

Politechnika  Białostocka
Wydział Elektryczny

Katedra Automatyki i Elektroniki

RODZINA MIKROKOMPUTERÓW JEDNOUKŁADOWYCH MCS51

opracował dr inż. Lech Grodzki

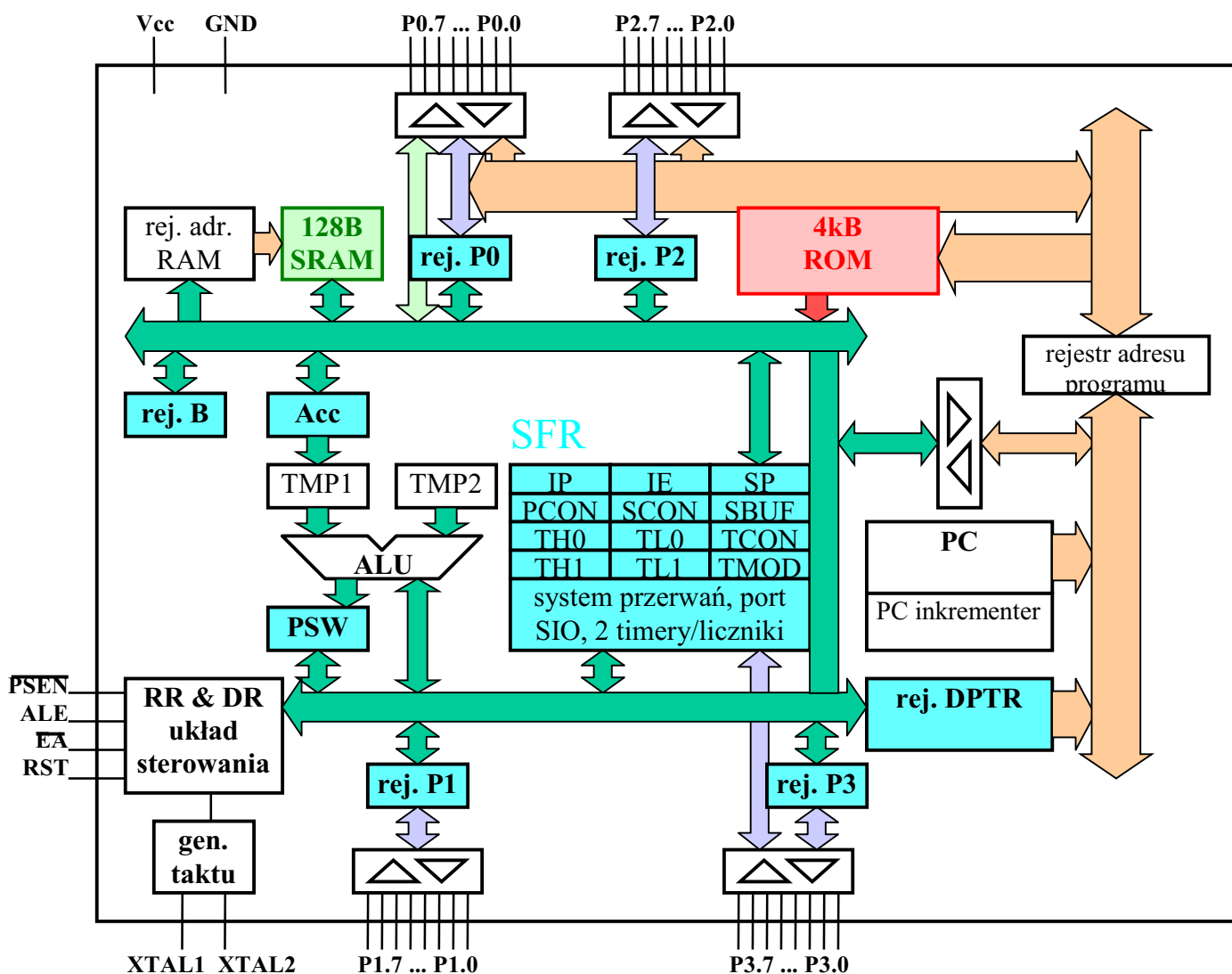
Białystok 2006

1. Krótka charakterystyka mikrokomputerów serii 8051 i ich budowa

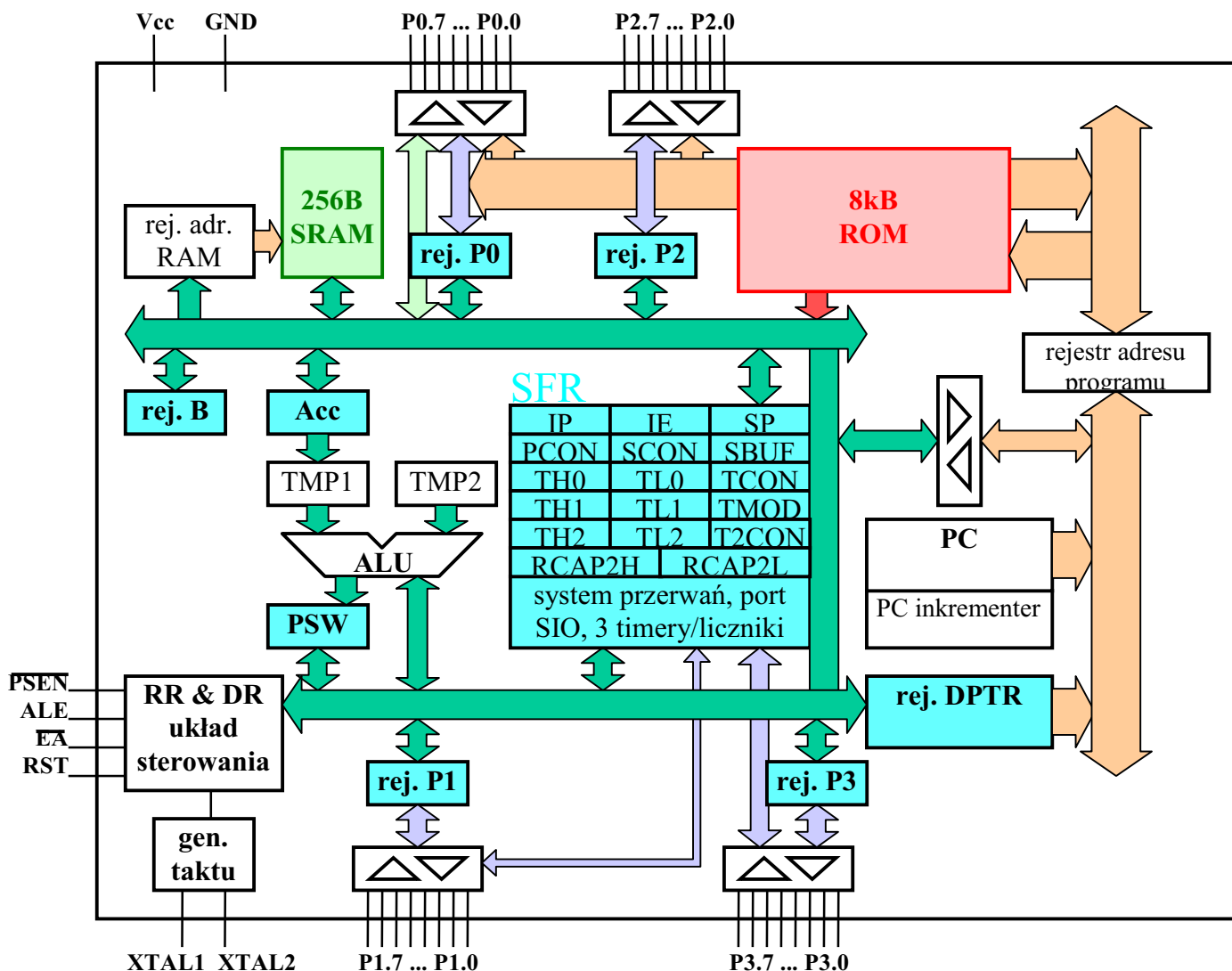
Podstawowymi reprezentantami mikrokomputerów jednoukładowych MCS51 są układy: 8031, 8051, 8751, 8032, 8052 i 8752 oraz ich wersje CMOS (obecnie w nowych konstrukcjach spotyka się już tylko układy wykonane w technologii CHMOS).

Układy serii MCS51 charakteryzują się następującymi cechami:

- ośmiobitową jednostką CPU;
- wbudowanym procesorem operacji bitowych;
- szeroką gamą 255 instrukcji, w tym rozkazów szybkiego mnożenia i dzielenia liczb 1-bajtowych;
- wbudowanym obwodem zegara systemu;
- obecnością 32 linii we/wy;
- wbudowaną wewnętrzną pamięcią RAM o pojemności 128 bajtów;
- wbudowaną wewnętrzną pamięcią ROM (8051) albo EPROM (8751) o pojemności 4kB;
- przestrzenią adresową 64kB dla zewnętrznej pamięci RAM;
- przestrzenią adresową 64kB dla zewnętrznej pamięci ROM (EPROM);
- wbudowanymi dwoma 16-bitowymi timerami;
- wbudowanym dwukierunkowym portem transmisji szeregowej;
- priorytetowym, wektorowym systemem przerwań.



Rys.1. Struktura wewnętrzna 8051



Rys.2. Struktura wewnętrzna 8052

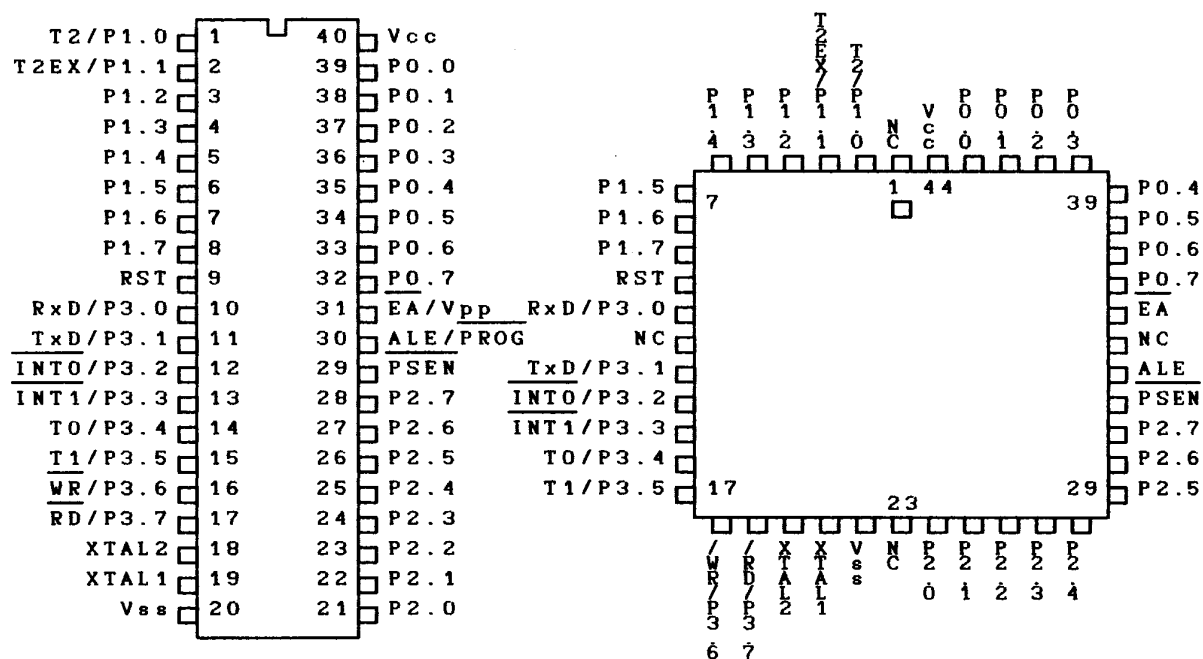
Rysunki 1 i 2 przedstawiają ogólną budowę mikrokomputerów 8051 i 8052. Z ich porównania wynika, że 8051:

- ma mniejszą pamięć wewnętrzną RAM i ROM;
- nie posiada rejestrów specjalnych: TH2, TL2, T2MOD, RCAP2H i RCAP2L;
- nie ma połączenia bloku rejestrów z buforem portu P1.

Różnice konstrukcyjne poszczególnych odmian takie jak: rodzaj i wielkość wewnętrznej pamięci programu nie zmieniają w istotny sposób struktury wewnętrznej. Aktualnie układy zaliczane do rodziny MCS51 są produkowane przez kilku dużych dostawców (Philips, Siemens, Atmel, Dallas Semiconductors) i różnią się między sobą zestawem urządzeń peryferyjnych wbudowanych w strukturę mikrokomputera, a dostępnych za pośrednictwem bloku rejestrów specjalnych SFR.

2.Opis wyprowadzeń

Obie wersje mikrokomputera ('51 i '52) mają identyczny rozkład wyprowadzeń (rysunek 3), w związku z czym mogą być stosowane zamiennie (o ile pozwalają na to różne wielkości wewnętrznych RAM i ROM).



Rys.3. Podstawowe wersje obudów układów MCS51

Sygnały zewnętrzne są następujące:

Vcc - napięcie zasilania; w wykonaniu CHMOS także doprowadzenie napięcia podtrzymującego wewnętrzną pamięć RAM;

Vss - masa zasilania;

XTAL1 - wejście wewnętrznego układu oscylatora; wejście dla sygnału z zewnętrznego oscylatora w wersji CHMOS, a w wersji HMOS winno być uziemione przy zewnętrznym oscylatorze;

XTAL2 - wyjście wewnętrznego układu oscylatora - wejście generatora zegara systemu; przy użyciu zewnętrznego oscylatora - wejście dla jego sygnału w wersji HMOS;

RST - wejście sygnału zerującego, zerowanie zachodzi gdy na RST utrzymuje się wysoki poziom napięcia przez min. 2 cykle połączenie tego pinu opornikiem 8.2kΩ z Vss i kondensatorem 10μF z Vcc zapewnia zerowanie od zasilania; dla wersji HMOS jest to także wejście napięcia podtrzymującego wewnętrzną pamięć RAM.

ALE/PROG - wyjście sygnału służącego do zapamiętania w zewnętrznym rejestrze zatraskowym młodsze bajtu adresu przy dostępie do zewnętrznej pamięci RAM/ROM; wejście impulsów programujących dla 8751/52, obciążalność 8 TTL-LS.

PSEN - wyjście impulsu odczytu z zewnętrznej pamięci programu; aktywny poziom niski; obciążalność 8 TTL-LS.

EA/Vpp - sygnał wejściowy wymuszający pobieranie rozkazów tylko z zewnętrznej pamięci programu, przy wysokim poziomie napięcia procesor sięga do zewnętrznej pamięci programu tylko po przekroczeniu obszaru wewnętrznej ROM, przy poziomie niskim - czyni to cały czas; w 8751 wejście to jest wykorzystywane do podawania napięcia programującego Up=21V.

P0.x - port 0 - ośmiobitowy, dwukierunkowy port z wyjściami typu otwarty dren, służący do multipleksowanego przesyłania mniej znaczącego bajtu adresu i bajtu danych podczas

dostępu do pamięci zewnętrznej; w trakcie programowania i weryfikacji (8751/52) służy do wprowadzania i odczytu kodu instrukcji; obciążalność 8 TTL-LS.

P1.x - port 1 - ośmiobitowy, dwukierunkowy port z wewnętrznymi opornikami łączącymi go z Vcc; w trakcie programowania i weryfikacji (8751/52) służy podaniu mniej znaczącego bajtu adresu; obciążalność: 4 TTL-LS; w przypadku 8032/52 bity zerowy i pierwszy mają dodatkowe funkcje:

T2/P1.0 - zewnętrzne źródło impulsów timera 2;

T2EX/P1.1 - wyzwianie timera 2;

P2.x - port 2 - ośmiobitowy, dwukierunkowy port z wewnętrznymi opornikami łączącymi go z Vcc, podczas dostępu do zewnętrznej pamięci RAM/ROM wysyła bardziej znaczący bajt adresu; podobnie podczas programowania i weryfikacji przyjmuje starszy bajt adresu; obciążalność 4 TTL-LS.

