygnały są podawane na przetwornik A/C za pomocą 8-wejściowego multipleksera analogowego oznaczonego jako U10, który umożliwia przyłączenie jednego z 8 wejść oznaczonych jako IN0...IN7 do wejścia przetwornika ADC0804. Multiplekser jest sterowany zatrzaskiem 8-bitowym oznaczonym jako U18, umieszczonym pod adresem 18h (w tabl. 1.1 oznaczony jako CSMX) w przestrzeni adresowej. Do układu należy wpisać daną oznaczającą numer wejścia (0 do 7), które ma zostać przyłączone do przetwornika. W danej chwili może być przetwarzana wartość napięcia tylko z jednego wejścia. Wartość napięcia z zakresu 0 ÷ 5V zostaje przetworzona na wartości liczbowe z zakresu 00h do ffh. Przetwornik ADC0804 (element U17) jest umieszczony pod adresem 10h (CSAD) w przestrzeni adresowej urządzeń we/wy. Zapis dowolnej wartości pod ten adres powoduje rozpoczęcie przetwarzania. W chwili zakończenia przetwarzania generuje on przerwanie. Wyjście sygnału przerwania jest przyłączone do systemu przerwań DSM-51 (sygnał IAD). Odczytanie wyniku przetwarzania (z adresu CSAD) może być zrealizowane

jako reakcja na przerwanie wygenerowane przez przetwornik lub po upływie odpowiedniego czasu potrzebnego na dokonanie przetworzenia.

