

3. Szczegółowy opis kontrolera SAB 80C166.

Szczegółowej analizie poddany zostanie mikrokontroler SAB 80C166 firmy SIEMENS.

3.1 Najważniejsze cechy rodziny C16x.

Rodzina mikrokomputerów jednoukładowych charakteryzuje się następującymi cechami:

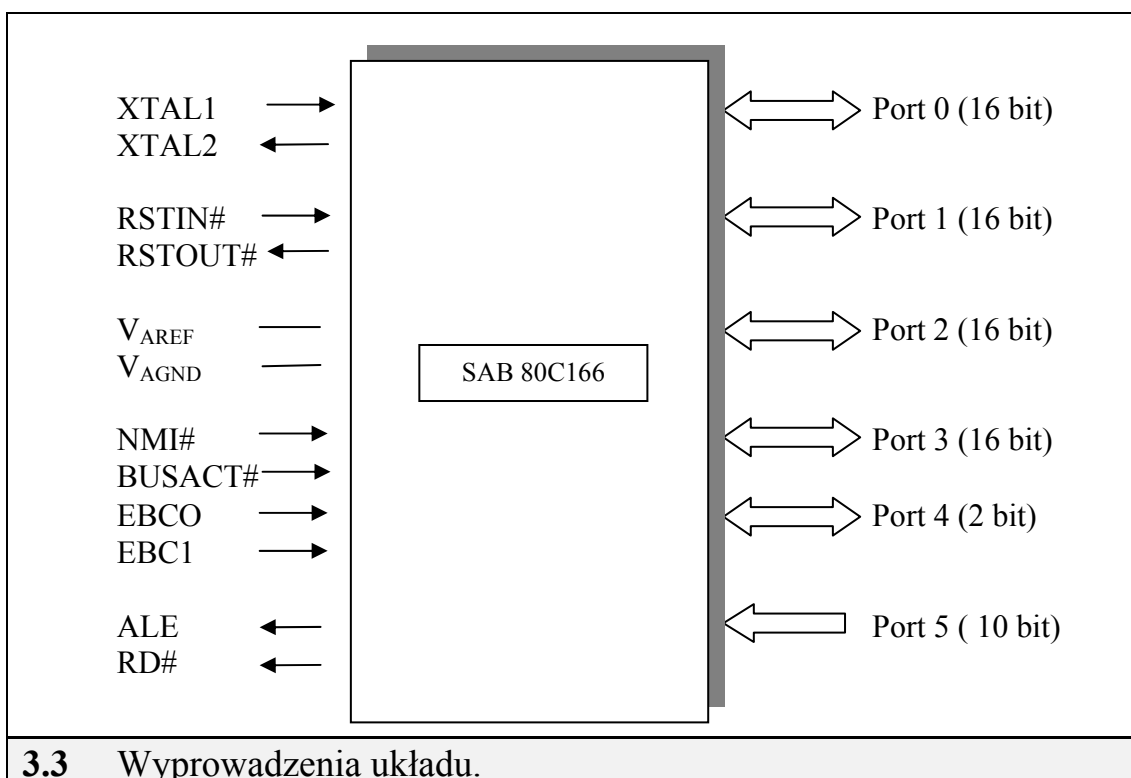
- Wysoka wydajność 16-bitowego procesora z 4-elementową kolejną rozkazów.
- 100ns cykl rozkazowy dla zegara 20 MHz.
- 500ns czas mnożenia (16x16 bitów); 1 μ s na dzielenie (30/16 bitów).
- Rozszerzona i ułatwiona arytmetyka boolowska.
- Wykorzystywanie pojedynczego Context Switching w jednym cyklu rozkazowym.
- Do 256 KB liniowej przestrzeni adresowej.
- Programowana charakterystyka zewnętrznej magistrali dla różnych przestrzeni adresowych.
- 8 lub 16 bitowa magistrala danych.
- Multipleksowana lub niemultipleksowana magistrala adresowa / danych.
- 515 bajtów specjalnych rejestrów funkcyjnych SFR.
- 1 KB wewnętrznej pamięci RAM.
- 32 KB wewnętrznej pamięci ROM typu FLASH.
- Tryb bezczynności i obniżonego poboru prądu.
- 8 kanałowe przerwanie - przeniesienie. Przeniesienie pojedynczego słowa przez obwodowy kontroler zdarzeń (Peripheral Event Controller PEC).
- 16 poziomowy system przerwań.
- 10 kanałowy - 10 bitowy przetwornik analogowo - cyfrowy z 9.7 μ s czasem konwersji.
- 16 kanałowy moduł przechwytyjąco – porównujący.

- Dwa wielofunkcyjne bloki timerów z 5 timerami.
- Dwa porty szeregowy.
- Programowalny Watchdog.
- Do 76 linii wejścia / wyjścia ogólnego przeznaczenia.
- Bezpośrednie wejście zegarowe.
- Wbudowane ładowanie początkowe (Bootstrap Loader).
- 100 - końcówkowa obudowa.

3.2 Wprowadzenie.

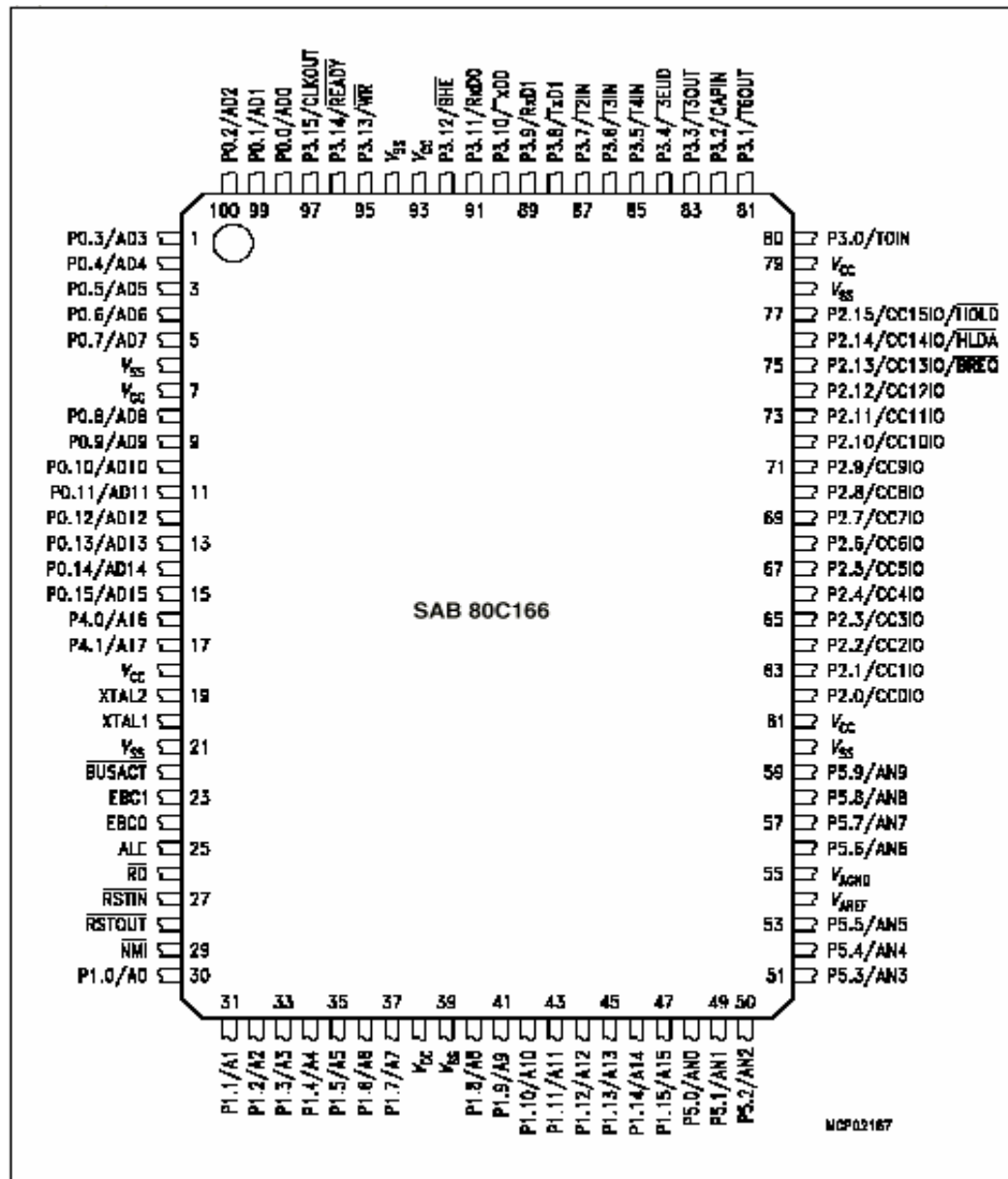
SAB 80C166 jest 16-bitowym reprezentantem rodziny Siemens SAB 80C16x. Posiada szybki procesor (do 10 milionów instrukcji na sekundę) z rozszerzonymi możliwościami portów wejścia / wyjścia. Układy te pobierają sygnał zegarowy procesora bezpośrednio z wewnętrznego oscylatora bez przeskalowania, co redukuje zakłócenia w układzie.