P3.x - port 3 - ośmiobitowy, dwukierunkowy port z wewnętrznymi opornikami łączącymi go z Vcc; obciążalność 4 TTL-LS; jego bity mają alternatywny zestaw funkcji:

RxD/P3.0 - wejście odbiornika portu szeregowego przy pracy asynchronicznej albo wejście/wyjście danych portu szeregowego przy pracy synchronicznej;

TxD/P3.1 - wyjście nadajnika portu szeregowego przy pracy asynchronicznej albo wyjście zegara przy pracy synchronicznej;

INT0/P3.2 - wejście przerwania zewnętrznego INT0 albo wejście bramkujące timera 0;

INT1/P3.3 - wejście przerwania zewnętrznego INT1 albo wejście bramkujące timera 1;

T0/P3.4 - wejście impulsów timera 0;

T1/P3.5 - wejście impulsów timera 1;

WR/P3.6 - impuls zapisu do zewnętrznej pamięci RAM lub portów;

RD/P3.7 - impuls odczytu z zewnętrznej pamięci RAM lub portów;

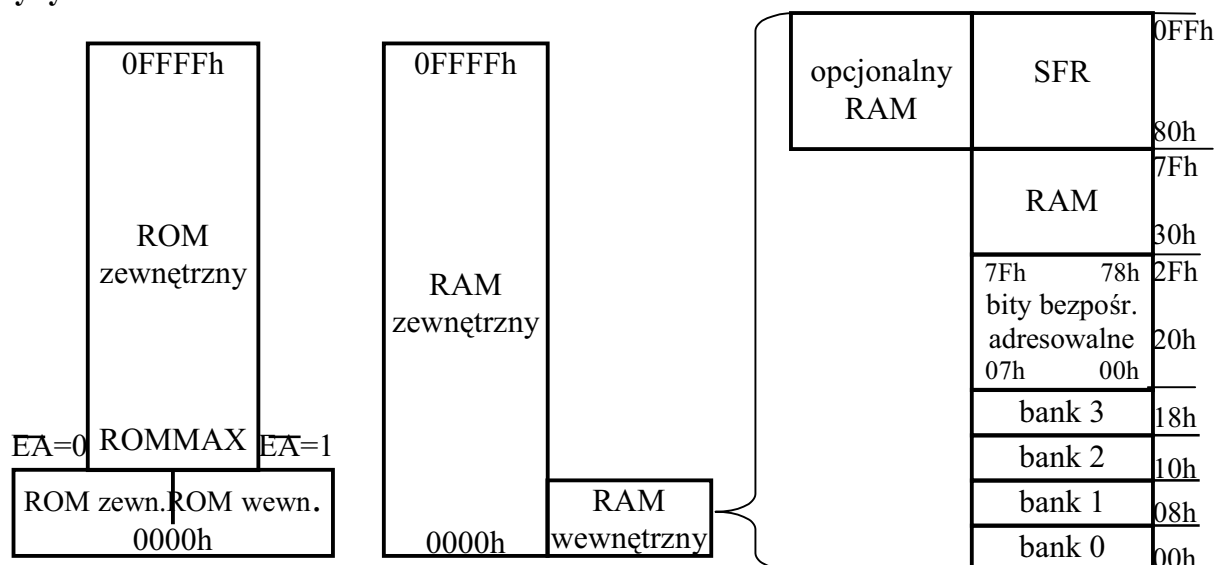
3.Organizacja pamięci

Mikrokomputery 8051/52 mają trzy przestrzenie adresowe:

- pamięć programu 64kB - składającą się z pamięci wewnętrznej 4/8kB i zewnętrznej do 64kB; jeżeli na pinie EA jest wysoki poziom napięcia to układ sięga do zewnętrznej ROM (EPROM) tylko gdy PC przekroczy wartość ROMMAX (rysunek 4), w przeciwnym przypadku czyni to dla każdej wartości PC; obszar pamięci programu o adresach 03h..23h (03h..2Bh w 8052) jest przeznaczony na procedury obsługi przerwań;
- zewnętrzna pamięć danych - zajmująca do 64kB, dostęp do niej jest możliwy za pośrednictwem specjalnych rozkazów MOVX;
- wewnętrzna pamięć danych - składająca się ze 128B pamięci RAM (256B w 8052) i bloku rejestrów specjalnych (Special Function Registers - SFR), dowolny fragment tej pamięci może być przeznaczony na obszar stosu programowego, w obrębie wewnętrznej RAM są umieszczone:
 - 4 8-bajtowe banki rejestrów;
 - 16-bajtowy obszar bitów bezpośredniego dostępu;
 - 80 bajtów (208 w 8052) ogólnego przeznaczenia.

Mapę pamięci 8051/52 ilustruje rysunek 4.

4. Tryby adresowania w 8051



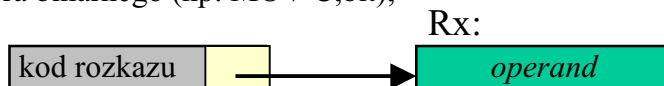
Rys.4. Przestrzeń adresowa MCS51

Możliwe są następujące tryby adresowania argumentów operacji:

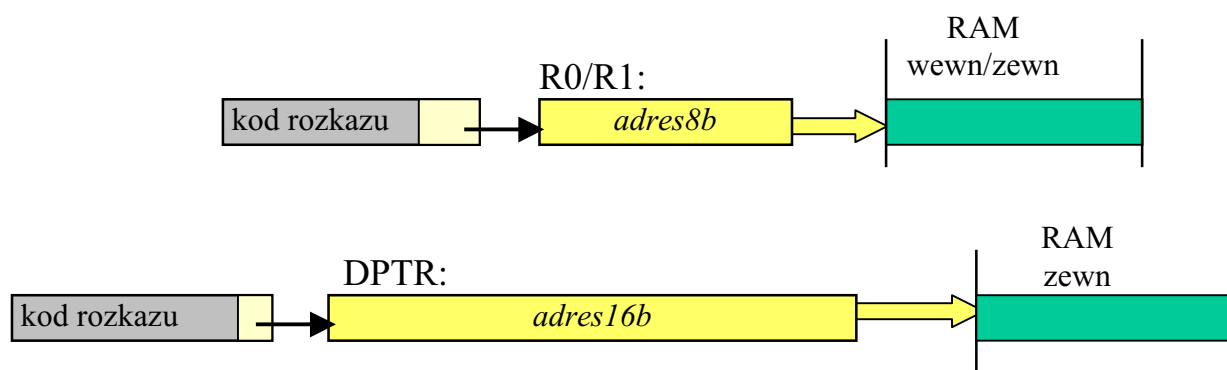
- *adresacja natychmiastowa* - argument jest umieszczony w kodzie instrukcji np.:
MOV A,#0Ch; ANL A,#0F0h;



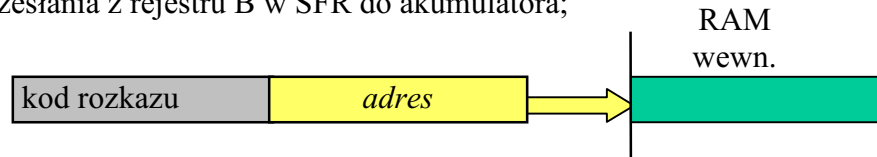
- *adresacja rejestrowa* - argument jest umieszczony we wskazanym rejestrze np.:
ADD A,R5, INC A, można adresować tą metodą rejestry R0..R7 w wybranym banku rejestrów oraz A, B i DPTR; również wskaźnik CY może być tak adresowany w instrukcjach procesora binarnego (np. MOV C,bit);



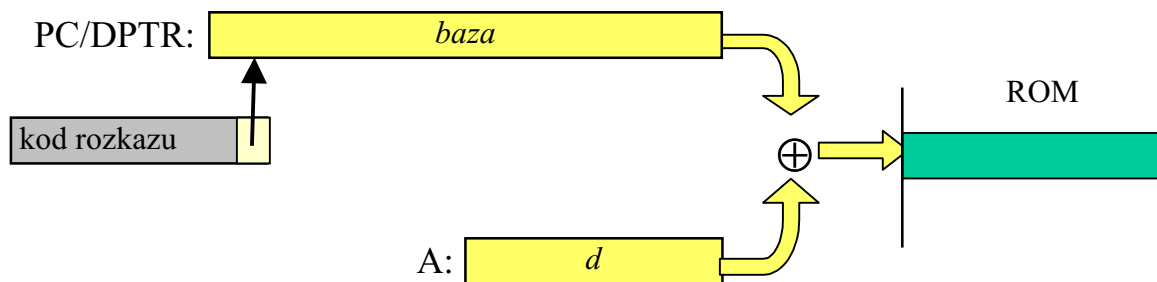
- *adresacja rejestrowa pośrednia* - argument umieszczony jest w bajcie pamięci adresowanym przez jeden z rejestrów R0 lub R1, ten sposób umożliwia dostęp do dolnych 128B wewnętrznej RAM (INC @R0) i do najniższych 256B zewnętrznej RAM (MOVX @R1,A), jest to również jedyny sposób dostępu do górnych 128B wewnętrznej RAM w odmianie 8052, stosując 16-bitowy rejestr DPTR można sięgać do całych 64kB zewnętrznej pamięci RAM (np. MOVX A,@DPTR), także dostęp do obszaru stosu rozkazami PUSH i POP jest odmianą tej adresacji;



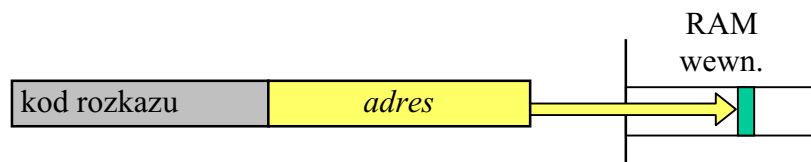
- *adresacja bezpośrednia* - argument operacji jest umieszczony w bajcie pamięci o podanym adresie, jest to jeszcze jeden tryb dostępu do dolnych 128B wewnętrznej RAM i jedyny tryb dostępu do rejestrów SFR, przykładowo instrukcje: MOV A,B i MOV A,0F0h dotyczą przesłania z rejestru B w SFR do akumulatora;



- *adresacja indeksowo-względna* - argument operacji jest umieszczony w pamięci programu pod adresem będącym sumą wartości rejestru DPTR lub PC z zawartością akumulatora, przykładami instrukcji, które działają zgodnie z tą zasadą, są: MOVC A,@A+PC i MOVC A,@A+DPTR.



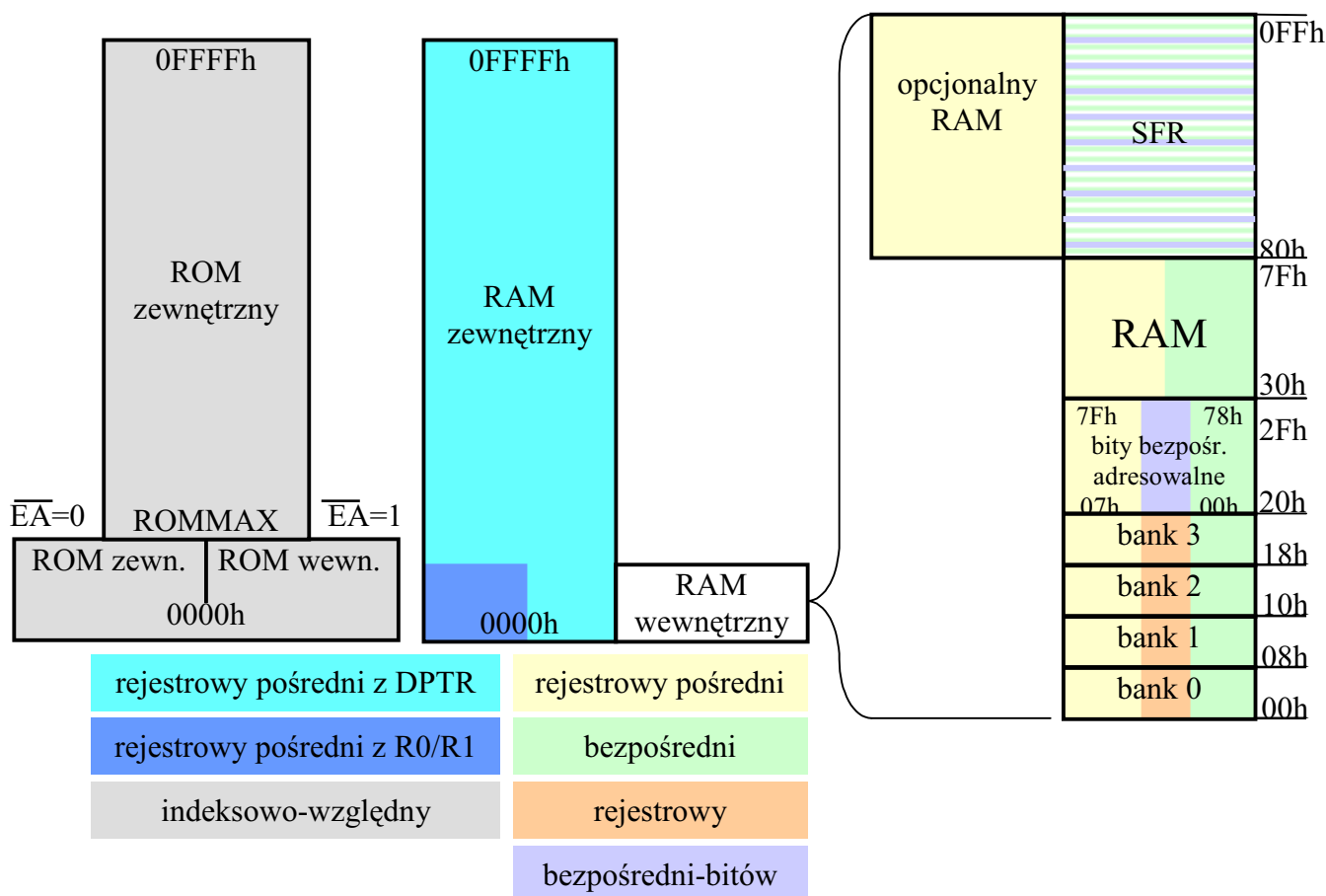
Mikrokontrolery MCS51 posiadają tzw. bity bezpośrednio adresowalne. Dostęp do nich polega na podaniu adresu bezpośredniego (0..255) takiego bitu w kodzie rozkazu:



Dostępność poszczególnych fragmentów przestrzeni adresowych MCS51 dla opisanych trybów adresowania ilustruje rysunek 5.

W odniesieniu do adresowania programu (zmiany PC) możliwe są następujące tryby:

- *adresacja stronicowa* - określa się jedynie 11 najmłodszych bitów adresu w obrębie bieżącej strony np.: AJMP a11, ACALL a11;
- *adresacja względna* - określa się 8-bitowe przesunięcie w kodzie U2 względem wartości PC, np.: JZ r8, SJMP r8;
- *adresacja bezwzględna* - określa się pełny 16-bitowy adres następnej instrukcji, np.: LJMP a16, LCALL a16;
- *adresacja pośrednia* - 16-bitowy adres następnej instrukcji jest wyznaczany jako suma zawartości DPTR i A, np.: JMP @A+DPTR.



Rys.5. Dostępność przestrzeni adresowych dla trybów adresowania operandów

5. Rejestry mikrokomputera 8051

Bajty wewnętrznej pamięci RAM o adresach 00h..1Fh składają się na cztery 8-bajtowe banki rejestrów. Wewnątrz każdego z banków rejestry są oznaczane jako R0, R1, ... R7. Wybranie odpowiedniego banku odbywa się przez ustawienie bitów RS0 i RS1 w PSW. Rejestry R0 i R1 mają specjalne przeznaczenie (niezależnie od wybranego banku): umożliwiają pośredni dostęp do całej wewnętrznej oraz 256 bajtów zewnętrznej RAM.

Struktura mikroprocesora pozwala traktować bajty wewnętrznego RAM o adresach 20h..2Fh jako zbiór 128 jednobitowych rejestrów. Dostęp do nich jest możliwy za pośrednictwem odpowiednich instrukcji tzw. procesora binarnego. Wyboru bitu dokonuje się przez podanie jego adresu, adres bitu można określić wzorem:

$$\text{adrbitu} = (\text{adrbajtu} - 20\text{h}) * 8 + \text{pozycjajwbajcie}$$

Podobnie, część rejestrów w SFR może być również traktowana jako zestawy jednobitowych rejestrów.

B	B.7 _{F7}	B.6 _{F6}	B.5 _{F5}	B.4 _{F4}	B.3 _{F3}	B.2 _{F2}	B.1 _{F1}	B.0 _{F0}	F0h
A	A.7 _{E7}	A.6 _{E6}	A.5 _{E5}	A.4 _{E4}	A.3 _{E3}	A.2 _{E2}	A.1 _{E1}	A.0 _{E0}	E0h
PSW	CY _{D7}	AC _{D6}	F0 _{D5}	RS1 _{D4}	RS0 _{D3}	OV _{D2}	- _{D1}	P _{D0}	D0h
TH2*									CDh
TL2*									CCh
RCAP2H*									CBh
RCAP2L*									CAh
T2CON*	TF2 _{CF}	EXF2 _{CE}	RCLK _{CD}	TCLK _{CC}	EXEN2 _{CB}	TR2 _{CA}	C/T2 _{C9}	CP/RL2 _{C8}	C8h
IP	- _{BF}	- _{BE}	PT2* _{BD}	PS _{BC}	PT1 _{BB}	PX1 _{BA}	PT0 _{B9}	PX0 _{B8}	B8h
P3	P3.7 _{B7}	P3.6 _{B6}	P3.5 _{B5}	P3.4 _{B4}	P3.3 _{B3}	P3.2 _{B2}	P3.1 _{B1}	P3.0 _{B0}	B0h
IE	EA _{AF}	- _{AE}	ET2* _{AD}	ES _{AC}	ET1 _{AB}	EX1 _{AA}	ET0 _{A9}	EX0 _{A8}	A8h
P2	P2.7 _{A7}	P2.6 _{A6}	P2.5 _{A5}	P2.4 _{A4}	P2.3 _{A3}	P2.2 _{A2}	P2.1 _{A1}	P2.0 _{A0}	A0h
SBUF									99h
SCON	SM0 _{9F}	SM1 _{9E}	SM2 _{9D}	REN _{9C}	TB8 _{9B}	RB8 _{9A}	TI ₉₉	RI ₉₈	98h
P1	P1.7 ₉₇	P1.6 ₉₆	P1.5 ₉₅	P1.4 ₉₄	P1.3 ₉₃	P1.2 ₉₂	P1.1 ₉₁	P1.0 ₉₀	90h
TH1									8Dh
TH0									8Ch
TL1									8Bh
TL0									8Ah
TMOD	GATE	C/T	M1	M0	GATE	C/T	M1	M0	89h
TCON	TF1 _{8F}	TR1 _{8E}	TF0 _{8D}	TR0 _{8C}	IE1 _{8B}	IT1 _{8A}	IE0 ₈₉	IT0 ₈₈	88h
PCON	PCON	-	-	-	GF1	GF0	PD	IDL	87h
DPH									83h
DPL									82h
SP									81h
P0	P0.7 ₈₇	P0.6 ₈₆	P0.5 ₈₅	P0.4 ₈₄	P0.3 ₈₃	P0.2 ₈₂	P0.1 ₈₁	P0.0 ₈₀	80h

Rys.6. Struktura bloku SFR

Najistotniejszymi dla pracy procesora są rejestry bloku SFR: akumulator, rejestry sterujące, rejestry pomocnicze i rejestry portów zewnętrznych. Mikrokomputer 8052, w porównaniu z 8051, ma w SFR dodatkowe rejestry, związane z trzecim układem timera: TL2, TH2, RCAP2L, RCAP2H i T2CON. Rysunek 5 przedstawia strukturę bloku SFR. Część z rejestrów ma adresowane bezpośrednio bity (w prawym dolnym rogu podany jest wtedy adres szesnastkowy bitu). Jedynie bity rejestrów PCON i TMOD mimo ich wyróżnienia nie mogą być adresowane bezpośrednio. Adres bitu może być określany trojako: nazwa bitu, nazwa rejestru.numer bitu, adres szesnastkowy. Przykładowo zapisy: AC, PSW.6 i 0D6h oznaczają ten sam bit przeniesienia pomocniczego. Na wymienionym rysunku bity oznaczone kreską są niedostępne. Dodatkowe rejestry i bity występujące w SFR w wersji 8052 są na rysunek 5 oznaczone gwiazdkami. Poszczególne rejestry bloku SFR zostaną teraz dokładniej przedstawione. Obok symbolu rejestru, jego nazwy i adresu, podane zostaną bliższe informacje na temat jego wykorzystania. Gwiazdką wyróżniono bity i rejestry występujące wyłącznie w 8052.