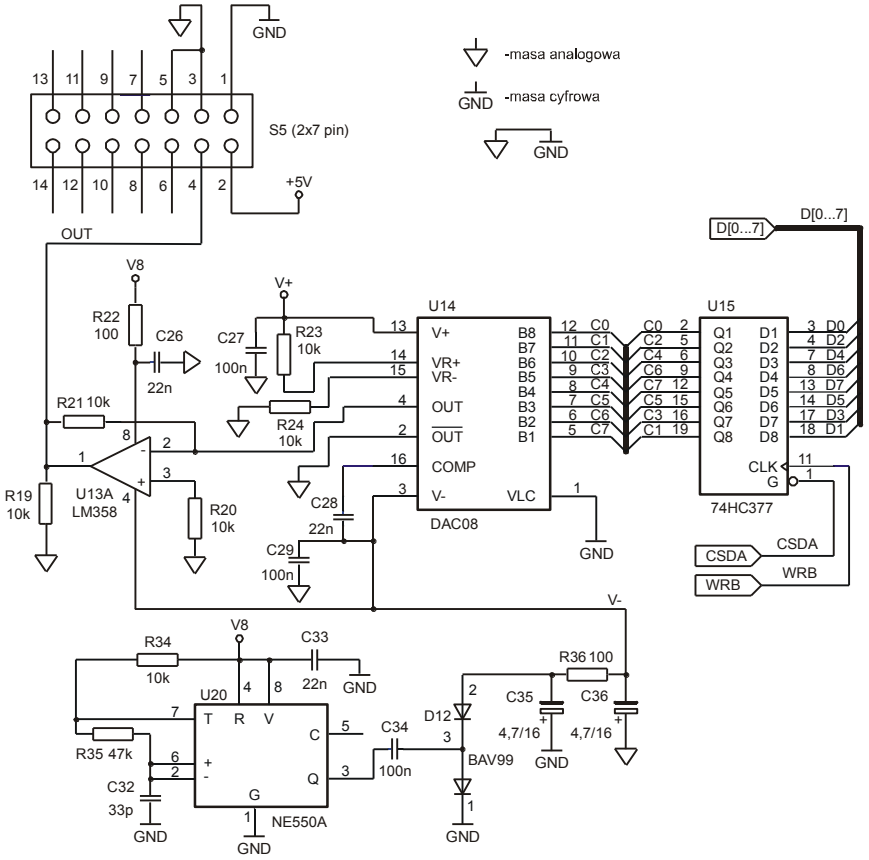


Rys. 1.14 Schemat układu z przetwornikiem A/C

2.13. Przetwornik cyfrowo-analogowy C/A

Przetwornik cyfrowo-analogowy to układ przetwarzający dyskretny sygnał cyfrowy na równoważny mu sygnał analogowy. Przetwornik ma wejście w postaci słowa o określonej liczbie bitów i zwykle jedno wyjście. Liczba bitów zależy od rozdzielczości wyjściowego sygnału analogowego jaką chce się uzyskać. Na wyjściu przetwornika pojawia się sygnał analogowy (np. w postaci napięcia), proporcjonalny do wartości liczbowej podanej na wejście oraz do napięcia odniesienia. Schemat przyłączenia przetwornika C/A DAC08 (U14) w systemie mikroprocesorowym DSM-51 pokazano na rys. 1.15. Układ ten jest przetwornikiem o rozdzielczości 8-bitowej i jednym wyjściu analogowym. Napięcie odniesienia o wartości ok. 5V jest podawane między wyprowadzenia VR+ i VR-. Do poprawnej pracy urządzenie wymaga nie tylko dodatniego, ale również ujemnego napięcia zasilania, które jest wytwarzane

w przetwornicy, wykorzystującej układ U20 (NE550A). Na wyjściu przetwornika oznaczonym IOUT pojawia się napięcie z z zakresu 0 ÷ 5V, zależne od wartości (00h do ffh) wpisanej do zatrzasku U15 podającego 8 bitów danych na wejścia przetwornika C/A. Zatrzask znajduje się pod adresem 08h oznaczonym w tabl. 1.1 jako CSDA. Sygnał wyjściowy z przetwornik jest podawany na wzmacniacz analogowy, wykorzystujący układ U13A (LM358) i z wyjścia wzmacniacza trafia na wyprowadzenia 4 łączówki S5.



Rys. 1.15. Schemat układu z przetwornikiem C/A

2.14. Linie wejść i wyjść cyfrowych

System mikroprocesorowy DSM-51 został wyposażony w układ 8255, zawierający w sobie trzy porty 8-bitowe, które mogą być indywidualnie konfigurowane. Dzięki wykorzystaniu tego układu w prosty sposób rozszerzono możliwości systemu o dodatkowe 24 linie we/wy cyfrowych. Układ może pełnić w nim funkcję portu we/wy lub też realizować transmisję równoległą. Na rysunku 1.16 pokazano połączenia układu 8255 w systemie DSM-51. Oprócz przyłączenia szyny danych, sygnałów RD i WR, sygnału wyboru o nazwie CS55 podawanego z dekodera adresów, są do niego przyłączone dwie linie adresowe A0 i A1 oraz 24 linie zewnętrzne przyłączone do złącza S6. Z tego względu, na opisany powyżej układ pracy, element 8255 zajmuje w przestrzeni adresowej mikrokontrolera cztery kolejne adresy, oznaczone w tabl. 1.1 jako CS55A, CS55B, CS55C, CS55D. Pierwsze trzy z nich, to w rzeczywistości 3 porty układu 8255 (A, B, C). Czwarty rejestr steruje trybami pracy portów, znaczenie bitów sterujących jest następujące:

7 — musi być równy 1, aby pozostałe dane zostały zaakceptowane jako konfiguracja układu 8255

6 oraz 5 — ustawiają tryb pracy portu A (rys. 1.17)