Rysunek 1. Symbol logiczny kontrolera.



3.3 Wyprowadzenia układu.

Rysunek 2. Konfiguracja wyprowadzeń układu.



3.4 Oznaczenia końcówek i ich funkcje.

Tabela 1. Oznaczenia końcówek i ich funkcje.

Numer	Symbol	Wejście I Wyjście O	Funkcja	
16-17	P4.0-P4.1	I/O	Port 4 jest 2-bitowym dwukierunkowym portem wejścia / wyjścia. Każdy z bitów może być niezależnie ustawiany jako wejście lub wyjście. Aby ustawić port jako wyjściowy należy wpisać '1' do odpowiednich bitów konfiguracyjnych DP4.0 i DP4.1. Dla konfiguracji z zewnętrzną magistralą port ten pełni rolę najstarszej części magistrali adresowej (bity A16 i A17).	
16		O	P4.0 A16 Najmniej znaczący segment linii adresowych	
17		O	P4.1 A17 Najbardziej znaczący segment linii adresów.	
20	XTAL1	I	XTAL1 Wejście oscylatora i wewnętrznego generatora.	
19	XTAL2	O	XTAL2 Wyjście generatora. Sygnał taktujący z zewnętrznego układu podłącza się do wejścia XTAL1, natomiast wyjście XTAL2 pozostawia się niepodłączone. Należy pamiętać o zachowaniu odpowiednich charakterystyk czasowych i napięciowych.	
22	BUSACT#	I	External Bus Configuration (Konfiguracja zewnętrznej magistrali). Te trzy wejścia są sprawdzane podczas trwania sygnału RESET i na ich podstawie wybierana jest jedna z dostępnych konfiguracji magistrali zewnętrznej. BUSACT EBC1 EBC0 TRYB KONFIGURACYJNY 0 0 0 8-bitowa demultipleksowana magistrala 0 0 1 8-bitowa multipleksowana magistrala 0 1 0 16-bitowa multipleksowana magistrala 0 1 1 16-bitowa demultipleksowana magistrala 1 0 0 Bez pamięci zewnętrznej 1 0 1 Zarezerwowane 1 1 0 Zarezerwowane 1 1 1 Zarezerwowane	
23	EBC0	I		
24	EBC1	I		
27	RSTIN#	I		Reset Input. Wejście sygnału RESET z charakterystyką Schmitta. Niski poziom na tym wejściu w momencie działania oscylatora wywołuje reset układu.
28	RETOUT#	O		Internal Reset Indication Output. Wyjście sygnału RESET. Sygnał ustawiony jest w stan niski gdy wykonywany jest sprzętowy lub programowy RESET.
29	NMI#	I		Non-Maskable Interrupt. Wejście przerwania niemaszowanego. Przejście ze stanu wysokiego na niski powoduje wykonanie przerwania. W stanie uśpienia (power down PWDN), końcówka NMI musi być w stanie niskim, podanie stanu wysokiego kończy stan uśpienia i procesor wraca do normalnej pracy. Jeśli nie używamy NMI należy ustawić go w stan wysoki (podłączyć do +Vcc).

<i>Numer</i>	<i>Symbol</i>	<i>Wejście I Wyjście O</i>	<i>Funkcja</i>
29	ALE	O	Address Latch Enable - wyjście używane w trybach z multipleksowaną szyną adresową / danych jako sygnał do zapamiętania stanu magistrali przez rejestr typu LATCH.
26	RD#	O	External Memory Read Strobe . Sygnał jest aktywny kiedy procesor czyta z pamięci zewnętrznej.
30-37 40-47	P1.0-P1.15	I/O	Port 1 jest 16-bitowym dwukierunkowym portem wejścia / wyjścia. Każdy z bitów może być niezależnie ustawiany jako wejście lub wyjście. Aby ustawić port jako wyjściowy należy wpisać '1' do odpowiednich bitów konfiguracyjnych DP2.x . Port 1 pełni także rolę magistrali adresowej (bity A0 – A15) przy trybach z niemultipleksowaną magistralą.
48-53 56-59	P5.0 -P5.9	I I	Port 5 jest 10 bitowym portem wejściowym z bramkami Schmitta. Wejścia portu 5 mogą być także wykorzystywane jako wejścia analogowe przetwornika analogowo-cyfrowego.
62-77	P2.0-P2.15	I/O	Port 2 jest 16-bitowym dwukierunkowym portem wejścia / wyjścia. Każdy z bitów może być niezależnie ustawiany jako wejście lub wyjście. Aby ustawić port jako wyjściowy należy wpisać '1' do odpowiednich bitów konfiguracyjnych DP2.x. Opcjonalnie port 2 może pełnić także następujące funkcje:
62		I/O	P2.0 CC0IO CAPCOM: CC0 wejście zliczające lub wyjście porównujące
75		I/O P2.13 CC13IO CAPCOM: CC13 wejście zliczające lub wyjście porównujące
76		O	BREQ# External Bus Request - wyjście
76		I/O	P2.14 CC14IO CAPCOM: CC14 wejście zliczające lub wyjście porównujące
77		O	HLDA# External Bus Hold Acknowl. – wyjście
77		I/O	P2.15 CC15IO CAPCOM: CC15 wejście zliczające lub wyjście porównujące
		I	HOLD# External Bus Hold Request - wejście

<i>Numer</i>	<i>Symbol</i>	<i>Wejście I Wyjście O</i>	<i>Funkcja</i>		
80-92 95-97	P3.0- P3.15	I/O I/O	<p>Port 3 jest 16-bitowym dwukierunkowym portem wejścia / wyjścia. Każdy z bitów może być niezależnie ustawiany jako wejście lub wyjście. Aby ustawić port jako wyjściowy należy wpisać '1' do odpowiednich bitów konfiguracyjnych DP3.x.</p> <p>Opcjonalnie port ten może pełnić następujące funkcje:</p>		
80		I	P3.0	T0IN	CAPCOM Timer T0 Count –wejście
81		O	P3.1	T6OUT	GPT2 Timer T6 Toggle Latch – wyjście
82		I	P3.2	CAPIN	GPT2 Rejestr CAPREL wychwytywanie – wejście
83		O	P3.3	T3OUT	GPT1 Timer T3 Toggle Latch – Wyjście
84		I	P3.4	T3EUD	GPT1 Timer T3 Ext.Up/Down Ctrl.Input
85		I	P3.5	T4IN	GPT1 Timer T4 – Wejście dla zliczania / bramkowania / przeładowania / wychwytywania
86		I	P3.6	T3IN	GPT1 Zegar / Dane – Wyjście (asyn./syn.)
87		I	P3.7	T2IN	GPT1 Zegar T2 – Wejście dla zliczania / bramkowania / przeładowania / wychwytywania
88		O	P3.8	TxD1	ASC1 Zegar / Dane – Wyjście (asyn./syn)
89		I/O	P3.9	RxD1	ASC1 Dane Wejściowe (asyn.) lub I/O (syn.)
90		O	P3.10	TxD0	ASC0 Zegar / Dane – Wyjście (asyn./syn)
91		I/O	P3.11	RxD0	ASC0 Dane Wejściowe (asyn.) lub I/O (syn.)
92		O	P3.12	BHE#	Sygnal wybrania starszej części zewnętrznej pamięci (16-bitowa szyna danych dwa 8-bitowe układy pamięci)
95		O	P3.13	WR#	Sygnal zapisu do pamięci zewnętrznej
96		I	P3.14	READY#	Sygnal gotowości
97		O	P3.15	CLKOUT	Wyjście sygnału taktującego