A - akumulator, 0E0h.

Jest to podstawowy rejestr dla operacji arytmetycznych i logicznych. Umożliwia również dostęp indeksowy do pamięci RAM i ROM.

B - rejestr pomocniczy, 0F0h.

Współpracuje z A w operacjach mnożenia i dzielenia.

PSW - słowo stanu, 0D0h.

Jego bity mają następujące znaczenie:

CY	AC	F0	RS1	RS0	OV	-	P
----	----	----	-----	-----	----	---	---

CY - flaga przeniesienia;

AC - flaga przeniesienia pomocniczego;

F0 - flaga przeznaczona dla użytkownika;

RS1,RS0 - bity wybierające bank rejestrów:

RS1,RS0 = 0,0 - bank 0 (rejestry o adresach 00..07h),

0,1 - bank 1 (rejestry o adresach 08..0Fh),

1,0 - bank 2 (rejestry o adresach 10..17h),

1,1 - bank 3 (rejestry o adresach 18..1Fh);

OV - flaga przepełnienia;

P - flaga parzystości liczby w A;

Wpływ rozkazów mikroprocesora na wymienione flagi jest opisany przy okazji omawiania listy rozkazów.

SP - wskaźnik stosu, 81h.

Jest to ośmiobitowy rejestr przechowujący adres ostatniego zajętego bajtu stosu. Stos może być ulokowany w dowolnym miejscu wewnętrznej pamięci RAM. Rejestr SP jest inkrementowany przy instrukcji PUSH i wywołaniu procedury, oraz dekrementowany przy rozkazie POP i powrocie z procedury.

DPH,DPL (DPTR) - rejestr adresu danych, 83h,82h.

Dwa kolejne rejestry DPH i DPL przechowują dwubajtowy adres, który może być wykorzystywany przy dostępie do zewnętrznej RAM lub do ROM jako pamięci danych.

PCON - rejestr kontrolny zasilania, 87h.

Jego bity mają następujące znaczenie:

SMOD	-	-	-	GF1	GF0	PD	IDL
------	---	---	---	-----	-----	----	-----

SMOD - bit zdwojenia częstotliwości pracy portu szeregowego w trybach 1, 2 i 3; gdy jest on ustawiony transmisja odbywa się z dwukrotnie większą szybkością;

GF1,GF0 - flagi przeznaczone dla użytkownika;

PD - ustawienie tego bitu „usypia” mikroprocesor (maksymalne obniżenie poboru prądu zasilającego);

IDL - ustawienie tego bitu wprowadza mikroprocesor w stan czuwania (obniża pobór prądu zasilającego). Przy jednoczesnym wpisie 1 do bitów PD i IDL dominującym jest bit PD. Więcej szczegółów na temat pracy procesora przy obniżonym poborze prądu znajduje się w rozdz. 12.

UWAGA: wiele opracowań (z niektórymi katalogami firmowymi włącznie) podaje błędnie adres PCON jako 97h.

IE - rejestr maskowania przerwań, 0A8h.

Jego bity służą do włączania/wyłączania przerwań:

EA	-	ET2	ES	ET1	EX1	ET0	EX0
----	---	-----	----	-----	-----	-----	-----

Ustawienie danego bitu odblokowuje wybrane przerwanie, a jego wyzerowanie - blokuje je.

Poszczególne bity dotyczą:

EA - wszystkich przerwań;

ET2 - przerwań od timera 2;

ES - przerwań od portu szeregowego;

ET1 - przerwań od timera 1;

EX1 - przerwań zewnętrznych od INT1;

ET0 - przerwań od timera 0;

EX0 - przerwań zewnętrznych od INT0.

IP - rejestr priorytetów przerwań, 0B8h.

Jego bity pozwalają określić priorytety przerwań:

-	-	PT2	PS	PT1	PX1	PT0	PX0
---	---	-----	----	-----	-----	-----	-----

Ustawienie danego bitu przypisuje przerwaniu wyższy, a wyzerowanie - niższy z dwóch możliwych priorytetów. Poszczególne bity dotyczą:

PS - przerwań od portu szeregowego;

PT2 - przerwań od timera 2;

PT1 - przerwań od timera 1;

PX1 - przerwań zewnętrznych od INT1;

PT0 - przerwań od timera 0;

PX0 - przerwań zewnętrznych od INT0.

Więcej informacji na temat przerwań znajduje się w rozdz. 7.

TH0 - starszy bajt timera 0, 8Ch.

TL0 - młodszy bajt timera 0, 8Ah.

TH1 - starszy bajt timera 1, 8Dh.

TL1 - młodszy bajt timera 1, 8Bh.

TMOD - rejestr trybów pracy timerów, 89h.

Jego bity służą do określania trybów pracy obu timerów:

GATE	C/T	M1	M0	GATE	C/T	M1	M0
------	-----	----	----	------	-----	----	----

Oznaczenia bitów powtarzają się, ponieważ starsza połowa tego rejestru dotyczy timera 1, a młodsza - timera 0. Znaczenie bitów jest następujące:

GATE - ustawienie tego bitu włącza bramkowanie wejścia timera Tx przez odpowiednie wejście INTx, a wyzerowanie tego bitu - uniezależnia wejście timera od wejścia INTx;

C/T - bit określający rodzaj pracy, ustawiony dołącza wejście timera do odpowiedniego pinu Tx, a wyzerowany - do układu oscylatora;

M1,M0 - definiują tryb pracy timera:

M1,M0 = 0,0 - młodszy bajt timera pracuje jako licznik modulo 32;

0,1 - młodszy i starszy bajt timera tworzą 16-bitowy licznik;

1,0 - młodszy bajt jest 8-bitowym licznikiem z automatycznym

przeładowaniem wartością początkową pamiętaną w starszym bajcie;

1,1 - dla timera 0: TL0 jest 8-bitowym licznikiem sterowanym przez bity

kontrolne timera 0, a TH0 jest 8-bitowym licznikiem sterowanym przez bity kontrolne timera 1

1,1 - dla timera 1: wyłącza timer 1.

Więcej informacji na temat timerów znajduje się w rozdziale 8.

TCON - rejestr kontrolny timerów i przerwań zewnętrznych, 88h.

Jego bity służą do kontroli pracy timerów i przerwań zewn.:

TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0
-----	-----	-----	-----	-----	-----	-----	-----

TF1 - flaga przepełnienia w timerze 1, ustawiana sprzętowo po wystąpieniu przepełnienia, kasowana również sprzętowo po wejściu w procedurę obsługi przerwania od timera 1;

TR1 - ustawiany/kasowany programowo bit startu/stopu timera 1;

TF0 - flaga przepełnienia w timerze 0, ustawiana sprzętowo po wystąpieniu przepełnienia, kasowana również sprzętowo po wejściu w procedurę obsługi przerwania od timera 0;

TR0 - ustawiany/kasowany programowo bit startu/stopu timera 0;

IE1 - flaga przerwania od INT1, ustawiana sprzętowo po wykryciu przerwania, kasowana również sprzętowo po wejściu w procedurę obsługi przerwania, jeżeli przerwanie to jest aktywne tylko opadającym zboczem sygnału;

IT1 - ustawiany/kasowany programowo bit wybierający kształt sygnału przerywającego na INT1, przy IT1=1 wykrywane jest opadające zbocze, a przy IT1=0 poziom niski sygnału;

IE0 - flaga przerwania od INT0, ustawiana sprzętowo po wykryciu przerwania, kasowana również sprzętowo po wejściu w procedurę obsługi przerwania, jeżeli przerwanie to jest aktywne tylko opadającym zboczem sygnału;

IT0 - ustawiany/kasowany programowo bit wybierający kształt sygnału przerywającego na INT0, przy IT0=1 wykrywane jest opadające zbocze, a przy IT0=0 poziom niski sygnału.

TH2 - starszy bajt timera 2, 0CDh.

TL2 - młodszy bajt timera 2, 0CCh.

RCAP2H - starszy bajt rejestru pamiętającego timera 2, 0CBh.

RCAP2L - młodszy bajt rejestru pamiętającego timera 2, 0CAh.

T2CON - rejestr kontrolny timera 2, 0C8h.

Jego bity służą sterowaniu pracą timera 2 (tylko w 8052):

TF2	EXF2	RCLK	TCLK	EXEN2	TR2	C/T2	CP/RL2
-----	------	------	------	-------	-----	------	--------

TF2 - flaga przepełnienia w timerze 2, ustawiana sprzętowo po wystąpieniu przepełnienia pod warunkiem, że bity RCLK i TCLK są wyzerowane, może być skasowana tylko programowo;

EXF2 - flaga zewnętrzna ustawiana przy przeładowaniu albo zapamiętaniu stanu timera 2, spowodowanym pojawieniem się ujemnego zbocza sygnału na pinie T2EX (P1.1) przy EXEN2 = 1, jeżeli przerwania od timera 2 są odblokowane to ustawienie się tej flagi powoduje wejście w obsługę odpowiedniego przerwania, może być kasowana tylko programowo;

RCLK - ustawiana i kasowana wyłącznie programowo flaga źródła zegara odbiornika portu szeregowego w modach 1 i 3:

RCLK = 0 - źródłem są przepełnienia w timerze 1;

RCLK = 1 - źródłem są przepełnienia w timerze 2;

TCLK - ustawiana i kasowana wyłącznie programowo flaga źródła zegara nadajnika portu szeregowego w modach 1 i 3:

TCLK = 0 - źródłem są przepełnienia w timerze 1;

TCLK = 1 - źródłem są przepełnienia w timerze 2;

EXEN2 - flaga odblokowująca (gdy ustawiona) wejście T2EX, gdy jest wyzerowana uniemożliwia sterowanie timerem 2 za pośrednictwem pinu T2EX, ustawiana i kasowana tylko programowo;

TR2 - ustawiany/kasowany programowo bit startu/stopu timera 2;

C/T2 - bit określający rodzaj pracy, ustawiony dołącza wejście timera do pinu T2, a wyzerowany - do układu oscylatora;

CP/RL2 - określa tryb pracy timera 2:

gdy ustawiony i bit EXEN2 = 1 to ujemne zbocze sygnału na pinie T2EX spowoduje zapamiętanie aktualnego stanu TH2/TL2 w RCAP2H/RCAP2L; gdy wyzerowany i RCLK = 0 i TCLK = 0 to timer 2 pracuje z automatycznym przeładowaniem po wystąpieniu przepełnienia lub pojawieniu się ujemnego zbocza na wejściu T2EX przy EXEN2 = 1; jeżeli bity RCLK lub TCLK są ustawione to bit CP/RL2 jest ignorowany i przepełnienie w timerze 2 powoduje jego automatyczne przeładowanie.

SCON - rejestr kontrolny portu szeregowego, 98h.

Jego bity mają następujące znaczenie:

SM0	SM1	SM2	REN	TB8	RB8	TI	RI
-----	-----	-----	-----	-----	-----	----	----

SM0, SM1 - bity określające mod pracy portu szeregowego:

SM0, SM1 = 0,0 - mod 0;

0,1 - mod 1;

1,0 - mod 2;

1,1 - mod 3;

SM2 - wpływa na pracę łącza szeregowego zależnie od ustawionego modu pracy:
 mod 0 - winien być skasowany;
 mod 1 - ustawiony wymusza oczekiwanie na poprawny bit stopu;
 mod 2 i 3 - ustawiony powoduje ignorowanie odebranej danej, jeżeli jej bit dziewiąty był zerem;

REN - bit ustawiany/zerowany programowo włączający/wyłączający odbiór szeregowy;

TB8 - wpisywany programowo dziewiąty bit nadawanej informacji (w modzie 2 lub 3);

RB8 - odebrany (w modzie 2 lub 3) dziewiąty bit informacji, w modzie 1 - odebrany bit stopu, w modzie 0 - nie używany;

TI - flaga przerwania od nadajnika, ustawiana sprzętowo po wysłaniu ostatniego bitu, musi być kasowana programowo w celu kontynuacji pracy nadajnika;

RI - flaga przerwania od odbiornika, ustawiana sprzętowo po odebraniu ostatniego bitu, musi być kasowana programowo w celu kontynuacji pracy odbiornika.

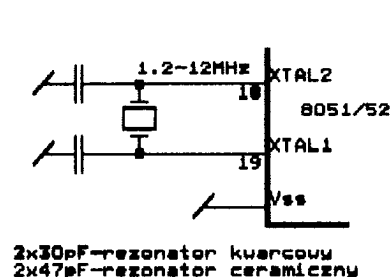
SBUF - rejestr danych portu szeregowego, 99h.

Zapis do tego rejestru powoduje w rzeczywistości zapis do bufora nadajnika, a odczyt z niego - odczyt bufora odbiornika.

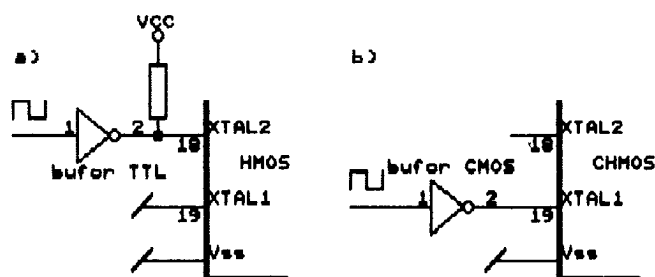
P0, P1, P2, P3 - rejestry portów zewn., 80h, 90h, 0A0h, 0B0h.

Bliższe informacje na temat tych rejestrów znajdują się w rozdz. 10.

6. Cykl maszynowy i rozkazowy



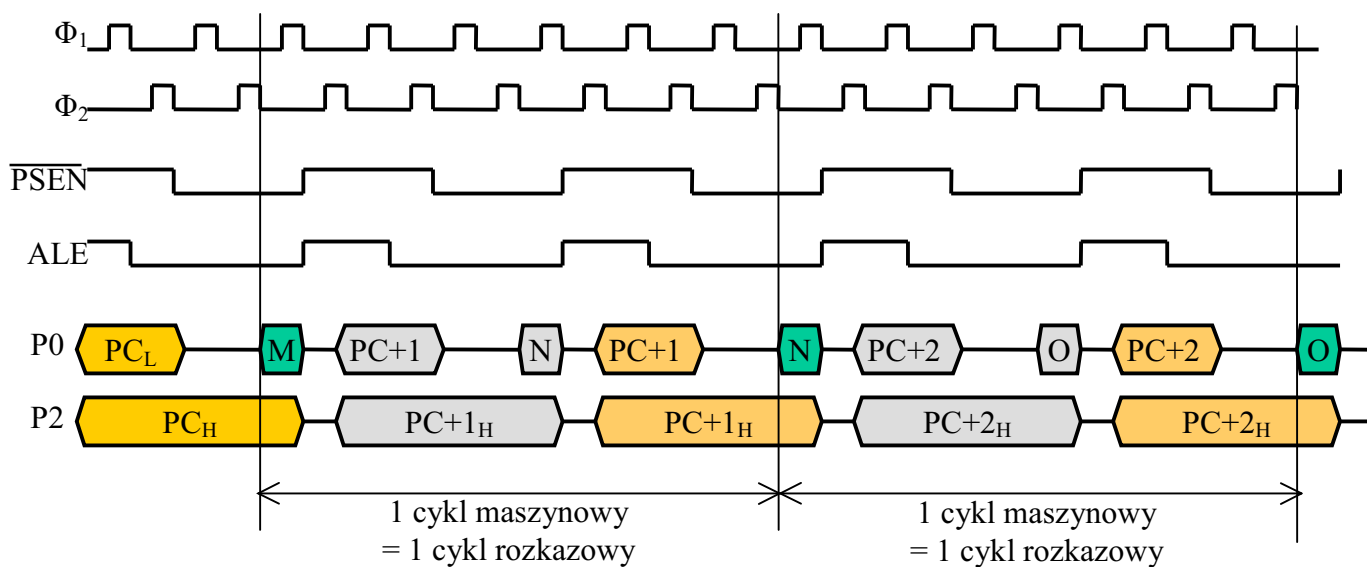
Rys.7. Wykorzystanie oscylatora



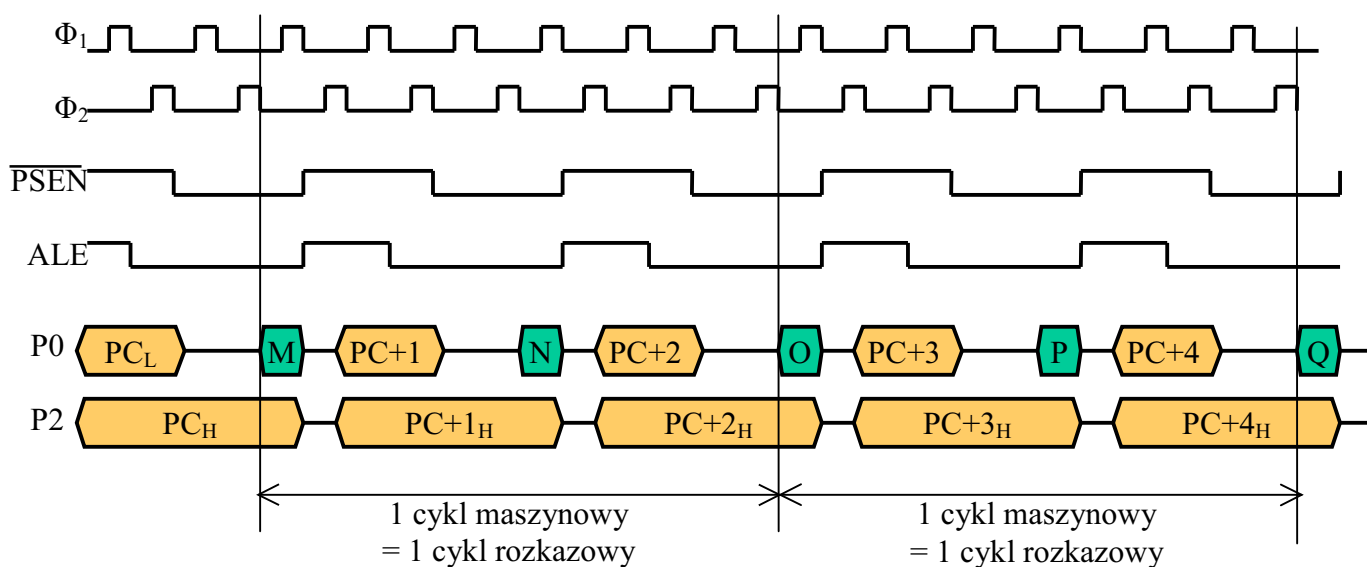
Rys.8. Wykorzystanie generatora

Wewnętrzny sygnał taktujący pracę mikrokomputera 8051/52 jest generowany przez wbudowany układ, który wymaga jedynie zewnętrznego oscylatora o częstotliwości 1,2..12MHz. Oscylator ten dołącza się do wyprowadzeń XTAL1 i XTAL2 wraz z kondensatorami (rysunek 7). Wartości kondensatorów zależą od rodzaju oscylatora (kwarcowy lub ceramiczny). Można również użyć zewnętrznego generatora przebiegu prostokątnego o wypełnieniu $\frac{1}{2}$. Wyjście tego generatora, poprzez bufor, łączy się wtedy z wejściem XTAL1 albo XTAL2, zależnie od technologii wykonania mikroprocesora (rysunek 8).