00 — tryb 0

01 — tryb 1

1x — tryb 2

4 — ustawia port A jako wejście lub wyjście

0 — wyjście

1 — wejście

3 — ustawia starsze 4 bity portu C jako wejście lub wyjście

0 — wyjście

1 — wejście

2 — ustawia tryb portu B

0 — tryb 0

1 — tryb 1

1 — ustawia port B jako wejście lub wyjście

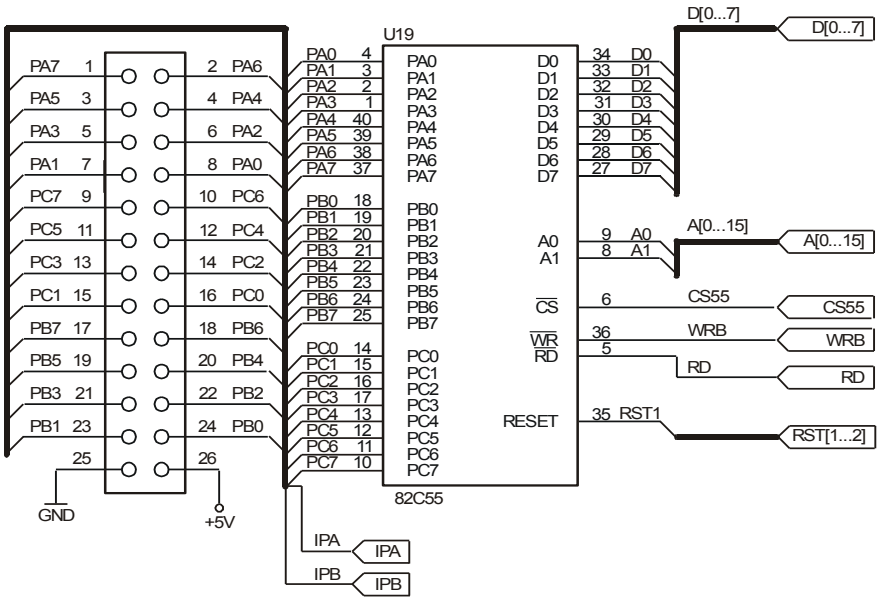
0 — wyjście

1 — wejście

0 — ustawia młodsze 4 bity poru C jako wejście lub wyjście

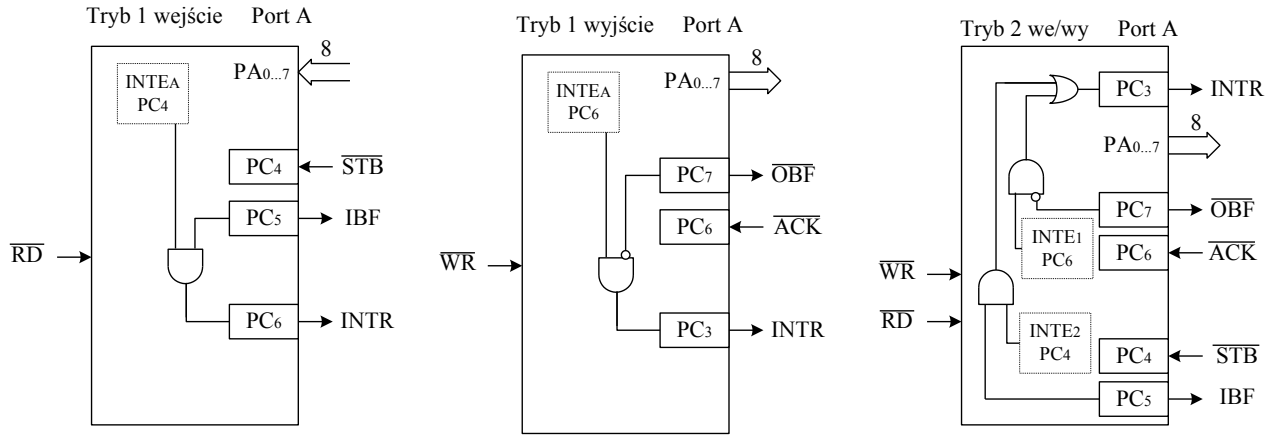
0 — wyjście

1 — wejście



Rys. 1.16. Schemat układu wejść i wyjść cyfrowych

Na rysunku 1.17. pokazano przykładową konfigurację portu A w jednym z 3 trybów.