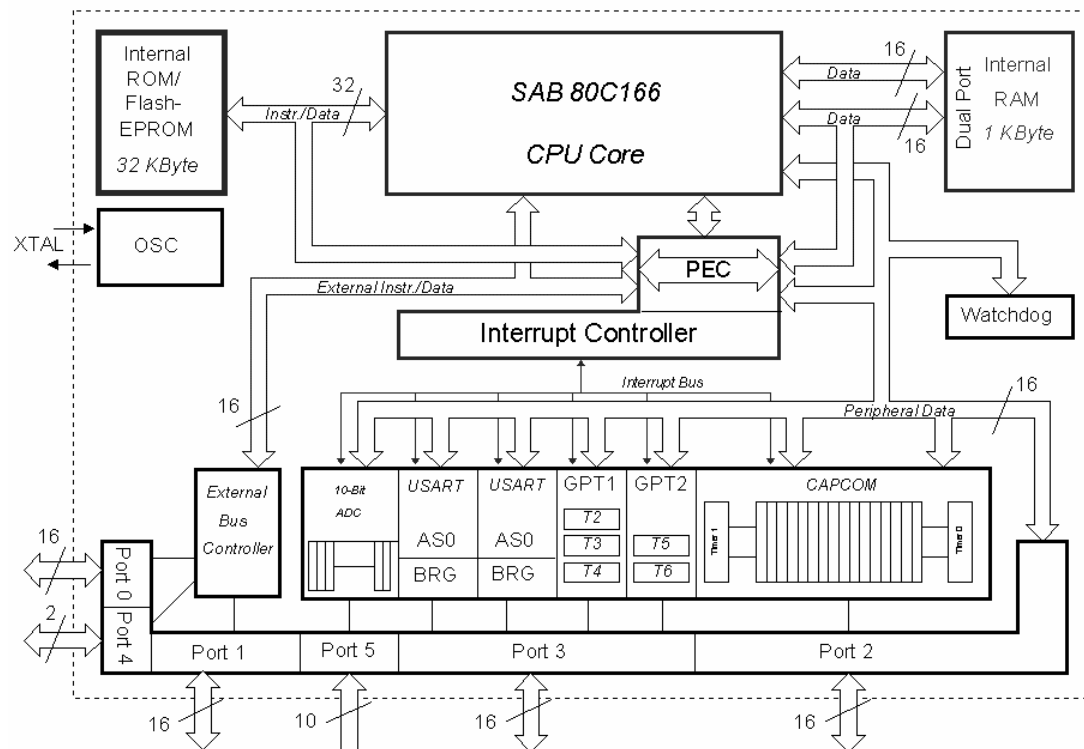
Numer	Symbol	Wejście I Wyjście O	Funkcja																		
98-5 8-15	P0.0- P0.15	I/O	<p>Port 0 jest 16-bitowym dwukierunkowym portem wejścia / wyjścia. Każdy z bitów może być niezależnie ustawiany jako wejście lub wyjście. Aby ustawić port jako wyjściowy należy wpisać '1' do odpowiednich bitów konfiguracyjnych DP0.x.</p> <p>Podczas pracy z zewnętrzną magistralą pełni także rolę magistrali danych lub adresowej i danych.</p> <p>Niemultipleksowany tryb pracy:</p> <table> <tr> <td>Ilość bitów</td> <td>8</td> <td>16</td> </tr> <tr> <td>P0.0-P0.7</td> <td>D0-D7</td> <td>D0-D7</td> </tr> <tr> <td>P0.8-P0.15</td> <td>wyjścia</td> <td>D8-D15</td> </tr> </table> <p>Multipleksowany tryb pracy:</p> <table> <tr> <td>Ilość bitów</td> <td>8</td> <td>16</td> </tr> <tr> <td>P0.0-P0.7</td> <td>AD0-AD7</td> <td>AD0-AD7</td> </tr> <tr> <td>P0.8-P0.15</td> <td>A8-A15</td> <td>AD8-AD15</td> </tr> </table>	Ilość bitów	8	16	P0.0-P0.7	D0-D7	D0-D7	P0.8-P0.15	wyjścia	D8-D15	Ilość bitów	8	16	P0.0-P0.7	AD0-AD7	AD0-AD7	P0.8-P0.15	A8-A15	AD8-AD15
Ilość bitów	8	16																			
P0.0-P0.7	D0-D7	D0-D7																			
P0.8-P0.15	wyjścia	D8-D15																			
Ilość bitów	8	16																			
P0.0-P0.7	AD0-AD7	AD0-AD7																			
P0.8-P0.15	A8-A15	AD8-AD15																			
54	Varef	-	Napięcie odniesienia dla przetwornika AC																		
55	Vagnd	-	Napięcie zerowe dla przetwornika AC																		
7, 18, 38, 61, 79, 93,	Vcc		Napięcie zasilania +5V dla normalnej pracy >2,5V podczas pracy z obniżonym poborem prądu																		
6, 21, 39, 60, 78, 94,	GND		Cyfrowa masa																		

3.5 Opis funkcjonalny.

Architektura SAB 80C166 łączy w sobie dwie technologie: RISC i CISC oraz zaawansowane systemy peryferyjne.

Następujący schemat blokowy przedstawia wszystkie moduły, które tworzą strukturę procesora SAB 80C166.

Rysunek 3. Schemat blokowy.



3.6 Organizacja pamięci.

Obszar pamięci w SAB 80C166 został skonfigurowany według architektury von Neumanna ^[1] w ten sposób, że pamięć kodów, pamięć przeznaczona na dane, rejestry i porty wejścia / wyjścia jest zorganizowana w granicach linearnej przestrzeni adresowej, która wynosi 256KB.

Przestrzeń adresową można rozszerzyć do 16 MB, w które będą zaopatrywane przyszłe wersje. Cała przestrzeń adresowa może być adresowana bajtowo lub całymi słowami. Określone części układów pamięciowych zostały wykonane jako bezpośrednio adresowalne bitowo.

SAB 80C166 posiada 32 KB maskowalnej, programowalnej pamięci ROM przeznaczonej na kody lub dane typu „constant”. Pamięć ROM może być odwzorowywana (mapowana) dla jednego z segmentów: 0 lub 1.

1 KB pamięci RAM jest dostarczany jako pamięć dla użytkownika, jako pamięć przeznaczona na zdefiniowane zmienne, dla systemu stosu, dla banków rejestrów, a nawet dla kodu.

Banki rejestrów mogą przechowywać do 16 słów (R0 – R15) lub bajty (RL0,RH0,....,RL7,RH7). Banki te nazywane są *Rejestrami Ogólnego Celu* (GPR - od ang. General Purpose Registers).

512 bajtów przestrzeni adresowej jest zarezerwowane dla obszaru SFR-ów (*Rejestry Specjalnych Funkcji* – od ang. Special Function Register). SFR-y są rejestrami 16-bitowymi, używanymi do kontroli i monitorowania różnych funkcji modułów jednostki centralnej. Aktualnie jest używanych 98 SFR-ów. Adresy nieużywanych SFR-ów są zarezerwowane dla przyszłych układów rodziny 80C16x.

W przypadku, gdy wymagany jest większy obszar pamięci niż ten dostarczany z układem (czyli powyżej 256 KB) – stosuje się rozszerzenia pamięci RAM lub pamięci ROM, które mogą zostać dołączone do mikrosterownika.

3.7 Zewnętrzny kontroler magistrali.

Dostęp do zewnętrznej pamięci odbywa się za pośrednictwem zewnętrznego kontrolera magistrali (EBC). Kiedy nie jest wymagana dodatkowa pamięć, EBC stanowi zaprogramowany pojedynczy moduł, w jednym z czterech różnych trybach dostępu do zewnętrznej pamięci.

Wyróżniamy cztery tryby dostępu do zewnętrznej pamięci:

- 16 lub 18 bitów adresowych, 16 bitów danych – magistrala multiplekserowana
- 16 lub 18 bitów adresowych, 8 bitów danych – magistrala multiplekserowana
- 16 lub 18 bitów adresowych, 16 bitów danych – magistrala demultiplekserowana
- 16 lub 18 bitów adresowych, 8 bitów danych – magistrala demultiplekserowana

W trybie *demultiplekserowanej magistrali* adresy pojawiają się na wyjściowym Porcie 1, a dane na wejściowo – wyjściowym Porcie 0.

W trybie *multiplekserowanej magistrali* adresy i dane pojawiają się na Porcie 0 (port wejściowo – wyjściowy).

Parametry charakterystyki czasowej magistrali (czas dostępu do pamięci, opóźnienie czytania / pisania, struktura adresu) została tak dobrane, by pozwolić użytkownikowi na możliwość adaptacji różnych układów pamięci.

Dla zastosowań, które wymagają mniej niż 64 KB obszaru pamięci, można wybrać nie segmentowany model pamięci. W tym przypadku cała rozmieszczona pamięć może być adresowana przy użyciu 16-tu bitów. W tym przypadku stosowanie Portu 4 jest zbędne (Port 4 dostarcza dodatkowego segmentu linii adresowych).