Wewnętrzny obwód 8051, na podstawie sygnałów z wejść XTAL1 i XTAL2, wytwarza dwufazowy sygnał synchronizujący pracę mikrokomputera. Obie fazy mają dwukrotnie mniejszą częstotliwość, niż sygnał wejściowy, i różne od $\frac{1}{2}$ wypełnienie. Pojedynczy okres dwufazowego sygnału taktującego jest określany mianem stanu (S1..S6), na który składają się dwie fazy (P1, P2). Sześć kolejnych stanów to tzw. cykl maszynowy. Łącznie jeden cykl maszynowy składa się z 12 odcinków czasu, które zwykło się oznaczać: S1P1, S1P2, S2P1..S6P2. Zewnętrzne sygnały sterujące takie jak: PSEN, ALE, RD i WR są ściśle powiązane z poszczególnymi stanami i fazami cyklu maszynowego. Relacje te prezentują rysunki 9a-e i 10a-b.

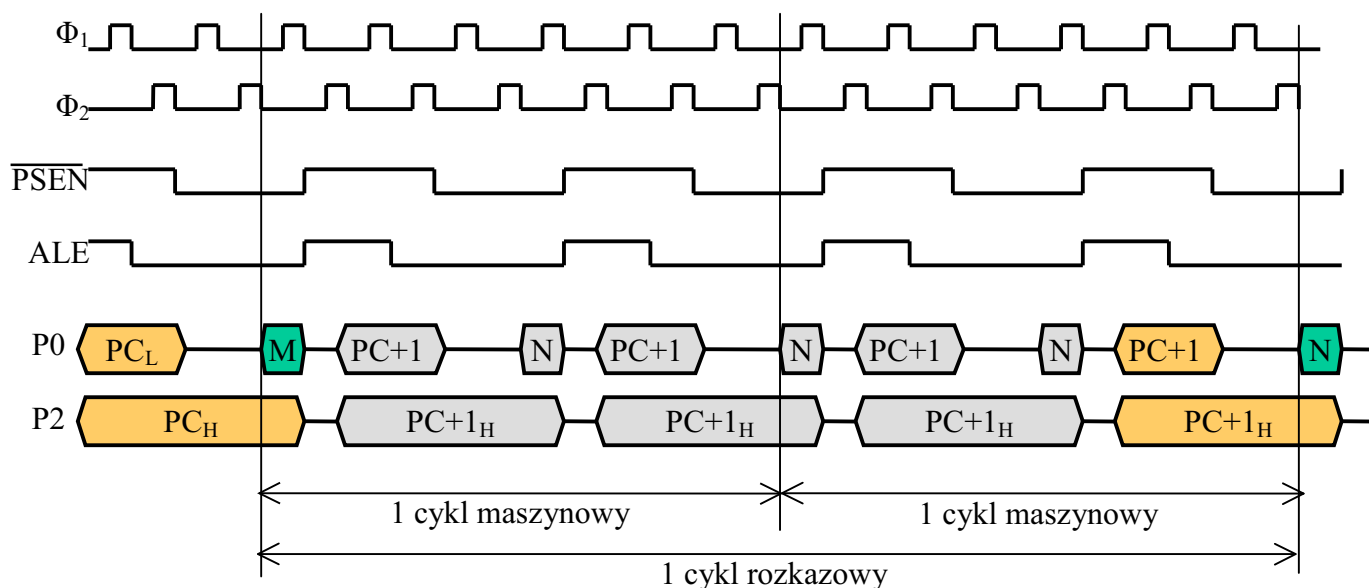


Rys.9a. Wykonanie dwóch rozkazów 1-bajtowych, każdy w 1 cyklu maszynowym (M, N, O - bajty kodów rozkazów: $M=ROM[PC]$, $N=ROM[PC+1]$, $O=ROM[PC+2]$)

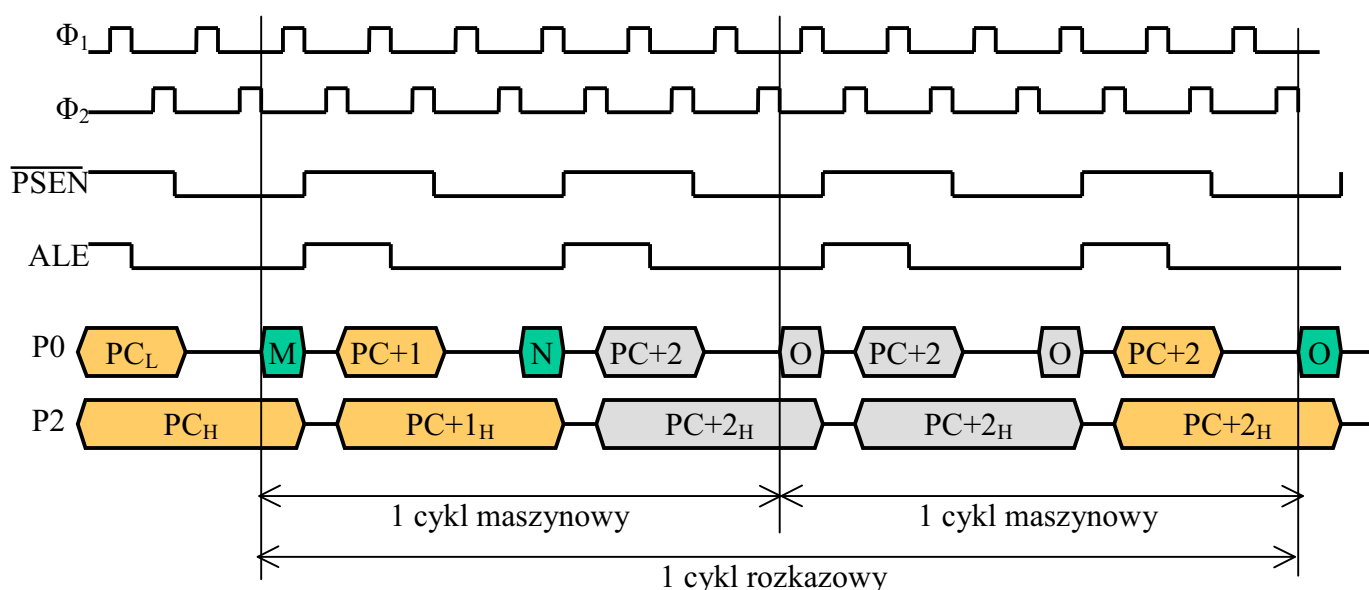


Rys.9b. Wykonanie dwóch rozkazów 2-bajtowych, każdy w 1 cyklu maszynowym (MN, OP - bajty kodów rozkazów: $M=ROM[PC]$, $N=ROM[PC+1]$, $O=ROM[PC+2]$, $P=ROM[PC+3]$)

Rozkazy mikrokomputerów MCS51 mają długość 1, 2 lub 3 bajty i realizowane są w 1, 2 lub 4 cyklach maszynowych. Realizacja rozkazu rozpoczyna się jeszcze w poprzednim cyklu maszynowym przez wysłanie w stanie S5P1 zawartości PC_n na linii $P0$ - $P2$. Natomiast wczytanie kodu rozkazu K_n następuje już w stanie S1P1 nowego cyklu maszynowego. Po tej operacji rejestr PC jest inkrementowany, a rozkaz wykonywany przez mikroprocesor. Ponieważ każdy cykl maszynowy umożliwia dwukrotny dostęp do pamięci programu, w rozkazach jednobajtowych (trzybajtowych) - wczytany podczas S4P1 kod następnego rozkazu K_{n+1} (K_{n+3}) jest ignorowany, a ważny odczyt tego kodu rozpoczyna się dopiero w stanie S5P1 (rysunek 9a, 9c, 9e).

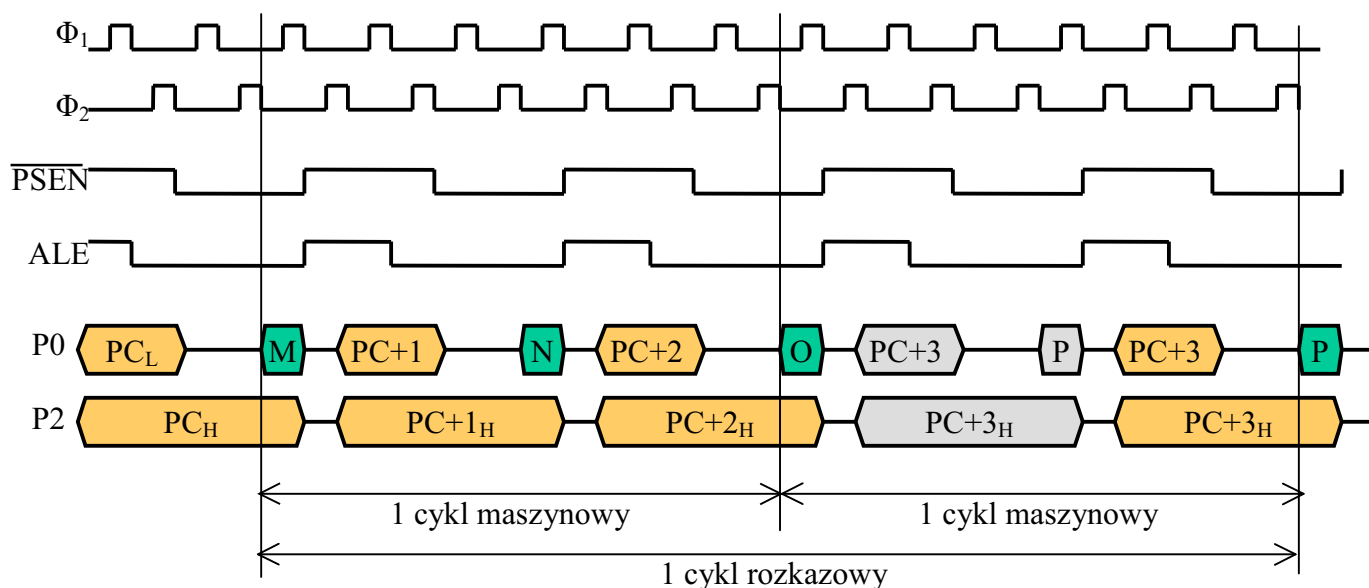


Rys.9c. Wykonanie jednego rozkazu 1-bajtowego w 2 cyklach maszynowych
(M, N - bajty kodów rozkazów: M=ROM[PC], N=ROM[PC+1])



Rys.9d. Wykonanie jednego rozkazu 2-bajtowego w 2 cyklach maszynowych
(MN, O - bajty kodów rozkazów: M=ROM[PC], N=ROM[PC+1], O=ROM[PC+2])

W przypadku rozkazów dwubajtowych kod Kn+1 odczytywany podczas S4P1 nie jest ignorowany, a licznik rozkazów jest inkrementowany ponownie. Wtedy w fazie S5P1 rozpoczyna się cykl pobrania kodu Kn+2 kolejnego rozkazu do wykonania (rysunek 9b). Możliwy jest także wariant rozkazu dwubajtowego, ale realizowanego w dwóch cyklach maszynowych - w takim przypadku oba kody pobierane w drugim cyklu maszynowym są ignorowane (rysunek 9d). Z braku miejsca rysunek 9 nie prezentuje jeszcze jednego wariantu - realizacji jednobajtowego rozkazu mnożenia lub dzielenia, trwającego aż cztery cykle maszynowe. Przebiegi czasowe są w tym przypadku analogiczne jak na rysunku 9c.

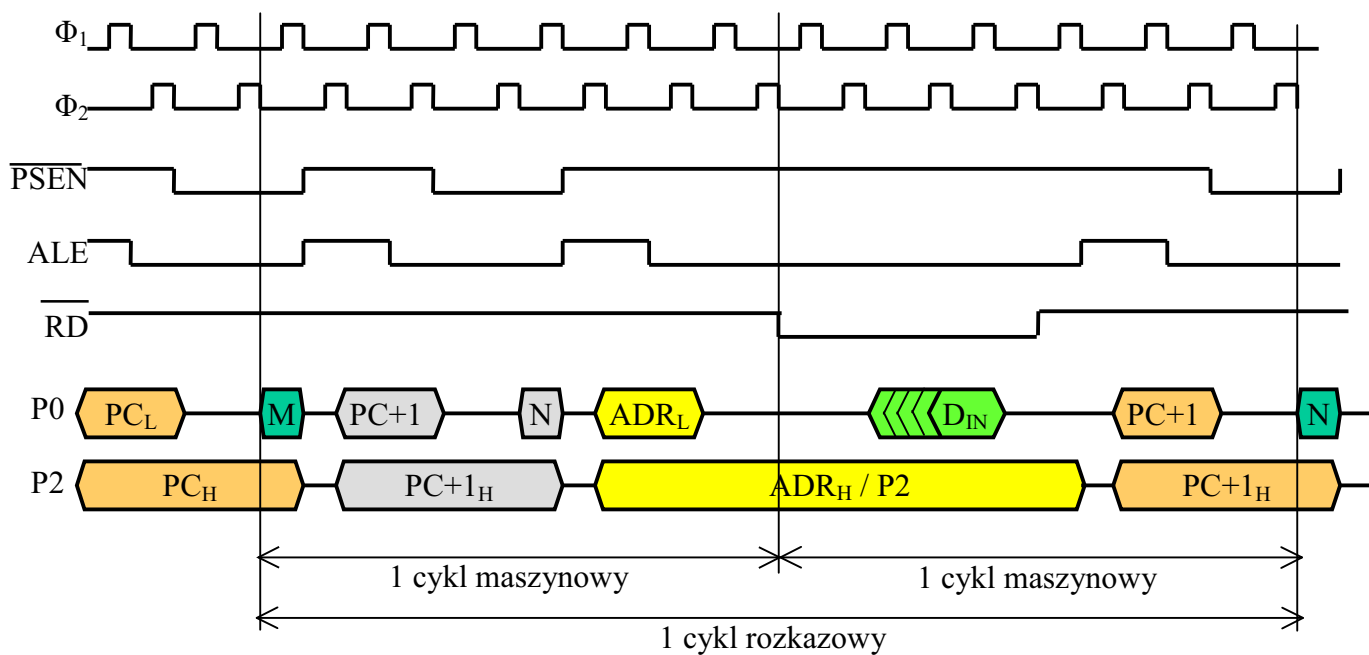


Rys.9e. Wykonanie jednego rozkazu 3-bajtowego w 2 cyklach maszynowych
(MNO, P - bajty kodów rozkazów: M=ROM[PC], N=ROM[PC+1], O=ROM[PC+2],
P=ROM[PC+3])

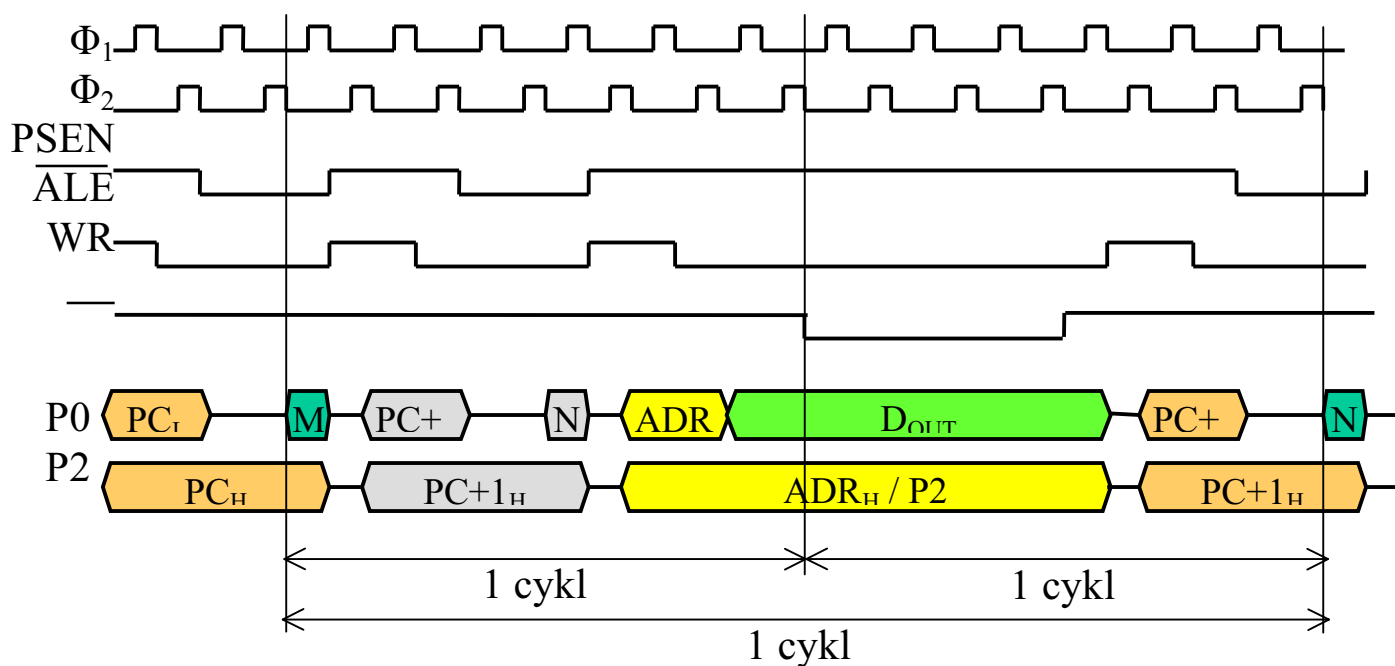
Nieco inaczej wyglądają przebiegi czasowe przy odwoływaniu się do zewnętrznej pamięci danych (urządzeń peryferyjnych). Nie występuje wtedy impuls PSEN podczas S6 pierwszego i S1 drugiego cyklu maszynowego. Zamiast impulsu ALE na początku drugiego cyklu maszynowego uaktywniane są linie RD albo WR (rysunek 10). Takie wydłużenie przebiegów czasowych umożliwia współpracę mikroprocesora z wolniejszymi pamięciami RAM i układami peryferyjnymi. 8051/52 ma także rozkazy dostępu jedynie do 256 najmłodszych bajtów zewnętrznej RAM - nie potrzebne są wtedy adresy A8-A15. Przy realizacji tych rozkazów na liniach P2 pojawia się wtedy stan P2 z SFR.