Rys. 1.17. Przykładowa konfiguracja portu A układu 8255 w trybach 1 i 2

Noty katalogowe opisanych układów znajdują się w katalogu NOTY na załączonym dysku CD.

2.15. Wewnętrzne oprogramowanie systemu mikroprocesorowego DSM-51

Dydaktyczny System Mikroprocesorowy DSM-51 zawiera oprogramowanie systemowe, którego zadaniem jest umożliwienie użytkownikowi uruchamiania programów, bez konieczności programowania w tym celu
pamięci EPROM. Oprogramowanie to jest zawarte w pamięci EPROM systemu i pozwala na wykorzystanie systemu DSM-51 we współpracy z komputerem lub bez.

System DSM-51 może pracować jako samodzielne urządzenie lub w połączeniu z komputerem PC. Dla potrzeb niniejszej pracy system musi zostać przyłączony do PC poprzez łącze szeregowe RS-232 (np. łączówka typu DB9). W celu uruchomienia skompilowanego programu należy plik wynikowy w postaci <nazwa_pliku>.HEX wysłać z komputera do systemu DSM-51. Wykonuje się to wywołując program o nazwie `dsm51.exe` i jako parametr programu podając nazwę pliku, który ma zostać przesłany do systemu. Przykładowe wywołanie programu przesyłania pliku do systemu pokazano poniżej.

```
dsm51.exe p1.hex
```

Wynikiem takiego działania będzie wysłanie pliku o nazwie `p1.hex` do systemu DSM-51 i uruchomienie programu zawartego w wysłanym pliku. Podczas przesyłania danych plik z danymi oraz plik programu muszą znajdować się w tym samym katalogu.

Przy współpracy z komputerem oprogramowanie systemu umożliwia następujące operacje:

- przesłanie programu użytkownika z komputera do pamięci RAM systemu,
- uruchomienie tak załadowanego programu w trybie pracy ciągłej (program użytkownika przejmuje pełną kontrolę nad systemem),
- uruchomienie programu użytkownika pod kontrolą monitora (w trybie pracy krokowej) umożliwia wykonywanie programu ze śledzeniem jego pracy na ekranie komputera.

Przy wykorzystaniu systemu mikroprocesorowego DSM-51 bez komputera jego oprogramowanie umożliwia:

- wprowadzanie (edycję) programu użytkownika w asemblerze z wykorzystaniem klawiatury systemu (odbywa się to na zasadzie wyboru z kolejnych menu),
- uruchomienie wprowadzonego programu w trybie pracy ciągłej,
- uruchomienie programu w trybie pracy krokowej z możliwością obserwacji zawartości rejestrów mikrokontrolera na wyświetlaczu LCD.

Poza tym, w pamięci EPROM systemu DSM-51 zostały umieszczone podprogramy, które mają ułatwić początkującemu użytkownikowi posługiwanie się systemem. Podprogramy te obsługują wyświetlacz LCD, klawiaturę, generują opóźnienia czasowe oraz wykonują kilkubajtowe operacje mnożenia i dzielenia. W tablicy 1.8 przedstawiono ich nazwy oraz adresy początkowe oraz parametry, z których korzystają.

Razem z systemem mikroprocesorowym DSM-51 jest dostarczana dyskietka, która zawiera:

- asembler, umożliwiający przetłumaczenie programów z asemblera na kod maszynowy;
- kody źródłowe przykładowych programów dla DSM-51;
- zestaw programów wchodzących w skład kolejnych lekcji, dotyczących programowania mikrokontrolera 8051;
- programy związane z poszczególnymi modelami urządzeń zewnętrznych;
- pliki tekstowe uzupełniające opis poszczególnych programów.