3.8 CPU – jednostka centralna.

Główny rdzeń CPU składa się z : 16-to bitowej jednostki arytmetyczno – logicznej (ALU) i specjalistycznych SFR-ów. Wewnątrz CPU znajduje się oddzielna jednostka mnożąco – dzieląca, generator maskowania bitów i część przesuwająca.

Najwięcej instrukcji jest wykonywanych w jednym cyklu maszynowym procesora, który trwa 100ns przy częstotliwości zegara 20 MHz.

Dla przykładu: instrukcje przesuwania i zamiany zawsze są realizowane podczas jednego cyklu maszynowego, niezależnie jaka liczba bitów jest przesuwana. Wszystkie pozostałe cykle instrukcyjne zostały zoptymalizowane w taki sposób, że mogą być wykonywane również bardzo szybko: instrukcje rozgałęzione – w 2 cyklach, mnożenie dwóch 16-to bitowych słów w 5-ciu cyklach i dzielenie słowa 32-bitowego przez 16-bitowe w 10 cyklach.

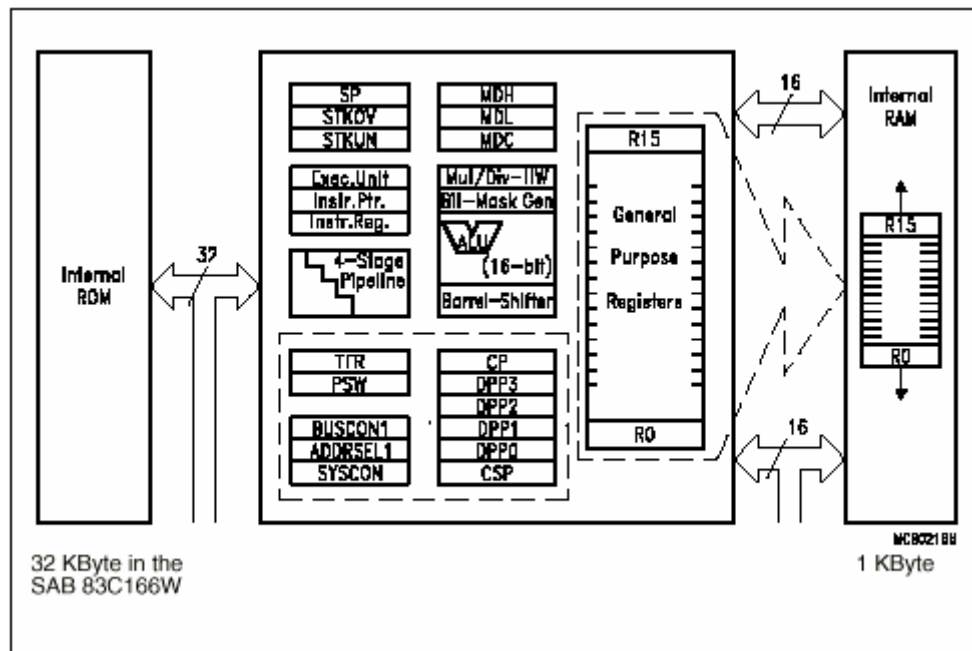
Inny ciąg optymalizacyjny pozwala na skrócenie czasu wykonywania skoków w pętli z 2 do 1 cyklu.

CPU zarządza faktycznymi rejestrami 16-to bitowymi – GPR, które fizycznie są umieszczane w granicach obszaru pamięci RAM. „Context Pointer” (CP) – wskaźniczka kontekstu – to rejestr, w którym umieszczony jest adres bazowy czynnego banku rejestrów, do którego ma dostęp w danym czasie CPU. Numery banków rejestrów są ograniczone przez dostępny wewnętrzny obszar pamięci RAM. Dla łatwiejszego przechodzenia banki rejestrów mogą zachodzić na inne banki.

System stosu do 512 bajtów jest traktowany jak „magazyn” dla tymczasowych danych. Stos jest umieszczany w obszarze pamięci RAM i CPU posiada do niego dostęp poprzez wskaźnik stosu SP (SP – rejestr, od ang. Stack Pointer). Dwa oddziel-

ne SFR-y: STKOV i STKUN domyślnie są porównywane ze wskaźnikiem stosu w celu wykrycia „góry” i „dołu” stosu.

Rysunek 4. Schemat blokowy CPU.



Rdzeniem CPU jest zbiór specjalnych rejestrów zwanych SFR-ami, które pogrupowane są w zależności od pełnionych funkcji:

- Konfiguracja systemu :SYSCON
- Kontrola i identyfikacja statusu CPU :PSW
- Kontrola dostępu kodu :IP, CSP
- Kontrola strony danych :DPP0, DPP1, DPP2, DPP3
- Kontrola dostępu GPR-ów :CP
- Kontrola dostępu systemu stosu :SP, STKUN, STKOV
- Wspomaganie przy mnożeniu i dzieleniu :MDL, MDH, MDC
- Wspomaganie jednostki ALU stałymi :ZEROS, ONES

3.8.1 Rejestry Specjalnych Funkcji CPU – SFR-y.

Rdzeń CPU wymaga zestawu specjalnych rejestrów by utrzymywać system w spójności, obsługiwać ALU, kontrolować konfigurację systemu, wspomagać działania mnożenia i dzielenia, segmentować obszar pamięci, stronicować pamięć i organizować dostęp do systemu stosu.

Mechanizm dostępu do tych SFR-ów jest identyczny jak dostęp do innych tego typu rejestrów.

Rejestry PSW, SP, MDC mogą być modyfikowane nie tylko przez programistę, ale też domyślnie przez CPU podczas przetwarzania normalnej instrukcji.

3.8.2 SYSCON – Rejestr Konfiguracji Systemu.

Jest to rejestr bitowo adresowalny, który dostarcza ogólną konfigurację systemu i kontroluje funkcje. Po RESET może być ustawiany na cztery sposoby, ponieważ bity BTYP i bit ROMEN są inicjowane po RESET zależnie od stanu końcówek wejściowych EBC0 i EBC1.

BTYP	Konfiguracja zewnętrznej magistrali
00b	Brak konfiguracji zewnętrznej magistrali
01b	16/18-bitów adresów, 8-bitów danych, multipleksowana
10b	16/18-bitów adresów, 16-bitów danych, multipleksowana
11b	16/18-bitów adresów, 16-bitów danych, niemultipleksowana

Rysunek 5. Rejestr konfiguracji systemu (SYSCON).

SYSCON(FF0Ch/86h)							Wartość RESET: 0400h, 0040h, 0080h,00C0h								
15	14	13	12	11	10	9	8								
-		STKSZ		RDYEN	SGTDIS	ROMEN	BYTDIS	CLKEN							
7	6	5	4	3	2	1	0								
BTYP		MTTC	RWDC	MCTC											

Tabela 2. Bity SYSCON.

<i>Bit</i>	<i>Pozycja</i>	<i>Funkcja</i>
MCTC	SYSCON[3..0]	Kontrola czasu cyklu pamięci.
RWDC	SYSCON.4	Kontrola zwłoki czytania / pisania.
MTTC	SYSCON.5	Kontrola czasu stanu pamięci.
BTYP	SYSCON[7..6]	Kontrola konfiguracji zewnętrznej magistrali.
CLKEN	SYSCON.8	System Clock Output (CLKOUT) Enable bit: CLKEN = 0: CLKOUT disabled; pin może być użyty jako zwykły I/O. CLKEN = 1: CLKOUT enabled; pin używany do system clock output.
BYTDIS	SYSCON.9	Byte High Enable (BHE#) bit kontrolny: BYTDIS = 0: BHE# enabled. BYTDIS = 1: BHE# disabled; pin może być używany jako zwykły I/O.
ROMEN	SYSCON.10	Internal ROM Access Enable (tylko do odczytu): ROMEN = 0: ROM Access disabled. ROMEN = 1: ROM Access enabled.
SGTDIS	SYSCON.11	Segmentation Disable bit kontrolny: SGTDIS = 0: A16 i A17 enabled: Port 4 używany do adresowania segmentu. SGTDIS = 1: A16 i A17 disabled: Port 4 może być używany jako zwykły I/O.
RDYEN	SYSCON.12	READY# Input Enable bit kontrolny: RDYEN = 0: READY# disabled; pin może być używany jako zwykły I/O. RDYEN = 1: READY# enabled; pin używany jako wejściowy READY#.
STKSZ	SYSCON[14..13]	Maximum System Stack Size Selection pomiędzy 32, a 256 słów.
-	SYSCON.15	Zarezerwowane.

3.8.3 PSW – Słowo Stanu Procesora (od ang. Processor Status Word).

Ten bitowo – adresowalny rejestr odzwierciedla stan mikrosterownika. Jest on podzielony na dwie części, z których pierwszy bajt przedstawia aktualny stan ALU, drugi natomiast przedstawia stan CPU w odniesieniu do systemu przerwań. Oddzielny bit USR0 jest używany jako flaga ogólnego przeznaczenia.