Również inne czynności mikroprocesora są synchronizowane wewnętrznym przebiegiem taktującym. Między innymi:

- wyjścia portów P0..P3 zmieniają swój stan w fazie S1P1 następującej po cyklu maszynowym, w którym miał miejsce wpis do rejestrów tych portów w SFR;
- przy wprowadzaniu informacji linie P0 są próbkowane w fazie S5P1, a linie P1..P3 – w S5P2, ostatniego cyklu maszynowego rozkazu odczytu;
- w fazie S5P2 każdego cyklu maszynowego badane są linie przerwań i testowane wejścia licznikowe wszystkich timerów zliczających impulsy zewnętrzne;
- stan timerów jest aktualizowany w fazie S3P1 następującej po cyklu maszynowym, w którym stwierdzono opadające zbocze na wejściu licznikowym.



Rys. 10a. Przebiegi czasowe przy odczycie z zewnętrznej pamięci RAM
(M, N - bajty kodów rozkazów: $\text{M}=\text{ROM}[\text{PC}]$, $\text{N}=\text{ROM}[\text{PC}+1]$)



Rys. 10b. Przebiegi czasowe przy zapisie do zewnętrznej pamięci RAM
(M, N - bajty kodów rozkazów: $\text{M}=\text{ROM}[\text{PC}]$, $\text{N}=\text{ROM}[\text{PC}+1]$)

7.Przerwania

7.1.Priorytety przerw

Mikrokomputer 8051 ma pięć źródeł przerw:

przerwanie zewn. INT0;

przerwanie od timera 0 TF0;

przerwanie zewn. INT1;

przerwanie od timera 1 TF1;

przerwanie od portu szeregowego RI+TI;

Układ 8052 posiada jeszcze jedno przerwanie - od timera 2 TF2+EXF2.

Przerwania te są włączane lub wyłączane razem bądź indywidualnie (rejestr IE). Każde z tych przerw może być indywidualnie przypisane do jednej z dwóch możliwych grup priorytetów (wyższej lub niższej). Wewnątrz tych dwóch poziomów istnieją dodatkowe relacje priorytetowe między źródłami przerw: najwyższy priorytet ma przerwanie INT0, a najniższy - przerwanie RI+TI (zgodnie z powyższym wyliczeniem źródeł przerw). Struktura priorytetów umożliwia rozstrzyganie konfliktów przy jednoczesnym pojawieniu się więcej niż jednego przerwania - w takich przypadkach obsługiwane jest przerwanie o wyższym priorytecie. Ponadto obsługa przerwania, przypisanego do niższej grupy priorytetów, może być przerywana przez przerwanie z wyższej grupy. Natomiast obsługa przerwania z wyższej grupy priorytetów nie może już być przerywana przez inne przerwanie. Przypisanie danego przerwania do wybranej grupy priorytetów umożliwia rejestr IP w SFR.

7.2.Proces obsługi przerw

Możliwe źródła przerw są testowane tylko raz pod koniec każdego cyklu maszynowego. Jeżeli w tym czasie zachodzą jednocześnie następujące warunki:

1.nie jest obsługiwane żadne przerwanie z tej samej, bądź wyższej grupy priorytetowej;

2.bieżący cykl maszynowy jest ostatnim cyklem wykonania rozkazu;

3.aktualnie nie jest realizowany rozkaz RETI lub inny operujący na rejestrach IP lub IE;

to po dokończeniu cyklu rozkazowego nastąpi wywołanie odpowiedniej procedury obsługi. Przejście do procedury obsługi przerwania jest równoważne z realizacją instrukcji LCALL a16. Pierwszy z podanych warunków zapewnia realizację, opisanego wcześniej, systemu priorytetów. Drugi z warunków zapewnia dokończenie aktualnie wykonywanej instrukcji, a trzeci powoduje, że przejście do obsługi przerwania nastąpi dopiero po wykonaniu jeszcze jednej instrukcji. Podane warunki określają zarazem czas reakcji na pojawienie się przerwania. Wynosi on, zależnie od sytuacji, od 3 do 8 cykli maszynowych.

Ponieważ operacja testowania potencjalnych źródeł przerw odbywa się w każdym cyklu maszynowym od nowa, istnieje możliwość przeoczenia przez mikroprocesor jakiegoś przerwania, gdy trwało ono bardzo krótko i nie zostało obsłużone z powodu któregoś z wymienionych warunków. Należy o tym pamiętać projektując wykorzystanie systemu przerw.

Wystąpienie przerwania jest odnotowywane sprzętowo poprzez ustawienie odpowiedniego bitu-flagi w rejestrach SFR. Bitami tymi są: IE0, IE1, TF0, TF1 w rejestrze TCON oraz RI i TI w rejestrze SCON. Wszystkie te flagi mogą być równie dobrze ustawiane i zerowane programowo. Pozwala to na programowe generowanie bądź też kasowanie przerw. Uwaga: w przypadku zaprogramowania wejść przerw zewnętrznych jako wejść aktywnych niskim poziomem (bity IT0 i IT1 w TCON równe 0) nie można wygenerować odpowiednich przerw programowo ustawiając bity IE0 i IE1, ale można to zrobić zerując odpowiednie bity P3.2 i P3.3 rejestru P3 w SFR.

Wejściu w procedurę obsługi przerwania może towarzyszyć sprzętowe skasowanie flagi tego przerwania (np. przerwania zewnętrzne). W innych przypadkach skasowanie tej flagi jest

możliwe jedynie na drodze programowej (np. port szeregowy). Poszczególnym przerwaniom są przypisane następujące adresy początkowe procedur ich obsługi:

IE0	0003h
TF0	000Bh
IE1	0013h
TF1	001Bh
RI+TI	0023h
TF2+EXF2	002Bh

Obsługa przerwania winna kończyć się instrukcją RETI. Rozkaz ten informuje mikroprocesor o zakończeniu obsługi przerwania, co jest ważne przy wykorzystywaniu systemu priorytetów. Warto zauważyć, że także rozkaz RET spowoduje powrót do przerwanej programu, ale z punktu widzenia procesora obsługa przerwania nie zostanie jeszcze zakończona.

7.3.Przerwania zewnętrzne

Możliwe są dwa źródła przerwania zewnętrznych: INT0 i INT1. Przy czym istnieje możliwość określenia czy ma być wykrywane opadające zbocze sygnału czy też jego niski poziom (bity IT0 i IT1 w rejestrze TCON). W przypadku uczulenia systemu generacji przerwania na zbocze (IT0, IT1 = 1), aby przerwanie zostało zauważone sygnał wejściowy musi znajdować się najpierw w stanie wysokim, przez czas minimum jednego cyklu maszynowego, a potem w stanie niskim, również przez taki sam czas. Rozpoznanie takiego przerwania objawia się, sprzętowo, ustawieniem bitu IE0 (IE1). Wejście w procedurę obsługi takiego przerwania automatycznie kasuje ten bit. W przypadku wyzwalania przerwania niskim poziomem sygnału (IT0, IT1 = 0), na wejściu przerywającym musi być utrzymywany stan niski przez urządzenie zewnętrzne aż do chwili rozpoczęcia obsługi tego przerwania. Flaga przerwania IE0 (IE1) nie jest automatycznie kasowana przy rozpoczęciu obsługi przerwania, lecz zeruje się wraz z wycofaniem przerwania przez urządzenie żądające obsługi. Cofnięcie żądania obsługi musi nastąpić przed wykonaniem rozkazu RETI, w przeciwnym razie nastąpi ponowne wywołanie procedury obsługującej.

W przypadku 8052 po zaprogramowaniu timera 2 jako generatora zegara transmisji szeregowej dostępne staje się trzecie wejście przerwania zewnętrznych - pin T2EX (P1.1). Sygnał podawany na to wejście jest bramkowany bitem EXEN2 (w rejestrze T2MOD). Układ wykrywa ujemne zbocze sygnału i przy EXEN2=1 ustawia flagę EXF2. Jeżeli przerwanie od timera 2 są odblokowane (ET2=1) to zostanie zgłoszone przerwanie. Należy pamiętać, że rozpoznanie przyczyny przerwania (od TF2 czy od EXF2) oraz skasowanie jego flagi w przypadku timera 2 może być wykonane tylko programowo.

7.4.Przerwania wewnętrzne

Pierwszym rodzajem przerwania wewnętrznych są przerwanie od timerów, sygnalizowane bitami TF0 i TF1 w rejestrze TCON oraz TF2 w rejestrze T2CON (w 8052). Powstają one w wyniku wystąpienia przepełnienia w timerach. Bity TF0 i TF1 ulegają automatycznemu skasowaniu wraz z wejściem do odpowiedniej procedury obsługi, a TF2 musi być kasowana programowo.

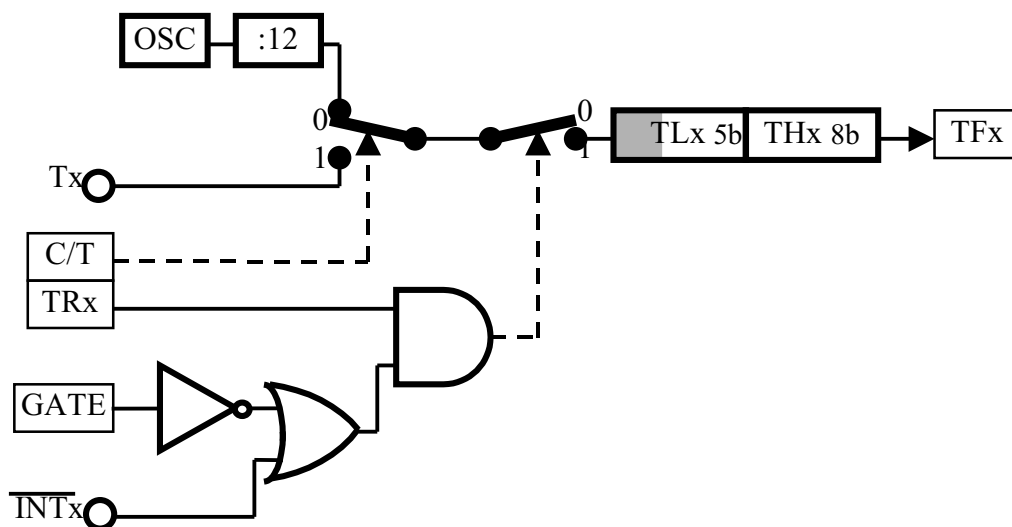
Drugim rodzajem przerwania wewnętrznych są przerwanie od portu transmisji szeregowy, sygnalizowane bitami RI i TI w SCON. Oba te przerwanie są traktowane jako jedno wspólne (suma logiczna RI∨TI). W przeciwieństwie do innych przerwania bity RI i TI nie są zerowane automatycznie - możliwe jest to tylko na drodze programowej.

8. Timery 8051/52

Mikroprocesor 8051 posiada dwa 16-bitowe układy liczące „w górę”. Mogą one być inkrementowane jako timery - zliczają wtedy cykle maszynowe (z częstotliwością 12-krotnie mniejszą od częstotliwości oscylatora), albo jako liczniki liczące impulsy zewnętrzne (opadające zbocza sygnału na odpowiednim pinie wejściowym). Decyduje o tym, dla każdego timera indywidualnie, bit C/T w rejestrze TMOD (C/T=0 - funkcja timera, C/T=1 - funkcja licznika). Ponieważ testowanie wejść T1 i T2 odbywa się tylko raz w cyklu maszynowym, rozpoznanie przejścia sygnału 1→0 zajmuje 2 cykle. W związku z tym maksymalna częstotliwość zliczanych impulsów zewn. jest 24-krotnie mniejsza od częstotliwości oscylatora. Ponadto, aby przejście 1→0 zostało zauważone stany wysoki i niski muszą trwać nie krócej niż jeden cykl maszynowy. Timer 0 i timer 1 mają cztery tryby pracy.

Mod 0

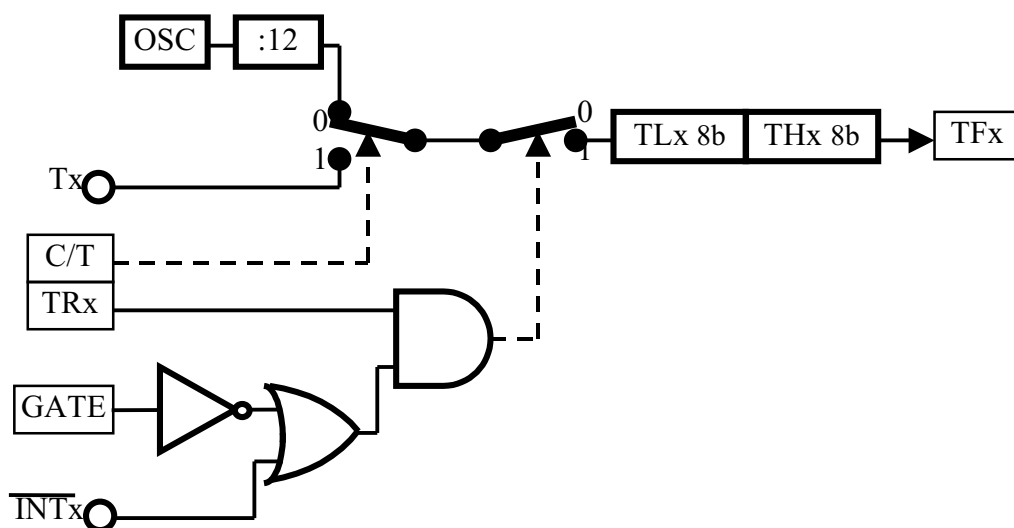
W trybie tym timer 0 (1) działa jako 13-bitowy licznik/timer. Na ten licznik składa się cały rejestr TH0 (TH1) i 5 najmłodszych bitów rejestru TL0(TL1). Trzy najstarsze bity TL0 (TL1) mają stan nieokreślony. Ustawienie flagi TR0 (TR1) nie zeruje timera. Zmiana stanu z 1FFFh na 0000h powoduje ustawienie flagi TF0 (TF1). Układ działa gdy jest ustawiony bit TR0 (TR1) w rejestrze TCON. Bit sterujący GATE w TMOD ma również wpływ na pracę timera. Gdy jest on wyzerowany timer 0 (1) pracuje tylko przy TR0=1 (TR1=1), gdy zaś jest on ustawiony, timer działa przy stanie TR0=1 i pinie INT0=1 (TR1=1 i INT1=1).



Rys.11. Timery 0 i 1 w trybie 13-bitowym

Mod 1

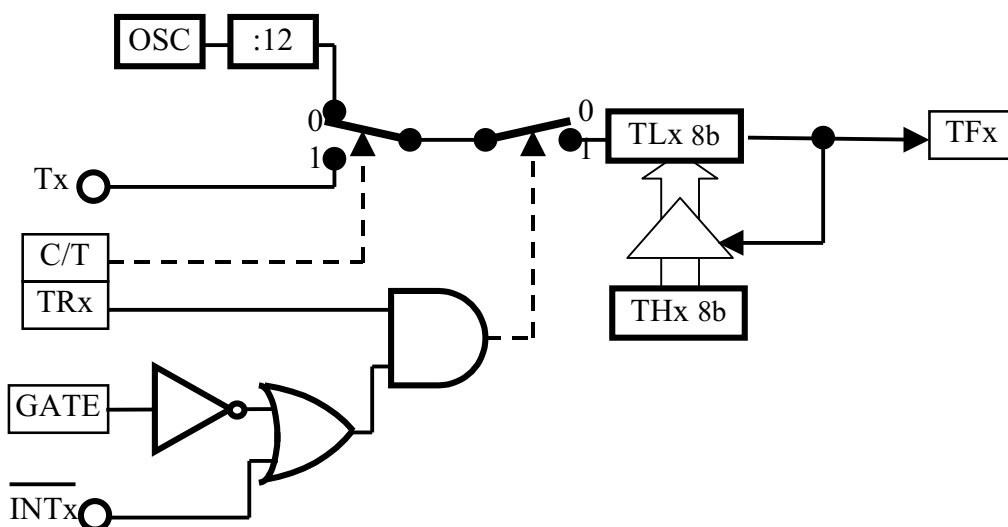
Jest to tryb pracy analogiczny do trybu 0 (rysunek 12). Jedyną różnicą jest to, że aktywny jest cały 16-bitowy licznik TH0TL0 (TH1TL1).