Tablica 1.8. Podprogramy w pamięci EPROM systemu mikroprocesorowego DSM-51

Podprogram	Adres w pamięci EPROM	Potrzebne miejsce na stosie	Zmienne rejestry	Używane rejestry	Opis
WRITE_TEXT	0x8100	2	A, PSW, DPTR	R0	wpisuje na wyświetlacz LCD tekst wskazany rejestrem DPTR, zakończony bajtem równym 0
WRITE_DATA	0x8102	2	A,PSW	R0	wypisuje znakowo bajt z akumulatora na wyświetlacz LCD
WRITE_HEX	0x8104	3	A,PSW	R0	wypisuje szesnastkowo bajt z akumulatora na wyświetlacz LCD
WRITE_INSTR	0x8106	2	A,PSW	R0	wysyła do wyświetlacza LCD rozkaz z akumulatora
LCD_INIT	0x8108	2	A,PSW	R0	inicjuje pracę wyświetlacza LCD
LCD_OFF	0x810A	1	A,PSW	R0	wyłącza wyświetlacz
LCD_CLR	0x810C	1	A,PSW	R0	kasuje zawartość wyświetlacza LCD i ustawia kursor na początku
DELAY_US	0x810E	0	A	-	oczekuje przez czas zgodnie ze wzorem (łącznie z wywołaniem procedury): czas = (a*2+6)*12/11,0592 [μs]
DELAY_MS	0x8110	1	A	-	oczekuje przez czas A [ms] (A = 0 oznacza 256 ms)
DELAY_100MS	0x8112	2	A	-	oczekuje przez czas A*100 [ms] (A = 0 oznacza 25,6 s)
WAIT_ENTER	0x8114	4	A,PSW	R0	wypisuje na wyświetlaczu „PRESS ENTER...” i czeka na naciśnięcie klawisza
WAIT_ENTER_NW	0x8116	2	A,PSW	R0	czeka na naciśnięcie klawisza [ENTER]
TEST_ENTER	0x8118	1	A,PSW	R0	sprawdza klawisz [ENTER]; C = 0 klawisz naciśnięty C = 1 klawisz wolny
WAIT_ENT_ESC	0x811A	2	A,PSW	R0	czeka na [ENTER] lub [ESC]; zwraca informację: C = 0 [ENTER] C = 1 [ESC]
WAIT_KEY	0x811C	2	A,PSW	R0	program czeka na dowolny klawisz z klawiatury matrycowej; nr klawisza zwraca w akumulatorze
GET_NUM	0x811E	4	A,PSW	R0	wczytuje liczbę BCD (4 cyfry) z klawiatury pod adres @R0; koniec wpisywania: [ENTER] (C = 0), po 4 cyfrze również [Esc] (C = 1)

BCD_HEX	0x8120	0	A,PSW,B	R0	zamienia liczbę z postaci upakowane BCD na 2 bajtach wskazanych przez @R0 na HEX na tych bajtach
HEX_BCD	0x8122	4	A,PSW,B	R0,R2	zamienia liczbę HEX na 2 bajtach (@R0) na postać upakowane BCD (3 bajty @R0)
MUL_2_2	0x8124	1	A,PSW,B	R0,R2	mnoży 2 bajty * 2 bajty (mnożna- @R0, mnożnik- B,A(B = high), iloczyn- @R0 (4 bajty))
MUL_3_1	0x8126	1	A,PSW,B	R0	mnoży 3 bajty * 1bajt (mnożna- @R0 (3 bajty), mnożnik- A, iloczyn @R0 (4 bajty))
DIV_2_1	0x8128	1	A,PSW	R0,R2	dzieli 2 bajty przez 1 bajt (dzielna- @R0, dzielnik- B, iloraz- na dzielnej (@R0), reszta - A)
DIV_4_2	0x812A	1	PSW	A,R0,R2	dzieli 4 bajty przez 2 bajty (dzielna- @R0, dzielnik- B,A(B = high), iloraz- na dzielnej (@R0), reszta - @(R0+4), @(R0+5))

2.16. Literatura

1. R.Kowalik, R.Gazarkiewicz, „Dydaktyczny System Mikroprocesorowy – ćwiczenia w języku C”, Wydawnictwo Naukowe PWN S.A. ,Warszawa, 2006