Rysunek 6. Słowo statusowe PSW (rejestr statusowy procesora).

PSW (FF10h/88h)				Wartość RESET: 0000h			
15	14	13	12	11	10	9	8
ILVL				IEN	-	-	-
7	6	5	4	3	2	1	0
-	USR0	MULIP	E	Z	V	C	N

Tabela 3. Bity PSW.

<i>Bit</i>	<i>Pozycja</i>	<i>Funkcja</i>
N	PSW.0	Bit ten reprezentuje negatywny wynik operacji przeprowadzonej przez ALU.
C	PSW.1	Bit ten informuje o przeniesieniu w wyniku operacji ALU.
V	PSW.2	Bit ten informuje o przekroczeniu możliwości zapisania wyniku operacji w jednym słowie.
Z	PSW.3	Bit ten informuje o zerowym wyniku operacji wykonywanej przez ALU.
E	PSW.4	Bit ten poprzedza przeszukiwanie tablicy przez oznaczenie końca tablicy.
MULIP	PSW.5	Bit ten informuje, czy przed zakończeniem operacji mnożenia lub dzielenia nastąpiło zgłoszenie przerwania i wejście do jego obsługi. MULIP = 0: nie wykonywano operacji mnożenia / dzielenia podczas zgłoszenia przerwania MULIP = 1: wykonywano operację mnożenia / dzielenia podczas zgłoszenia przerwania
USRO	PSW.6	Bit ten jest dostarczany jako flaga do wykorzystania przez użytkownika (ogólnego celu).
IEN	PSW.11	Bit umożliwienia przyjęcia przerwania IEN = 0: system przerwania zablokowany IEN = 1: system przerwania odblokowany
ILVL	PSW.[15...12]	Pole to przedstawia aktualny poziom przerwania CPU. Przy wejściu do procedury przerwania te cztery bity określające poziom pierwszeństwa są kopiowane do tego pola. Poprzez modyfikację tego pola można programowo ustalać poziom pierwszeństwa dla aktualnego stanu procesora.
-		Zarezerwowane.

- **Stan ALU przedstawiany za pomocą bajtu PSW** – wartości flag w PSW (N, C, V, Z, E, MULIP) wskazują stan po ostatnio wykonywanych operacjach przez jednostkę arytmetyczno – logiczną (ALU). Bity te są odpowiednio ustawiane w zależności od wyników operacji przeprowadzanych przez ALU.

Flaga E – może być zmieniana przez instrukcje, które wykonuje ALU, albo przez operacje przenoszenia danych. Flaga E jest zerowana przez instrukcje, które nie mogą być użyte do przeszukiwania tablicy danych. We wszystkich innych przypadkach flaga ta jest ustawiana. Jeżeli argument instrukcji (źródła) ma wartość 8000h dla słowa danych lub 80h dla bajtu danych to flaga ta jest ustawiana na ‘1’, w przeciwnym wypadku jest zerowana.

Flaga Z – bit ten jest ustawiany na ‘1’, gdy wynik operacji przyjął wartość zero, w przeciwnym wypadku flaga ta jest zerowana.

Flaga V – bit ten jest ustawiany, gdy wynik mnożenia lub dzielenia nie może być przedstawiony za pomocą jednego słowa. W przeciwnej sytuacji bit ten jest zerowany.

Flaga C – po operacji dodawania ustawiona flaga C informuje o przeniesieniu z najbardziej znaczącej pozycji słowa lub bajtu danych.

Flaga ta zawsze jest zerowana dla logicznych operacji mnożenia lub dzielenia, gdyż operacje te nigdy nie powodują przeniesienia.

Flaga N – ustawiana jest, gdy w wyniku operacji arytmetyczno – logicznej najstarszy bit w słowie lub bajcie wyniku jest ustawiony na wartość ‘1’. W przeciwnym przypadku bit ten jest zerowany.

Flaga MULIP – flaga ta jest ustawiana na ‘1’, gdy w momencie wykonywania działania mnożenia lub dzielenia przez ALU do procesora zostało zgłoszone przerwanie i nastąpiła jego obsługa. W zależności od stanu bitu MULIP jest podejmowana decyzja, czy mnożenie lub dzielenie ma być kontynuowane czy nie. Bit MULIP jest nadpisywany w stosie i odczytywany w momencie powrotu z obsługi przerwania (RETI).

Wszystkie flagi są zerowane po każdym wykonaniu operacji RESET.

- **Status systemu przerw CPU przedstawiony za pomocą bajtu PSW (IEN, ILVL).**

Bit IEN – bit ten umożliwia (IEN = 1) lub uniemożliwia (IEN = 0) przyjmowanie przerw przez CPU.

Pole ILVL – te cztery bity informują o aktualnym poziomie priorytetu CPU. Poziom ten jest aktualizowany przy wejściu do procedury obsługi przerwania, ale może też być zmieniany programowo. W przypadku przyścia przerwania na poziomie 15, obsługa ta może być przerywana tylko przez niemaskowalne przerwanie sprzętowe.

3.8.4 IP – Licznik Rozkazów (od ang. Instruction Pointer).

Ten rejestr określa nie-segmentowany adres instrukcji aktualnie ładowanej do pamięci w granicach segmentu kodu wybranego przez rejestr CSP. IP nie jest odwzorowywany do przestrzeni adresowej mikrokontrolera. IP może być pośrednio modyfikowany przez powrót z podprogramu. IP jest domyślnie aktualizowany przez CPU.

3.8.5 CSP - Wskaźnik Segmentu Kodu (od ang. Code Segment Pointer).

Jest to nie-adresowalny rejestr wyboru segmentu kodu, który jest używany podczas rozpoczynania wykonywania instrukcji. W SAB 80C166 wykorzystywane są tylko dwa bity, a bity od 2 do 15 są zarezerwowane do wykorzystania w przyszłych wersjach mikrosterownika. Rejestr CSP pozwala na dostęp do całej przestrzeni adresowej (aktualnie: do czterech segmentów, po 64 KB każdy).

Rysunek 7. Rejestr wskaźnika kodu segmentu kodu CSP.

CSP (FF08h/04h)				Wartość RESET: 0000h			
15	14	13	12	11	10	9	8
-	-	-	-	-	-	-	-
7	6	5	4	3	2	1	0
-	-	-	-	-	-	SEGNR	

Tabela 4. Bity CSP.

<i>Symbol</i>	<i>Pozycja</i>	<i>Funkcja</i>
SEGNR	CSP[1..0]	Wyszczególnia numer segmentu kodu, gdzie ładowana ma być aktualna instrukcja. Będzie ignorowany w przypadku uniemożliwionej segmentacji.
-	CSP[15..2]	Zarezerwowane.

W trybie segmentowanej pamięci, bity nr 1 i nr 0 rejestru CSP są dostępne na końcówkach A17 i A16 Portu 4.

W trybie nie - segmentowanym domyślnie używany jest segment 0.

Po każdorazowym załączeniu sygnału RESET bajt CSP ustawiany jest na wartość 0000h.

3.8.6 Wskaźniki strony danych – DPP0, DPP1, DPP2, DPP3.

Te cztery nie adresowalne bitowo rejestry decydują o wyborze aktywnych stron danych. Aktualnie jedynie cztery ostatnie bity każdego rejestru DPP są ustawiane lub kasowane, pozostałe (od 4 do 15) są zarezerwowane dla następnych wersji .

Rysunek 8. Rejestry wskaźników strony danych.

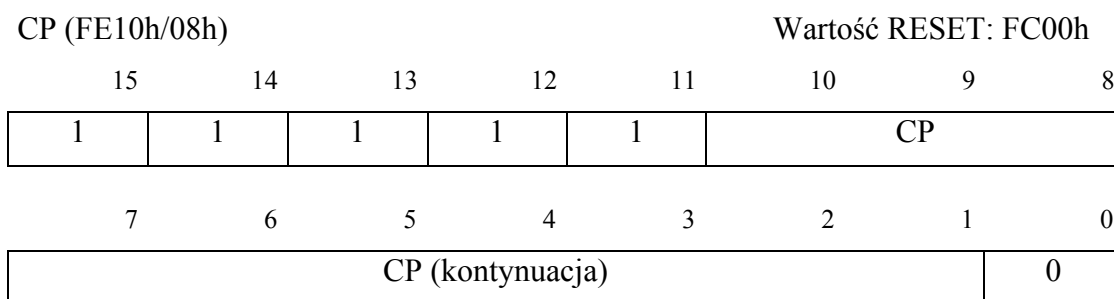
DPP0 (FF00h/00h)	Wartość RESET: 0000h						
DPP1 (FF02h/01h)	Wartość RESET: 0001h						
DPP2 (FF04h/02h)	Wartość RESET: 0002h						
DPP3 (FF06h/03h)	Wartość RESET: 0003h						
15	14	13	12	11	10	9	8
-	-	-	-	-	-	-	-
7	6	5	4	3	2	1	0
-	-	-	-	DPPxPN			

Tabela 5. Bity DPPx.