Rys.12. Timery 0 i 1 w trybie 16-bitowym

Mod 2

W modzie tym młodszy bajt timera TL0 (TL1) działa jako 8-bitowy licznik z automatycznym przeładowaniem po wystąpieniu przepełnienia TF0 (TF1). Wartość ładowaną ponownie do TL0 (TL1) przechowuje rejestr TH0 (TH1). Ten tryb pracy umożliwia zaprogramowanie ciągłej pracy timera, z zadaną częstotliwością pojawiania się przerwań. Ma on między innymi zastosowanie przy określaniu częstotliwości pracy portu szeregowego.

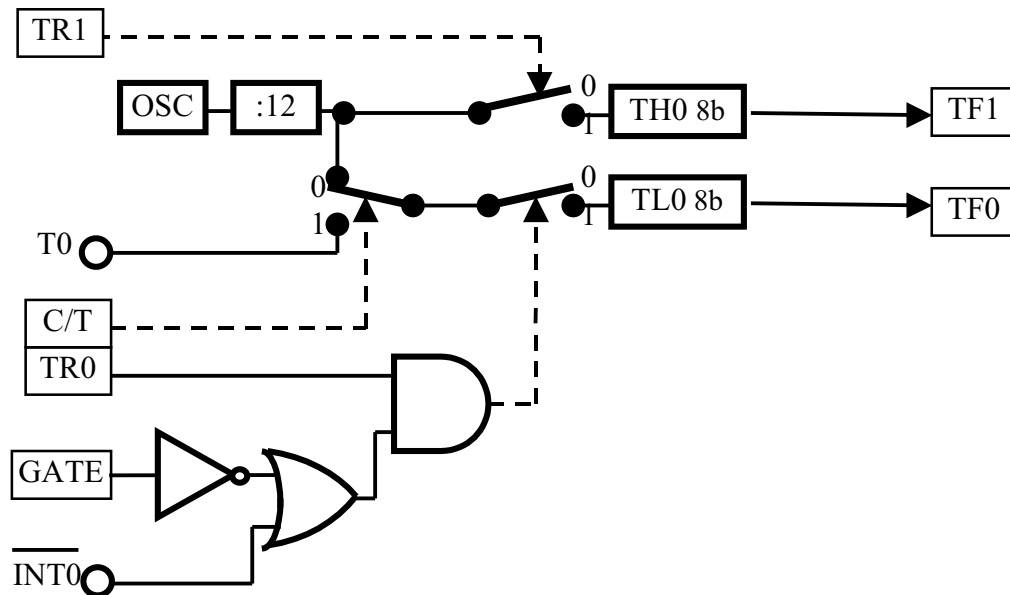


Rys.13. Timery 0 i 1 w trybie 8-bitowym z automatycznym przeładowaniem

Mod 3

W modzie tym pracuje tylko timer 0. Timer 1 w tym modzie jedynie utrzymuje swoją zawartość (tak jak po ustawieniu TR1=0). Rejestry TL0 i TH0 są traktowane jako dwa rozłączne liczniki. TL0 może pracować jako 8-bitowy licznik/timer, jest sterowany bitami GATE, C/T, TR0 i wejściem INT0, tak jak cały timer 0 w modzie 0. Przepełnienie w TL0 ustawia flagę TF0. Natomiast rejestr TH0 jest sterowany bitem TR1 i może pracować tylko jako 8-bitowy timer. Występujące w TH0 przepełnienia powodują ustawienie flagi TF1. Ten mod pracy jest przewidziany dla zastosowań wymagających dodatkowego 8-bitowego licznika lub timera. Ustawienie timera 0 w mod 3 nie przeszkadza wcale ustawieniu timera 1

w mod 0 lub 1 i wykorzystywaniu go np. do generowania częstotliwości pracy portu szeregowego.



Rys.14. Timer 0 w trybie 3 (dwa 8-bitowe liczniki). Timer 1 – wyłączony.

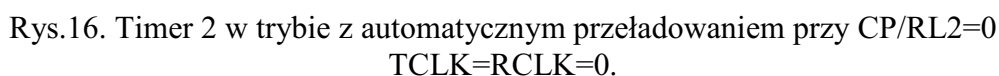
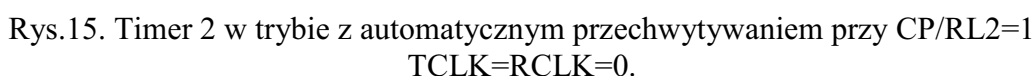
Możliwe zastosowania timerów/liczników 0 i 1:

- odmierzanie zadanych odcinków czasu, koniec pracy sygnalizowany przerwaniem lub tylko flagą TF_x;
- przerwania zegarowe;
- w trybie 8-bitowym z automatycznym przeładowaniem można uzyskać przy $f_{osc}=12\text{MHz}$ okres przerwań zegarowych do $256\mu\text{s}$ ($3,9\text{kHz}$);
- w trybie 16-bitowym z programowym przeładowaniem można uzyskać przy $f_{osc}=12\text{MHz}$ okres przerwań zegarowych do $65,5\text{ms}$ ($15,25\text{Hz}$);
- zliczanie impulsów / zdarzeń zewn.;
- zliczenie zadanej liczby impulsów zewn. i zgłoszenie przerwania;
- taktowanie portu transmisji szeregowej - tylko timer 1 !.

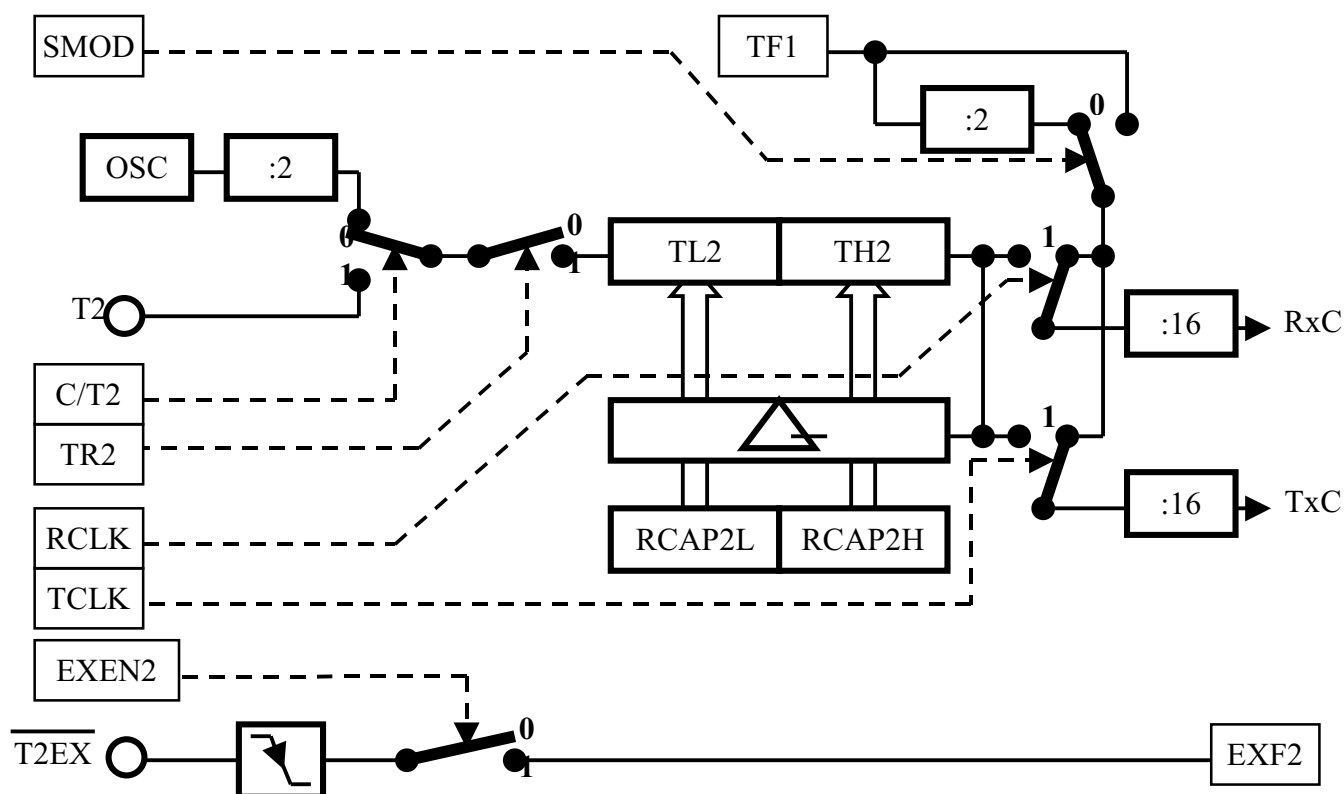
Timer 2 mikrokomputera 8052 może pracować jako 16-bitowy licznik impulsów zewnętrznych podawanych na pin T2 (P1.0) przy C/T2=0 albo timer zliczający okresy przebiegu $f_{osc}/12$ przy C/T2=1.

Ponadto dla obu tych wariantów można wybrać jeden z trzech rodzajów pracy:

- z zatrzymywaniem stanu rejestrów TH2 i TL2 w RCAP2H i RCAP2L, przy CP/RL2=1 i EXEN2=1 przez ujemne zbocze sygnału na T2EX (rysunek 15);
- z automatycznym przeładowaniem TH2 i TL2 zawartością RCAP2H i RCAP2L przy CP/RL2=0 po wystąpieniu przepełnienia, gdy EXEN2=0 albo po wystąpieniu przepełnienia lub ujemnego zbocza na pinie T2EX - gdy EXEN2=1 (rysunek 16);



- 23 -



Rys.17. Timer 2 jako generator szybkości transmisji portu szeregowego.

Możliwe zastosowania timera/licznika 2:

- odmierzanie zadanych odcinków czasu, koniec pracy sygnalizowany przerwaniem lub tylko flagą TF2;
- przerwania zegarowe;
- wykorzystanie 16-bitowej stałej przeładunku pozwala łatwo uzyskać długie okresy przerwań zegarowych;
- zliczanie impulsów / zdarzeń zewn.;
- zliczenie zadanej liczby impulsów zewn. i zgłoszenie przerwania;
- taktowanie portu transmisji szeregowej - niezależnie odbiornik i nadajnik;
- szeroka gama częstotliwości taktowania SIO;
- pomiar czasu trwania zjawisk, których początek i koniec są sygnalizowane opadającym zboczem na T2EX.

9.Port transmisji szeregowej

9.1.Mody pracy portu szeregowego

Wbudowany w strukturę 8051 port transmisji szeregowej może pracować w pełnym duplexie, tzn. jednocześnie odbierać i nadawać informację. Posiada także bufor odbiornika umożliwiający odbiór następnej danej przed odczytaniem z tego bufora poprzedniej. Oczywiście nie odczytanie w porę bufora odbiornika może spowodować w takiej sytuacji utratę którejś danej. Rejestr nadajnika i bufor odbiornika portu szeregowego są dostępne za pośrednictwem bajtu SBUF w bloku SFR. Do sterowania pracą portu służy rejestr SCON. Port szeregowy może pracować w jednym z 4 modów.

Mod 0 portu szeregowego

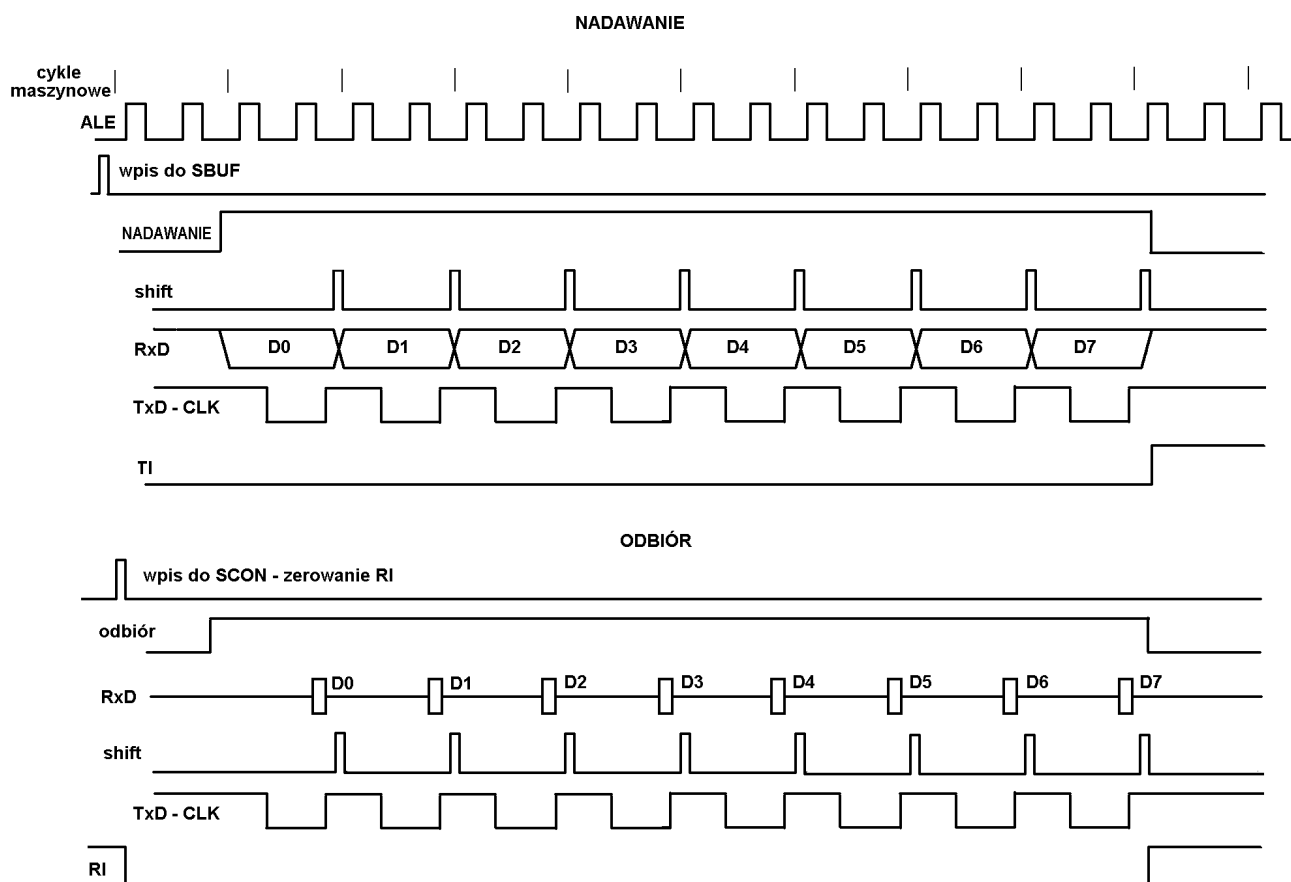
Format transmisji przedstawia rysunku 18. W modzie tym 8-bitowe dane szeregowo wchodzi i wychodzą poprzez pin RxD, a TxD służy jako wyjście sygnału taktującego. Jako

pierwszy jest nadawany/odbierany najmłodszy bit informacji. Częstotliwość sygnału taktującego jest stała i wynosi 1/12 częstotliwości zegara systemu.

Proces nadawania rozpoczyna się od wpisu bajtu do rejestru SBUF. Po upływie 1 cyklu maszynowego rozpoczyna się wysyłanie kolejnych bitów. Nadawanie kończy się ustawieniem TI w 10 cyklu maszynowym od chwili wpisu do SBUF. Proces odbioru informacji jest inicjowany przez ustawienie REN=1 i RI=0 w rejestrze SCON. Faktyczny odbiór rozpoczyna się o jeden cykl maszynowy później. Natomiast kończy się w 10 cyklu przepisaniem do bufora odbiornika skompletowanego bajtu oraz ustawieniem RI=1 (REN nie jest zmieniany).

Mod 0 programuje się wpisując do rejestru SCON:

SM0=0, SM1=0, SM2=0

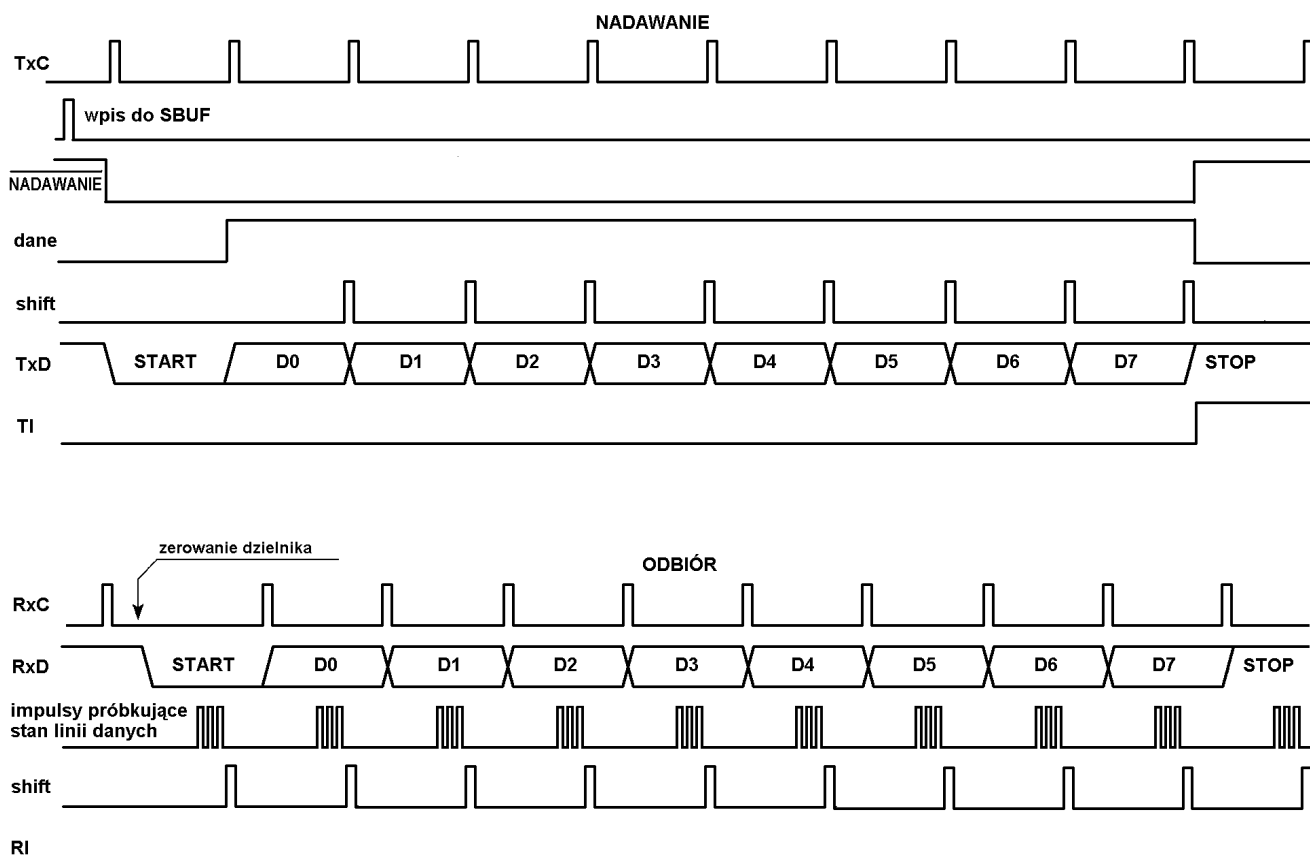


Rys.18. Praca portu szeregowego w trybie synchronicznym

Mod 1 portu szeregowego

Format transmisji przedstawia rysunek 19. Przy odbiorze bit stopu umieszczany jest w RB8 (SCON). Częstotliwość transmisji jest określana przez częstotliwość występowania przepełnień w timerze 1 i stan bitu SMOD w rejestrze PCON.

Nadawanie informacji rozpoczyna się po dokonaniu wpisu do rejestru SBUF. Zgodnie z podanym formatem transmisji najpierw wysyłany jest na wyjście TxD bit startu (0), po nim dane (od najmłodszego bitu), a na końcu bit stopu (1). Zakończenie nadawania jest sygnalizowane przez ustawienie flagi TI.



Rys.19. Format transmisji asynchronicznej ośmiobitowej.

Odbiór informacji rozpoczyna się po wykryciu przejścia 1→0 na wejściu RxD. Specjalny układ logiczny testuje kolejne bity w celu eliminacji fałszywych bitów startu i prawidłowego odczytu bitów danych. Jeżeli w chwili przyjmowania bitu stopu zachodzi:

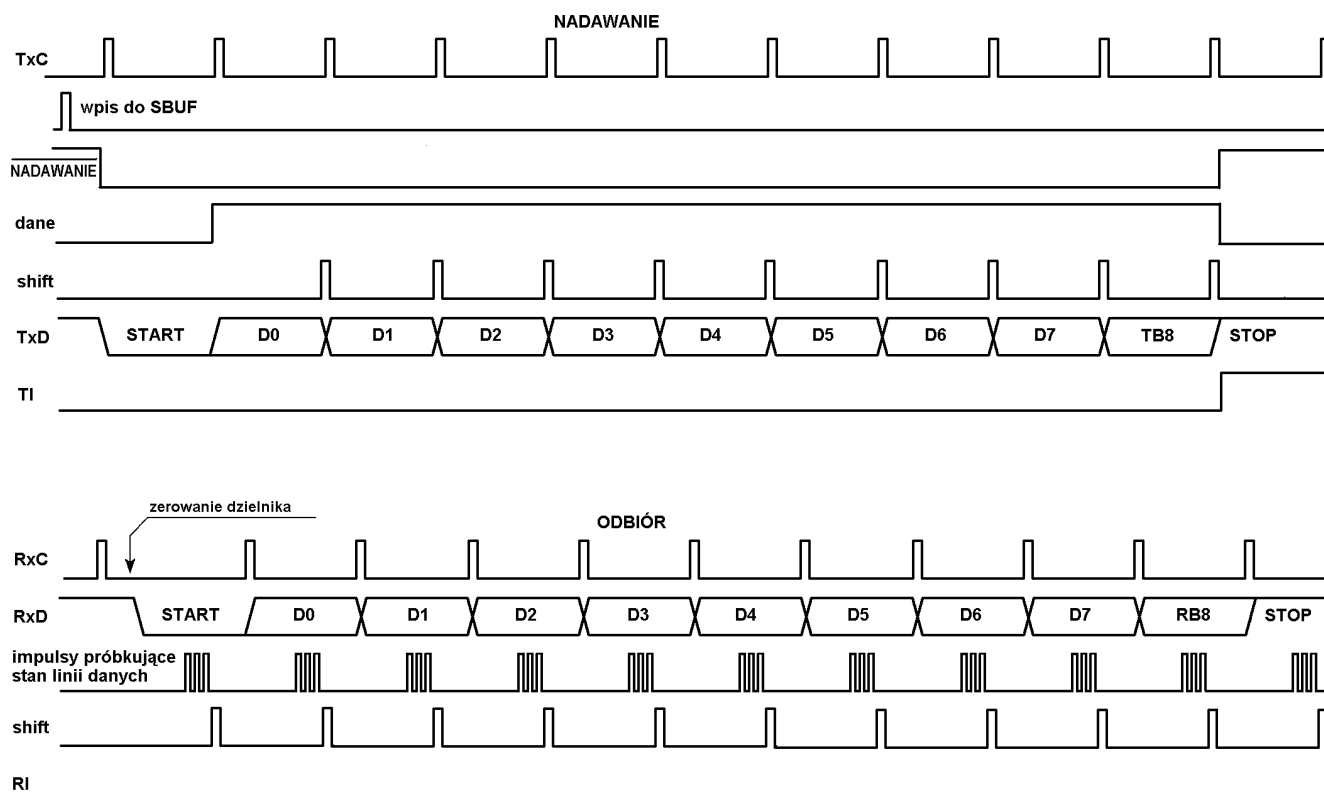
RI=0 i SM2=0 albo RI=0 i SM2=1 i bit stopu=1 to następuje: wpis bitu stopu do RB8, przepisanie odebranej informacji do bufora odbiornika i ustawienie RI=1. Jeżeli natomiast nie jest spełniony któryś z warunków to informacja jest ignorowana. Od tego momentu, układ przechodzi do oczekiwania na następne przejście 1→0 na wejściu RxD.

Mod 1 programuje się wpisując do rejestru SCON:

SM0=0, SM1=1, SM2=0 lub 1

Mod 2 portu szeregowego

W modzie tym przesyłana jest 11-bitowa informacja (rysunek 20). Przy nadawaniu i odbiorze, oprócz bitu startu (0), ośmiu bitów danych i bitu stopu (1), przesyłany jest, jako przedostatni, dodatkowy dziewiąty bit informacyjny będący np. bitem parzystości. Częstotliwość pracy może być równa 1/32 (SMOD=1) lub 1/64 (SMOD=0) częstotliwości zegara systemu.



Rys.20. Format transmisji asynchronicznej dziewięciobitowej.

Nadawanie informacji rozpoczyna się po dokonaniu wpisu do rejestru SBUF. Zgodnie z podanym formatem transmisji najpierw wysyłany jest na wyjście TxD bit startu (0), po nim dane (od najmłodszego bitu), dodatkowy dziewiąty bit, pobierany z TB8 w SCON, a na końcu bit stopu (1). Zakończenie nadawania jest sygnalizowane przez ustawienie flagi TI.

Odbiór informacji rozpoczyna się po wykryciu przejścia 1→0 na wejściu RxR. Specjalny układ logiczny testuje kolejne bity w celu eliminacji fałszywych bitów startu i prawidłowego odczytu bitów danych. Jeżeli w chwili przyjmowania bitu stopu zachodzi:

RI=0 i SM2=0 albo RI=0 i SM2=1 i dziewiąty bit danych=1 to następuje: wpis dziewiątego bitu do RB8, przepisanie odebranej informacji do bufora odbiornika i ustawienie RI=1. Jeżeli zaś nie jest spełniony któryś z warunków to informacja jest ignorowana. Od tego momentu, układ przechodzi do oczekiwania na następne przejście 1→0 na wejściu RxR.

Mod 2 programuje się wpisując do rejestru SCON:

SM0=1, SM1=0, SM2=0 lub 1

Mod 3 portu szeregowego

Ten mod pracy różni się od modu 2 jedynie możliwością doboru częstotliwości pracy portu przy pomocy timera 1. Mod 3 programuje się wpisując do rejestru SCON:

SM0=1, SM1=1, SM2=0 lub 1

9.2.Szybkość transmisji portu szeregowego

Przyjmując oznaczenia: f_{OSC} - częstotliwość oscylatora (kwarcu); f_{T1} - częstotliwość występowania przepełnienia w timerze 1; f_{tr} - częstotliwość transmisji; można podać wzory określające szybkość transmisji w poszczególnych modach pracy:

mod 0: $f_{tr}=f_{OSC}/12$;

mod 1: $f_{tr}=f_{T1}*2/32$;

mod 2: $f_{tr}=f_{OSC}*2/64$;

mod 3: $f_{tr}=f_{T1}*2/32$.

Przy wykorzystywaniu timera 1 do generacji częstotliwości f_{tr} (mod 1 i 3), timer ten powinien mieć zablokowane przerwania. Timer 1 może wtedy pracować jako licznik albo timer w jednym z trzech swoich trybów pracy. Najczęściej wykorzystuje się mod 2 i funkcję timera (górna połowa rejestru TMOD: 0010b). W takim przypadku występującą w podanych wzorach f_{T1} można określić:

$$f_{T1}=f_{OSC}/12/(256-TH1)$$

Można w takiej konfiguracji uzyskać szybkość transmisji z zakresu: 122bodów..62.5kbodów, przy $f_{OSC}=12\text{MHz}$ (tabela poniżej).

PORT SZEREGOWY				TIMER/LICZNIK 1		
mod	szybkość	f_{OSC} [MHz]	bit SMOD	bit C/T	tryb	stała przeładowania
0	1MHz	12,000	x	x	x	x
2	375000bod	12,000	1	x	x	x
2	187500bod	12,000	0	x	x	x
1,3	62500bod	12,000	1	0	2	0FFh
1,3	19200bod	11,059	1	0	2	0FDh
1,3	9600bod	11,059	0	0	2	0FDh
1,3	4800bod	11,059	0	0	2	0FAh
1,3	2400bod	11,059	0	0	2	0F4h
1,3	1200bod	11,059	0	0	2	0E8h
1,3	600bod	11,059	0	0	2	0D0h
1,3	300bod	11,059	0	0	2	0A0h
1,3	137,5bod	11,059	0	0	2	1Dh
1,3	110bod	6,000	0	0	2	72h
1,3	110bod	12,000	0	0	1	0FEEBh

Chcąc osiągnąć bardzo niskie częstotliwości pracy, np. 110 bodów, należy użyć oscylatora o mniejszej f_{OSC} albo zaprogramować timer 1 w mod 0 lub 1 pozostawiając odblokowane przerwania TF1. Ponieważ te mody pracy timera nie mają automatycznego przeładowania, operację tę wykonywałaby procedura obsługi przerwania TF1. Przykładowo: $f_{tr}=110\text{bodów}$ można uzyskać przy $f_{osc}=12\text{MHz}$ zapisując do TH1TL1 liczbę 0FEEBh.

W 8052 do generacji f_{tr} można także wykorzystać timer 2 ustawiając odpowiednio bity w rejestrze T2CON. Ponieważ w tym trybie pracy timer 2 zlicza na 16 bitach okresy przebiegu o częstotliwości $f_{osc}/2$, możliwą do osiągnięcia szybkość transmisji w modach 1 i 3 określa wzór:

$$f_{tr}=f_{OSC}/(65536-RCAP2H,RCAP2L)/32$$

Daje to jeszcze szerszy przedział możliwych szybkości transmisji. Należy zaznaczyć, że przy wykorzystywaniu timera 2 jako zegara portu szeregowego nie powinno się wykonywać operacji odczytu lub zapisu na żadnym z rejestrów: TH2, TL2, RCAP2H i RCAP2L. Operacje te z uwagi na zwiększoną szybkość pracy timera mogą prowadzić do błędów w działaniu portu szeregowego. Można je wykonać bezpiecznie tylko przy zatrzymanym timerze (TR2=0).

9.3.Komunikacja wieloprocessorowa

Mody 2 i 3 portu szeregowego umożliwiają zorganizowanie w prosty sposób transmisji wieloprocessorowej. Wykorzystuje się do tego dziewiąty bit danych i bit kontrolny SM2 w rejestrze SCON. Rozważmy sytuację, w której procesor nadrzędny wysyła informację do kilku procesorów podrzędnych. Blok informacji winien być wtedy poprzedzony adresem odbiorcy z ustawionym bitem dziewiątym, natomiast słowa danych powinny mieć ten bit wyzerowany. Procesory podrzędne, normalnie mając ustawiony bit SM2=1, śledzą nadawaną informację i oczekują na słowo z ustawionym bitem dziewiątym - słowa danych są ignorowane. Pojawienie się słowa adresowego powoduje zgłoszenie przerwania we wszystkich procesorach. Rozpoznają one odebrany adres i następnie procesor zaadresowany ustawia SM2=0, co pozwala mu przyjmować następującą po adresie informację. Procesory niezaadresowane pozostawiają SM2=1, czekając na następne słowo adresowe.

Przykładowy algorytm działania:

1. Procesor nadrzędny wysyła ramkę z adresem indywidualnym lub ogólnym i ustawionym bitem TB8 - adres ten określa do kogo jest adresowana następująca po nim transmisja.
2. Wszystkie pozostałe procesory, działając jako podrzędne, mają ustawiony bit SM2 i odbiorą wysłaną ramkę wywołania - adres wywołania trafi do ich rejestrów SBUF i ustawią się flagi RI.
3. Uruchomione zostaną procedury obsługi SIO, które w przypadku przerwania od odbiornika (RI=1) przy ustawionym bicie SM2=1 porównują bajt z SBUF z przyporządkowanymi adresami indywidualnymi lub ogólnym.
4. Procesor, który stwierdzi zgodność adresu wywołania z własnym, zeruje swój bit SM2 i kończy procedurę obsługi SIO, pozostałe procesory kończą obsługę SIO pozostawiając SM2=1.
5. Następuje wymiana informacji pomiędzy procesorem nadrzędnym a zaadresowanym podrzędnym przy SM2=0 i zerowych bitach TB8/RB8. SIO pozostałych procesorów ignorują tak przesyłane ramki danych.
6. Po zakończeniu transmisji, procesor podrzędny ponownie ustawia SM2=1.

10.Porty mikroprocesora 8051

Wszystkie cztery porty (P0, P1, P2 i P3) są dwukierunkowe, składają się z rejestru (w SFR), bufora wyjściowego i bufora wejściowego. Wyjściowe bufony P0 i P2 oraz bufor wejściowy P0 są wykorzystywane przy dostępie do pamięci zewnętrznej. P0 wysyła młodszy bajt adresu w pamięci, multipleksowany (w czasie) z bajtem czytanej lub pisanej informacji. P2, natomiast, jest używany tylko przy stosowaniu dłuższej niż 8-bitowa adresacji pamięci, w pozostałych przypadkach port ten wysyła zawartość rejestru z SFR. Piny portu P3 są wielofunkcyjne: mogą służyć jako zwykłe wyjścia portu bądź jako linie sterujące. Alternatywne funkcje tych pinów są następujące:

P3.0 - RxD - wejście portu szeregowego;

P3.1 - TxD - wyjście portu szeregowego;

XRL Px,A - różnica symetryczna Px z A;
 JBC Px.b,.. - skok warunkowy ze zmianą bitu warunku b w Px;
 CPL Px.b - negacja bitu b w Px;
 INC Px - inkrement Px;
 DEC Px - dekrement Px;
 DJNZ Px,.. - dekrement Px i skok warunkowy;
 MOV Px.b,C - zapis bitu CY do bitu b w Px;
 CLR Px.b - zerowanie bitu b w Px;
 SET Px.b - ustawienie bitu b w Px.

Ostatnie trzy instrukcje w celu zmiany stanu bitu Px.b wczytują najpierw cały rejestr Px, zmieniają bit b i odsyłają wynik do SFR.

Jak to już podano, port P2 wystawia starszą część adresu przy dostępie do pamięci zewnętrznej. Jednakże odpowiadający mu rejestr w SFR nie musi zawierać jedynek (0FFh). Co więcej, w cyklach maszynowych nie odwołujących się do pamięci zewn., na jego piny jest wysyłana niezmienną zawartość rejestru P2. Natomiast rejestr portu P0 w SFR, przy dostępie do pamięci zewnętrznej przyjmuje wartość 0FFh.

11.Przykłady programowania w asemblerze 8051

W rozdziale tym zostanie przedstawionych kilka prostych przykładów programowania w asemblerze 8051/52.

; Procedura przekształcająca liczbę binarną z akumulatora ; na dwa znaki ASCII będące cyframi szesnastkowymi ; parametry: A = liczba;

; R1 -> bufor na znaki;
 ; wyniki: M(R1),M(R1+1) = 2 znaki ASCII hex;
 ; zmiany: A, R1:=R1+2;
 ; długość: 32B;
 ; czas trwania: 28-30c.m.

```

TOASCII: PUSH    B                ;przechowanie rej. B na stosie
          MOV     B, #16
          DIV     AB              ;A:=A/16, B:=A mod 16
          ORL     A, #30H         ;tworzenie znaku ASCII
          CJNE    A, #3AH, TA1    ;dla starszej cyfry hex.
TA1:      JC      TA2            ;skok przy znakach '0'..'9'
          ADD     A, #7           ;korekcja kodu znaku
TA2:      MOV     @R1, A          ;znak do bufora
          INC     R1              ;następna pozycja w buforze
          MOV     A, B
          ORL     A, #30H         ;tworzenie znaku ASCII
          CJNE    A, #3AH, TA3    ;dla m•odszej cyfry hex.
TA3:      JC      TA4            ;skok przy znakach '0'..'9'
          ADD     A, #7           ;korekcja kodu znaku
TA4:      MOV     @R1, A          ;znak do bufora
          INC     R1              ;następna pozycja w buforze
          POP     B              ;odtworzenie rej. B
          RET
  
```

; Fragment programu realizujący skok z przełącznikiem.