<i>Symbol</i>	<i>Pozycja</i>	<i>Funkcja</i>
DPPxPN (x=0..3)	DPPx[3..0]	Wyszczególniają numer strony danych wybranej przez DPPx.
-	DPPx[15..4]	Zarezerwowane.

3.8.7 CP – wskaźnik kontekstu.

Rejestr ten bez możliwości adresowania bitowego jest wykorzystywany do wybierania aktualnego rejestru kontekstu. Wartość środkowej części rejestru CP określa adres pierwszego GPR-u w granicach banku rejestru do 16 słów lub bajtów GPR-ów.

Rysunek 9. Rejestr wskaźnika kontekstu (CP).**Tabela 6.** Bity CP.

<i>Symbol</i>	<i>Pozycja</i>	<i>Funkcja</i>
0	CP.0	Bit ustawiany sprzętowo.
CP	CP[10..1]	Pole do modyfikacji w rejestrze CP. Bit 10 jest zawsze zmuszany do odwrotnego stanu niż bit 9. Dla programu bit 10 może być tylko czytany oraz nie jest bezpośrednio zapisywany.
1	CP[15..11]	Bity ustawiane sprzętowo na '1'. Pozwala to na modyfikowanie zawartości rejestru w zakresie: 'FA00h' do 'FDFEh'. Wszystkie ważne adresy GPR-ów muszą być umieszczane w granicach wewnętrznej pamięci RAM.

Po RESET rejestr CP ustawiany jest na wartość 'FC00h'

3.8.8 SP – wskaźnik stosu.

SP jest rejestrem nie - adresowalnym bitowo. Rejestr ten jest używany do wskazywania szczytu wewnętrznego systemu stosu. SP jest rejestrem predekrementowanym, kiedy dane są umieszczane na systemie stosu i postinkrementowanym, kiedy dane są zciągane z systemu stosu. Rejestr SP może być modyfikowany przez każdą instrukcję, która jest zdolna do modyfikacji z SFR. Po RESET rejestr SP jest inicjowany tak długo, jak dynamiczne granice stosu nie przekraczają maksymalnego rozmiaru stosu. Środek rejestru SP bezpośrednio jest odwzorowywany na adres fizyczny systemu stosu.

Rysunek 10. Rejestr wskaźnika stosu (SP).

SP (FE12h/09h)					Wartość RESET: FC00h		
15	14	13	12	11	10	9	8
1	1	1	1	1	SP		
7	6	5	4	3	2	1	0
SP (kontynuacja)							0

Tabela 7. Bity SP.

<i>Symbol</i>	<i>Pozycja</i>	<i>Funkcja</i>
0	SP.0	Bit ustawiany sprzętowo.
CP	CP[10..1]	Pole do modyfikacji w rejestrze SP.
1	CP[15..11]	Bity ustawiane sprzętowo na '1'. Pozwala to na modyfikowanie zawartości rejestru w zakresie: 'F800h' do 'FFFEh'.

3.8.9 STKOV – wskaźnik przeładowania stosu.

Ten nie-adresowalny bitowo rejestr jest porównywany z rejestrem SP po każdej operacji, która umieszczała na stosie dane lub dane z niego zdejmowała. Jeżeli zawartość rejestru SP jest mniejsza niż zawartość STKOV stos „przelewa się” i zgłaszana jest pułapka sprzętowa: $(SP) < (STKOV)$.

Rysunek 11. Rejestr wskaźnika przeładowania stosu (STKOV).

STKOV (FE14h/0Ah)					Wartość RESET: FA00h									
15	14	13	12	11	10	9	8							
1	1	1	1	1	STKOV									
							7	6	5	4	3	2	1	0
STKOV (kontynuacja)														0

Tabela 8. Bity STKOV.

<i>Symbol</i>	<i>Pozycja</i>	<i>Funkcja</i>
0	STKOV.0	Bit ustawiany sprzętowo.
STKOV	STKOV[10..11]	Pole do modyfikacji w rejestrze STKOV.
1	STKOV[15..11]	Bity ustawiane sprzętowo na '1'. Pozwala to na modyfikowanie zawartości rejestru w zakresie: 'F800h' do 'FFFEh'.

3.8.10 STKUN – wskaźnik niedopełnienia stosu.

Ten nie-adresowalny bitowo rejestr jest porównywany z rejestrem SP po każdej operacji, która umieszczała na stosie dane lub dane z niego zdejmowała. Jeżeli zawartość rejestru SP jest większa niż zawartość STKUN stos jest „niedopełniony” i zgłaszana jest pułapka sprzętowa: (SP) > (STKUN).

Rysunek 12. Rejestr wskaźnika niedopełnienia stosu (STKUN).

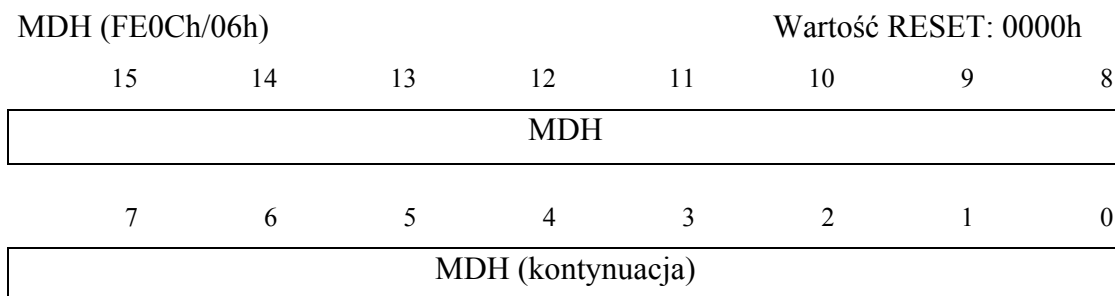
STKUN (FE14h/0Ah)					Wartość RESET: FC00h									
15	14	13	12	11	10	9	8							
1	1	1	1	1	STKUN									
							7	6	5	4	3	2	1	0
STKUN (kontynuacja)														0

Tabela 9. Bity STKUN.

<i>Symbol</i>	<i>Pozycja</i>	<i>Funkcja</i>
0	STKUN.0	Bit ustawiany sprzętowo.
STKUN	STKUN[10..1]	Pole do modyfikacji w rejestrze STKOV.
1	STKUN[15..11]	Bity ustawiane sprzętowo na '1'. Pozwala to na modyfikowanie zawartości rejestru w zakresie: 'F800h' do 'FFFEh'.

3.8.11 MDH – rejestr starszej części mnożenia / dzielenia.

Rejestr ten nie jest adresowalny bitowo. Jest on domyślnie używany przez CPU, gdy ten wykonuje operacje mnożenia lub dzielenia. Po wymnożeniu rejestr ten przedstawia starszą część (16 bitów) ze słowa 32-bitowego. Przed dzieleniem do rejestru MDH ładowana jest starsza część 32-bitowego słowa. Po dzieleniu MDH zawiera 16 bitów reszty.

Rysunek 13. Rejestr starszej części mnożenia / dzielenia – (MDH).**Tabela 10.** Bity MDH.

<i>Symbol</i>	<i>Pozycja</i>	<i>Funkcja</i>
MDH	MDH[15..0]	Wyszczególnienie wyższej części (16-tu bitów) ze słowa 32-bitowego przy mnożeniu i dzieleniu.

Po RESET MDH jest ustawiany na wartość 0000h.

3.8.12 MDL – rejestr młodszej części mnożenia / dzielenia.

Rejestr ten nie jest adresowalny bitowo. Jest on domyślnie używany przez CPU, gdy ten wykonuje operacje mnożenia lub dzielenia. Po wymnożeniu rejestr ten przedstawia młodszą część (16 bitów) ze słowa 32-bitowego. Przed dzieleniem do rejestru MDL ładowana jest młodszą część 32-bitowego słowa. Po dzieleniu MDL zawiera 16 bitów reszty.

Po RESET MDH jest ustawiany na wartość 0000h.

Rysunek 14. Rejestr młodszej części mnożenia / dzielenia – (MDL).