; Numer skoku do wykonania znajduje się w akumulatorze.

```

    RL      A                    ;zdwojenie warto•ci
  
```

```

MOV    DPTR, #JMPTAB
JMP    @A+DPTR

JMPTAB: . . . . .
AJMP   PROC1
AJMP   PROC2
. . . . .
AJMP   PROCN

```

; Procedura realizująca programowo zadane opóźnienie.

; parametry: R7 = stała opóźnienia;

; wyniki: czas wykonania procedury zależny od R7;

; zmiany: PSW, R7:=0;

; długość: 12B;

; czas trwania: 10+6*R7c.m. (liczba 10 wynika z PUSH, POP i RET).

```

DELAY:  PUSH    ACC            ;przechowanie A i B na stosie
        PUSH    B
DELAY1:  MUL     AB            ;w•a•ciwa p•tla opó•niaj•ca
        DJNZ    R7, DELAY1
        POP     B            ;odtworzenie stanu B i A
        POP     ACC
        RET

```

; Procedura zamieniająca liczbę binarną z akumulatora ; na pojedyncze cyfry BCD we wskazanym buforze. ; parametry: A = liczba;

; R0 -> bufor na znaki;

; wyniki: M(R0),M(R0+1),M(R0+2) = 3 cyfry BCD;

; zmiany: A, R0:=R0+2;

; długość: 21B;

; czas trwania: 25c.m.

```

TOBCD:  PUSH    B            ;przechowanie rej. B na stosie
        MOV     B, #100
        DIV     AB            ;A=cyfra setek
        MOV     @R0, A        ;cyfra do bufora
        INC     R0
        MOV     A, B
        MOV     B, #10
        DIV     AB            ;A=cyfra dziesi•tek, B=cyfra jedno•ci
        MOV     @R0, A        ;umieszczenie cyfr w buforze
        INC     R0
        MOV     @R0, B
        POP     B            ;odtworzenie rej. B
        RET

```

; Procedura dodająca 2-bajtowe liczby w kodzie BCD ; Młodsze bajty liczb umieszczane są na niższych adresach. ; parametry: R0 -> liczba1;

; R1 -> liczba2;

; wyniki: M(R0),M(R0+1) = liczba1+liczba2;

; zmiany: PSW, R1:=R1+1, R0:=R0+1;

; dlugosc: 16B;

; czas trwania: 17c.m.

```

DODBCD: PUSH    ACC            ;przechowanie rej. A na stosie
        CLR     C
        MOV     A, @R0        ;A:=młodszy bajt liczby1
        ADD     A, @R1        ;dodanie młodsze bajtu liczby2
        DA      A            ;korekcja dziesiętna
        MOV     @R0, A        ;młodszy bajt wyniku

```



```

INC    R0          ;nastepne bajty liczb
INC    R1
MOV    A,@R0       ;A:=starszy bajt liczby1
ADDC   A,@R1       ;dodanie starszego bajtu liczby2
DA     A           ;korekcja dziesietna
MOV    @R0,A       ;starszy bajt wyniku
POP    ACC         ;odtworzenie rej. A
RET                      ;CY=1 oznacza przepelnienie

```

; Fragment programu realizujący „odczyt w locie” stanu jednego z ; liczników. Przy takim odczycie konieczne jest zabezpieczenie ; się przed zmianą stanu licznika pomiędzy odczytem jego młodszego ; i starszego bajtu. Służy temu rozkaz CJNE.

```

RDT0:   . . . . .
        MOV    A,TH0      ;A:=starszy bajt licznika
        MOV    R0,TL0     ;R0:=młodszy bajt licznika
        CJNE   A,TH0,RDT0 ;skok gdy zmiane uległ TH0
        MOV    R1,A       ;odczyt poprawny

```

; Procedura konwersji cyfry szesnastkowej na kod wyświetlacza ; 7-segmentowego.
 Poszczególne segmenty są przyporządkowane ; bitom od najstarszego następująco: hgfdcba.
 ; parametry: A = cyfra hex.;
 ; wyniki: A = kod segmentowy;
 ; zmiany: - ;
 ; długość: 19B (łącznie z tablicą kodów);
 ; czas trwania: 5c.m.

```

HEXDISP:INC    A          ;dodanie przesunięcia tablicy kodów
                                ;względem rozkazu RET
        MOVC   A,@A+PC    ;pobranie kodu wyświetlacza
        RET
KODY:   DEFB   00111111b,00000110b,01011011b,01001111b
        DEFB   01100110b,01101101b,01111101b,00000111b
        DEFB   01111111b,01101111b,00000001b,01000000b
        DEFB   00001000b,01010000b,01111001b,00000000b

```

12. Tryby pracy z obniżonym poborem prądu

Mikroprocesory wykonane w technologii HMOS mają tylko jeden tryb pracy - uśpienie. W trybie tym podtrzymywana jest jedynie zawartość wewnętrznej RAM za pomocą napięcia podanego na wejście RST procesora. Przejście do tego trybu pracy polega na tym, że: zewnętrzny obwód wykrywa zanik zasilania i wystawia sygnał przerwania, w trakcie obsługi tego przerwania do wewnętrznej RAM zostają zapisane istotne dane i następuje podłączenie napięcia podtrzymania do pinu RST (czynności te muszą być wykonane przed zniknięciem zasilania procesora). Po ponownym włączeniu zasilania napięcie podtrzymujące winno być utrzymane na RST przez czas potrzebny na wyzerowanie mikroprocesora. Po odłączeniu tego napięcia procesor wraca do normalnej pracy.

Mikroprocesory wykonane w technologii CHMOS mają dwa tryby pracy z obniżonym poborem prądu. W obu przypadkach napięcie podtrzymujące winno być podane na ten sam pin co zasilanie (Vcc).

Tryb czuwania.

Przy przejściu do tego trybu ostatnim normalnie wykonanym rozkazem jest rozkaz ustawiający bit IDL (PCON.0). W trybie tym działa nadal system przerwań, liczniki i port szeregowy. A, PSW, PC, SP i inne rejestry zachowują swój stan. Wyjścia portów pozostają w swym stanie sprzed ustawienia bitu IDL. Linie ALE i PSEN są nieaktywne.

Wyjście ze stanu czuwania może być zrealizowane dwojako: poprzez przerwanie lub reset (w obu wypadkach bit IDL jest zerowany sprzętowo). W przypadku przerwania możliwy jest ponowny powrót do trybu czuwania: na końcu procedury obsługi należy rozpoznać w jakim stanie znajdował się procesor przed przerwaniem - mogą być do tego użyte flagi GF1 i GF0 w PCON - i podjąć odpowiednie działania (należy pamiętać aby nie narastał stos programu). Ponieważ reset zeruje cały rejestr PCON nie można w prosty sposób wrócić do trybu czuwania.

Tryb uśpienia.

Przejście do tego trybu następuje po ustawieniu bitu PD (PCON.1). W stanie tym wszystkie funkcje wewnętrzne procesora są zawieszone. Utrzymywana jest jedynie zawartość wewnętrznej RAM i SFR. Wyjścia portów utrzymują stan zgodny z zawartością ich rejestrów w bloku SFR, a linie ALE i PSEN pozostają w stanie niskim. Napięcie podtrzymujące może być podane na pin Vcc dopiero po przejściu w tryb uśpienia.

Wyjście z trybu uśpienia jest możliwe jedynie poprzez reset (reset sprzętowy powoduje między innymi wyzerowanie całego PCON). Przed zakończeniem trybu uśpienia na pin Vcc musi być podane normalne napięcie zasilające.

Z przeprowadzonych badań wynika, że wartość prądu podtrzymania pobieranego przez mikroprocesor w trybie uśpienia zależy od stanu systemu przerwań w momencie ustawiania bitu PD. Chcąc maksymalnie zredukować ten prąd należy: zatrzymać liczniki, wyłączyć port szeregowy i zablokować przerwanie, zanim ustawi się bit PD.

Stan portów i linii sterujących, w obu trybach pracy z obniżonym poborem prądu, jest zależny także od tego, gdzie jest umieszczony rozkaz wprowadzający mikrokomputer w stan uśpienia lub czuwania. Możliwe są dwa warianty: lokalizacja w wewnętrznej bądź zewnętrznej pamięci programu. Szczegóły podaje niniejsza tabela.

tryb	rozkaz pobrany z pamięci	ALE	PSEN	P0	P1	P2	P3
czuwanie	wewnętrznej	1	1	P0	P1	P2	P3
	zewnętrznej	1	1	xx	P1	adr	P3
uśpienie	wewnętrznej	0	0	P0	P1	P2	P3
	zewnętrznej	0	0	xx	P1	P2	P3

13.Praca krokowa

Struktura przerwania 8051 pozwala zorganizować pracę krokową w bardzo prosty sposób. Jak to wcześniej przedstawiono, przerwanie nie może zostać obsłużone dopóki nie zakończono obsługi przerwania o tym samym priorytecie, bądź też dopóki nie zostanie wykonana jeszcze jedna instrukcja po rozkazie RETI. Tak więc procedura obsługi przerwania nie może być ponownie uruchomiona, jeżeli w międzyczasie nie zostanie wykonana przynajmniej jedna instrukcja przerywanego programu. Jednym ze sposobów wykorzystania powyższej właściwości do organizacji pracy krokowej jest zaprogramowanie jednego z przerw zewnętrznych jako wrażliwego na poziom.

Procedura obsługująca to przerwanie winna się kończyć następująco:

JNB P3.2, \$;oczekiwanie na przejście INT0 w stan wysoki

JB P3.2, \$;oczekiwanie na powrót INT0 do stanu niskiego

RETI ;powrót w celu wykonania jednej instrukcji

Działanie przedstawionego rozwiązania jest następujące: wymuszenie niskiego poziomu na wejściu INT0 powoduje wywołanie procedury obsługi. Podana sekwencja instrukcji na końcu tej procedury, zezwala na jej zakończenie dopiero po pojawieniu się dodatniego impulsu na wejściu INT0. W chwili wykonywania rozkazu RETI wejście INT0 jest znowu w stanie niskim. Wykonana więc zostanie tylko jedna instrukcja z programu tła, a po niej nastąpi ponowne uruchomienie omawianej procedury.

14.Reset mikroprocesora

Do sprzętowego resetu (zerowania) 8051 służy wejście RST, będące wejściem przerzutnika Schmitt'a. Aby reset był skuteczny, należy utrzymać wysoki poziom napięcia na tym wejściu przez minimum 24 okresy pracy oscylatora (co odpowiada 2 cyklom maszynowym). Wewnętrzny reset procesora następuje w drugim z cykli i jest powtarzany w każdym następnym, aż do spadku napięcia na wejściu RST. Zawartość wewnętrznej RAM nie jest zmieniana w trakcie operacji zerowania, natomiast rejestry SFR przyjmują wartości:

PC = 0000h	A = 00h	TMOD = 00h
DPTR = 0000h	B = 00h	TCON = 00h
PSW = 00h	P0 = 0FFh	TH0 = 00h
SP = 07h	P1 = 0FFh	TL0 = 00h
PCON = 00h	P2 = 0FFh	TH1 = 00h
SCON = 00h	P3 = 0FFh	TL1 = 00h
SBUF = ??	IP = 00h	IE = 00h

15.Mikrokomputery z wbudowaną pamięcią EPROM

W mikrokomputerach jednomodułowych 8751 i 8752 miejsce zwykłej pamięci ROM programu zajmuje pamięć EPROM. Umożliwia to programowanie elektrycznie i kasowanie ultrafioletem przez użytkownika kolejnych wersji pamięci programu. Rozwiązanie z wbudowanym EPROMem, eliminując co najmniej dwa dodatkowe układy scalone, pozwala na zmniejszenie wymiarów konstrukcji i zwiększenie jej niezawodności.

Układ 8751 (8752), w trakcie programowania, musi pracować z oscylatorem o częstotliwości 4-6MHz, ponieważ procedura programowania wymaga działania wewnętrznej magistrali mikrokomputera.

Programowanie pojedynczego bajtu pamięci EPROM polega na:

- utrzymaniu wyprowadzeń P2.4, P2.5, P2.6 i PSEN na potencjale masy, a P2.7, RST, EA/VPP i ALE/PROG - na poziomie 2,5..5V;
- podaniu na port P1 młodszego bajtu adresu, a na piny P2.0÷P2.3 czterech najstarszych bitów adresu;
- podaniu na port P0 programowanego bajtu;

- podniesieniu napięcia V_{pp} do 21V i zwarcia ALE/PROG do masy na czas 50 milisekund.

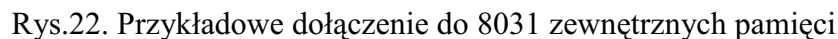
Wersje mikrokomputerów z EPROMem posiadają mechanizm ochrony zawartości pamięci programu przed odczytem przez niepowołane osoby - jest to tzw. bit ochronny. Zaprogramowanie go uniemożliwia jakikolwiek dostęp do pamięci programu, nawet w celu weryfikacji jej zawartości. Niemożliwe jest także doprogramowanie pamięci EPROM, czy też dołączenie zewnętrznej pamięci programu. Programowanie tego bitu jest analogiczne do programowania dowolnego bajtu pamięci: podaje się dowolny adres i bajt programu, jedynie potencjał pinu P2.6 musi być tym razem wysoki. Skasowanie bitu ochronnego następuje jedynie przez skasowanie całej pamięci programu.

Po zapisaniu pamięci EPROM programem, o ile bit ochronny nie został ustawiony możliwa jest weryfikacja wpisanej treści, tzn. porównanie jej ze wzorcem. Procedura odczytu pamięci programu jest podobna do procedury jej zapisu. Wyprowadzenia ALE/PROG i EA/VPP winny być cały czas na poziomie logicznym wysokim, a jedynie podanie zera na bit P2.7 powoduje wyprowadzenie na port P0 zawartości bajtu o zadanym adresie. Do tej operacji linie portu P0 powinny być podłączone do napięcia zasilania przez oporniki o wartości około 10k Ω .

16. Przykłady łączenia mikroprocesora 8051 z otoczeniem

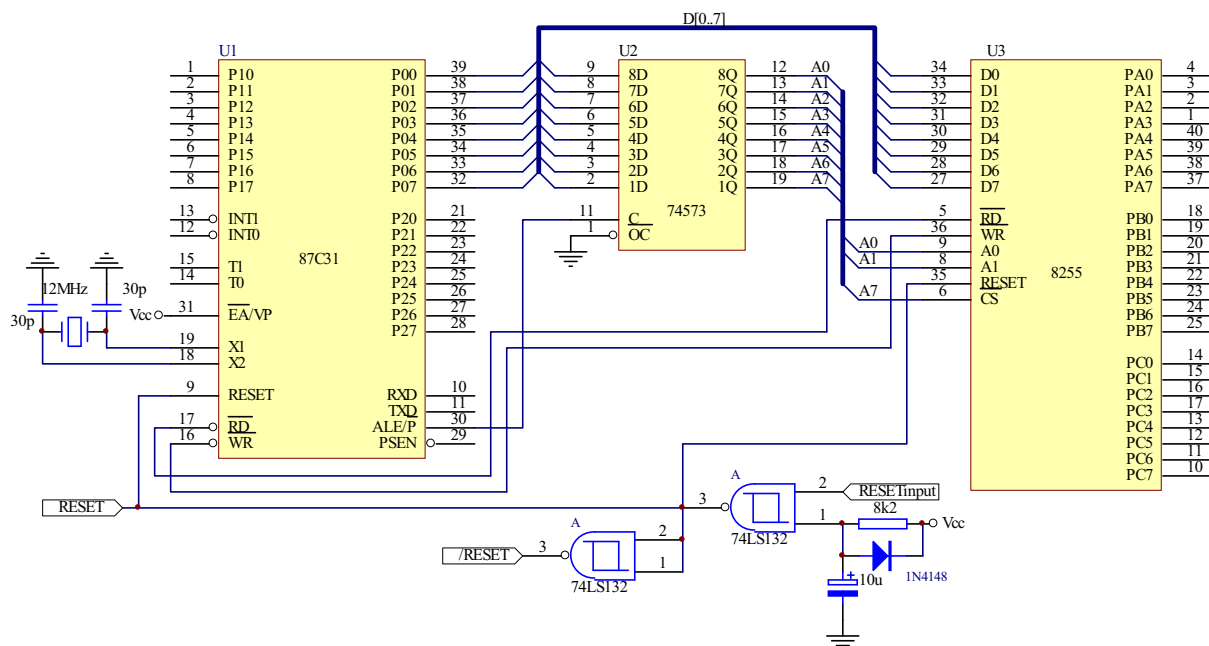
Niniejszy rozdział podaje kilka prostych przykładów łączenia mikroprocesora z innymi układami systemu mikroprocesorowego. Przy projektowaniu systemu bazującego na 8051/52 należy pamiętać o:

- fakcie multipleksowania danych z adresami, co wymaga stosowania dodatkowego zatrasku na młodszy bajt adresowy;
- obciążalności poszczególnych linii mikroprocesora;
- braku specjalnych sygnałów sterujących do portów zewnętrznych (oznacza to jednolity sposób adresowania urządzeń zewn. i pamięci RAM);
- rozdzieleniu przestrzeni adresowych zewnętrznej pamięci programu i danych;
- możliwości stosowania zamiennie różnych odmian mikroprocesora (rodzaj i wielkość wewnętrznej pamięci programu).



Rys.23. Dołączanie układów peryferyjnych mikroprocesora 8085.

Rysunki 23 i 24 przedstawiają przykład wykorzystania typowych układów peryferyjnych rodziny Intel 8080/8085. Szczególnie proste jest dołączenie układu 8155, dostosowanego do multipleksowanej szyny danych mikroprocesora 8085. Przy stosowaniu układów peryferyjnych mikroprocesora 8080 konieczne jest użycie zatrasku zapamiętującego młodszy bajt adresów. W obu przykładach jako sygnały CS i CE wykorzystano po prostu linie adresowe. Jest to rozwiązanie dopuszczalne jedynie w małych systemach. W większych konstrukcjach, zwłaszcza modułowych konieczne jest dokładniejsze generowanie tych sygnałów. Na obu rysunkach występuje zewnętrzny obwód formujący sygnały RESET i /RESET.



Rys.24. Dołączanie układów peryferyjnych mikroprocesora 8080.