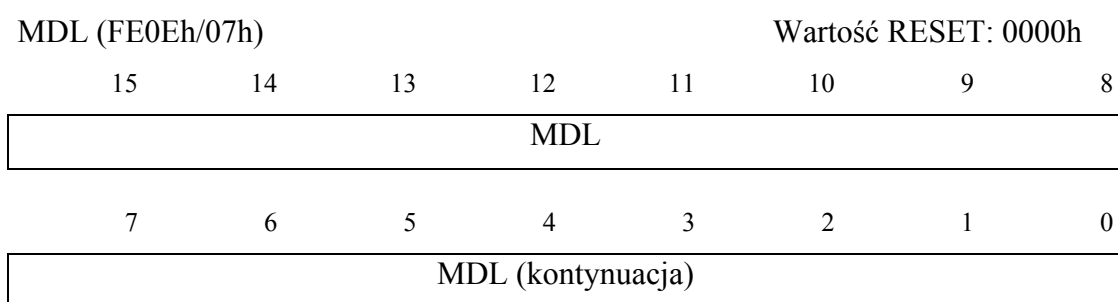


Tabela 11. Bity MDL.

<i>Symbol</i>	<i>Pozycja</i>	<i>Funkcja</i>
MDL	MDL[15..0]	Wyszczególnienie młodszej części (16-tu bitów) ze słowa 32-bitowego przy mnożeniu i dzieleniu.

3.8.13 MDC – rejestr kontrolny mnożenia / dzielenia.

Ten 16-to bitowy, bitowo adresowalny rejestr jest używany domyślnie przez CPU podczas wykonywania operacji mnożenia / dzielenia. Przechowuje on wymagane informacje dla odpowiednich operacji. Rejestr ten jest aktualizowany przez sprzęt podczas każdego pojedynczego cyklu instrukcji mnożenia lub dzielenia.

Rysunek 15. Rejestr kontrolny mnożenia / dzielenia – (MDC).

MDC (FE0Eh/87h)				Wartość RESET: 0000h			
15	14	13	12	11	10	9	8
-	-	-	-	-	-	-	-
7	6	5	4	3	2	1	0
!	!	!	MDRIV	!	!	!	!

Tabela 12. Bity MDC.

<i>Symbol</i>	<i>Pozycja</i>	<i>Funkcja</i>
MDRIV	MDC.4	Bit ten jest ustawiany na '1', kiedy rejestry MDL, MDH są zapisywane przez program, lub gdy wykonywana jest instrukcja mnożenia albo dzielenia. Bit ten jest kasowany, gdy rejestr MDL będzie czytany przez program.
!	MDC[3..0] MDC[7..5]	Bity te są używane przez procesor do kontrolowania wewnętrznie działań mnożenia i dzielenia. Nigdy nie powinny być modyfikowane przez użytkownika z wyjątkiem sytuacji zachowywania, a potem odtwarzania zawartości rejestru MDC.
-	MDC[15..8]	Zarezerwowane.

3.8.14 Rejestr stałych jedynek – (ONES).

Wszystkie bity tego rejestru są sprzętowo ustawione na poziom '1'. Jest to rejestr tylko do odczytu. Rejestr ten może być używany jako rejestr adresowy z samymi jedynekami, dla manipulacji bitami lub jako maska. Może być wykorzystywany przez każdą instrukcję, która jest zdolna do adresowania SFR-ów.

Rysunek 16. Rejestr stałych jedynek – (ONES).

ONES (FE1Eh/8Fh)				Wartość RESET: FFFFh			
15	14	13	12	11	10	9	8
1	1	1	1	1	1	1	1
7	6	5	4	3	2	1	0
1	1	1	1	1	1	1	1

Tabela 13. Bity ONES.

<i>Symbol</i>	<i>Pozycja</i>	<i>Funkcja</i>
1	ONES[15..0]	Wszystkie bity są sprzętowo ustawione w stan '1'. Jest to rejestr tylko do odczytu.

3.8.15 Rejestr stałych zer – (ZEROS).

Wszystkie bity tego rejestru są sprzętowo ustawione na poziom logiczny '0'. Jest to rejestr tylko do odczytu. Rejestr ten może być używany jako rejestr adresowy z samymi zerami, dla manipulacji bitami lub jako maska. Może być wykorzystywany przez każdą instrukcję, która jest zdolna do adresowania SFR-ów.

Rysunek 17. Rejestr stałych zer – (ZEROS).

ZEROS (FF1Ch/8Eh)				Wartość RESET: 0000h			
15	14	13	12	11	10	9	8
0	0	0	0	0	0	0	0
7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0

Tabela 14. Bity ZEROS.

<i>Symbol</i>	<i>Pozycja</i>	<i>Funkcja</i>
1	ZEROS[15..0]	Wszystkie bity są sprzętowo ustawione w stan '0'. Jest to rejestr tylko do odczytu.

Wysokie osiągi uzyskiwane przez sprzęt w stosunku do wydajności jednostki centralnej (CPU) mogą być wykorzystywane przez programistę dzięki zastosowaniu instrukcji następujących klas:

- Rozkazy arytmetyczne
- Rozkazy logiczne
- Rozkazy operujące na algebrze Boolea
- Rozkazy porównania i pętli kontroli
- Rozkazy przesłania i zamiany
- Rozkazy nadrzędne
- Rozkazy przesyłania danych
- Rozkazy realizujące operacje na stosie
- Rozkazy skoku warunkowego i do podprogramu
- Rozkazy powrotu
- Rozkazy systemu kontroli
- Pozostałe instrukcje

Podstawowa instrukcja ma format 2 albo 4 bajtów. Argumenty mogą być : bitami, bajtami lub słowami.

Adresowanie może być: bezpośrednie, pośrednie lub natychmiastowe pod warunkiem, że wyszczególnimy wymagane argumenty.

3.9 System przerwań.

3.9.1 Przerwania i funkcje pułapki.

Architektura systemu SAB 80C166 dysponuje mechanizmami pozwalającymi szybko i elastycznie reagować na żądania obsługi przerwania generowanych ze źródeł wewnętrznych lub zewnętrznych.

3.9.1.1 Zwykły system przerwań.

Przy tego typu systemie aktualnie wykonywany program zostaje zawieszony i mikrokontroler przechodzi do procedury obsługi przerwania obsługując „żądanego”. Stan aktualnie wykonywanego programu zostaje zachowany w wewnętrznym systemie stosu.

Układ 16-to poziomowego systemu przerwań pozwala użytkownikowi na wyszczególnianie tych źródeł przerwań, które mają być obsługiwane jako pierwsze.

3.9.1.2 Przerwania w systemie Zewnętrznego Kontrolera Zdarzeń (PEC).

Mechanizm ten jest bardzo szybki i alternatywny do typowego programowania ukierunkowanego na przetwarzanie przerwań. W tym systemie źródło przerwań jest obsługiwane przez zewnętrzny kontroler zdarzeń (PEC) sterownika SAB 80C166.

Sterownik PEC jest zintegrowany z mikrokontrolerem.

Przy żądaniu obsługi przerwania, PEC ma zdolność przenoszenia pojedynczego słowa lub bajtu danych pomiędzy dwiema lokacjami w pamięci rozmieszczonymi w segmencie 0 przy użyciu jednego z ośmiu programowalnych kanałów usługi PEC.

Podczas operacji przenoszenia normalne wykonywanie programu przez CPU zostaje zawieszona na czas jednego cyklu instrukcyjnego. Dodatkowo żaden stan programu nie musi być na ten okres zachowywany na stosie. Dla usługi PEC stosowany jest identyczny układ priorytetowy jak w przypadku zwykłego systemu przerwań. Przy obsłudze przerwania w systemie PEC, system posiada dwa najwyższe poziomy pierwszeństwa.

3.9.1.3 Funkcje pułapek.

W odpowiedzi na wykonanie pewnych instrukcji funkcje pułapek zostają aktywowane. Pułapka może być spowodowana przez niemaskowalny sygnał z końcówki NMI#. System dostarcza kilku pułapek sprzętowych reagujących na błędne warunki

lub wyjątki powstające podczas wykonywania programu. Pułapki sprzętowe mają najwyższy stopień uprzywilejowania i zawsze powodują natychmiastową reakcję systemu bez względu na to, co jest obecnie wykonywane przez CPU.

Odpowiedni wektor przerwania powoduje start części programu odpowiedzialnego za obsługę pułapki. Dla wszystkich typów pułapek, stan aktualnie wykonywanego programu przerywanego przez obsługę przerwania zostaje zachowany w systemie stosu.

3.9.2 Struktura Systemu Przerwań.

W celu umożliwienia tworzenia konsekwentnego i modułowego oprogramowania każde źródło przerwania albo żądanie usługi PEC jest zaopatrywane w oddzielny rejestr kontroli przerwania i wektor przerwania. Rejestr kontroli zawiera flagę żądania przerwania, bit umożliwiający przerwanie i oznaczenie pierwszeństwa przerwania. Każde źródło przerwania jest aktywowane tylko i wyłącznie przez jedno konkretne zdarzenie. Wyjątek stanowią dwa szeregowy kanały SAB 80C166, gdzie żądanie przerwania błędu może być generowane przez równość dwóch wartości.

SAB 80C166 wyposażony jest w wektorowy system przerwania. W systemie tym niektóre z wektorów rozmieszczone w obszarze pamięci zarezerwowane są dla ponownego ustawiania, dla pułapek i dla funkcji obsługi przerwania. Ilekroć pojawia się żądanie, CPU sięga do jednego z tych rozmieszczeń, które z góry jest określone przez sprzęt. Pozwala to na bezpośrednie identyfikowanie źródła powodującego żądanie. Jedyny wyjątek to pułapki sprzętowe klasy B, które posiadają ten sam wektor adresu. Flaga statusu w Rejestrze Flagowy Pułapki (TFR) określa typ pułapki. Dla specjalistycznego oprogramowania wykorzystującego instrukcje pułapek wektor adresu jest wyszczególniany przez argument instrukcji, który jest 7-mio bitowym numerem pułapki.

W zarezerwowanym obszarze pamięci SAB 80C166 znajduje się tablica skoków. Znajdują się tutaj instrukcje skoków do konkretnych miejsc w pamięci, gdzie umieszczone są faktyczne procedury obsługi przerwania. Wejście do tablicy skoków jest ulokowane na najniższych adresach w kodzie segmentu zerowego przestrzeni pamięci. Odległości między wejściami do tablicy mają odległość 4 bajtów, poza zarezerwowanymi wektorami i wektorami pułapek sprzętowych, w których odległość wynosi 8 lub 16 bajtów.

Następująca tabela zawiera wszystkie możliwe źródła przerwania lub usługi PEC w SAB 80C166 wraz z wektorami przerwania i numerami pułapek.

Tabela 15. Źródła przerwania i związane z tym wektory przerwania.

<i>Źródło przerwania lub żądanie usługi PEC</i>	<i>Flaga żądania</i>	<i>Flaga umożliwienia</i>	<i>Wektor przerwania</i>	<i>Wektor lokalizacji</i>	<i>Numer pułapki</i>
CAPCOM Register 0	CC0IR	CC0IE	CC0INT	40h	10h
CAPCOM Register 1	CC1IR	CC1IE	CC1INT	44h	11h
CAPCOM Register 2	CC2IR	CC2IE	CC2INT	48h	12h
CAPCOM Register 3	CC3IR	CC3IE	CC3INT	4Ch	13h
CAPCOM Register 4	CC4IR	CC4IE	CC4INT	50h	14h
CAPCOM Register 5	CC5IR	CC5IE	CC5INT	54h	15h
CAPCOM Register 6	CC6IR	CC6IE	CC6INT	58h	16h
CAPCOM Register 7	CC7IR	CC7IE	CC7INT	5Ch	17h
CAPCOM Register 8	CC8IR	CC8IE	CC8INT	60h	18h
CAPCOM Register 9	CC9IR	CC9IE	CC9INT	64h	19h
CAPCOM Register 10	CC10IR	CC10IE	CC10INT	68h	1Ah
CAPCOM Register 11	CC11IR	CC11IE	CC11INT	6Ch	1Bh
CAPCOM Register 12	CC12IR	CC12IE	CC12INT	70h	1Ch
CAPCOM Register 13	CC13IR	CC13IE	CC13INT	74h	1Dh
CAPCOM Register 14	CC14IR	CC14IE	CC14INT	78h	1Eh
CAPCOM Register 15	CC15IR	CC15IE	CC15INT	7Ch	1Fh
CAPCOM Timer 0	T0IR	T0IE	T0INT	80h	20h
CAPCOM Timer 1	T1IR	T1IE	T1INT	84h	21h
GPT1 Timer 2	T2IR	T2IE	T2INT	88h	22h
GPT1 Timer 3	T3IR	T3IE	T3INT	8Ch	23h
GPT1 Timer 4	T4IR	T4IE	T4INT	90h	24h
GPT2 Timer 5	T5IR	T5IE	T5INT	94h	25h
GPT2 Timer 6	T6IR	T6IE	T6INT	98h	26h
GPT2 CAPREL Register	CRIR	CRIE	CRINT	9Ch	27h
A/D Conversion Complete	ADCIR	ADCIE	ADCINT	A0h	28h
A/D Overrun Error	ADEIR	ADEIE	ADEINT	A4h	29h
Serial Chanel 0 Transmit	S0TIR	S0TIE	S0TINT	A8h	2Ah
Serial Chanel 0 Receive	S0RIR	S0RIE	S0RINT	Ach	2Bh
Serial Chanel 0 Error	S0EIR	S0EIE	S0EINT	B0h	2Ch
Serial Chanel 1 Transmit	S1TIR	S1TIE	S1TINT	B4h	2Dh
Serial Chanel 1 Receive	S1RIR	S1RIE	S1RINT	B8h	2Eh
Serial Chanel 1 Error	S1EIR	S1EIE	S1EINT	BCh	2Fh

Wektor lokacji sprzętowych pułapek i odpowiednia flaga statusu w rejestrze TFR są przedstawione w tabeli 16. Znajdują się tam również spis priorytetu w przypadku równoczesnych zgłoszeń pułapek. Po każdym zerowaniu (reset sprzętowy, reset programowy przy użyciu instrukcji SRST lub reset spowodowany przepełnieniem watchdog'a) wykonywanie programu rozpoczyna się od adresu 0000h. Procedura zerowania posiada najwyższy priorytet.

Pułapki programowe mogą być wykonywane z wektorami lokacji pomiędzy 0h i 1FCh. Procedura pułapki programowej wykonywana aktualnie przez CPU ma priorytet, który jest wskazywany przez pole ILVL w PSW. Obsługiwana aktualnie pułapka programowa może zostać przerwana przez wszystkie pułapki sprzętowe lub przez pułapkę programową z wyższym priorytetem.

Tabela 16. Zerowanie i wektory lokalizacji pułapek.

<i>Stany wyjątkowe</i>	<i>Flaga pułapki</i>	<i>Wektor pułapki</i>	<i>Wektor lokacji</i>	<i>Numer pułapki</i>	<i>Priorytet pułapki</i>
Zerowanie(RESET):					
-sprzętowe	-	RESET	0h	0h	III
-programowe	-	RESET	0h	0h	III
-przez watchdog	-	RESET	0h	0h	III
Klasa A pułapek sprzętowych:					
-niemaskowalne przerwanie	NMI	NMITRAP	08h	2h	II
-przepełnienie stosu	STKOF	STOTRAP	10h	4h	II
-niedopełnienie stosu	STKUF	STUTRAP	18h	6h	II
Klasa B pułapek sprzętowych:					
-nieokreślony kod operacji	UNDOPC	BTRAP	28h	Ah	I
-chroniona instrukcja błędu	PRTFLT	BTRAP	28h	Ah	I
-nielegalne słowo argumentu dostępu	ILLOPA	BTRAP	28h	Ah	I
-nielegalna instrukcja dostępu	ILLINA	BTRAP	28h	Ah	I
-nielegalny zewnętrzny dostęp do magistrali	ILLBUS	BTRAP	28h	Ah	I
Zarezerwowane			[2Ch-3Ch]	[Bh-Fh]	
Pułapki programowe Instrukcja pułapki			Każde [0h-1FCh] z krokiem 4h	Jakiegokolwiek [0h-7Fh]	Aktualny Priorytet CPU

3.9.3 Zwykła obsługa przerw i usługi PEC.

Pierwszeństwo obsługi przerw i usługi PEC jest zupełnie programowalne. Każde źródło przerwania może być przypisane do konkretnego poziomu pierwszeństwa. Przez jeden cykl procesora wszystkie źródła próbują zgłosić przerwanie do systemu. Specjalny mechanizm (nazywany grupą priorytetu) będzie precyzyjnie rozpatrywał równoczesne żądania z tym samym priorytetem. Przy końcu cyklu tylko jedno źródło (z najwyższym poziomem pierwszeństwa) będzie dopuszczone do obsługi. Źródło to będzie obsługiwane przez CPU jeżeli stopień pierwszeństwa jest najwyższy ze stopni określonych w PSW.

Arbitraż ten, który zdarza się raz w każdym cyklu jest nazywany ciągłą priorytyzacją.

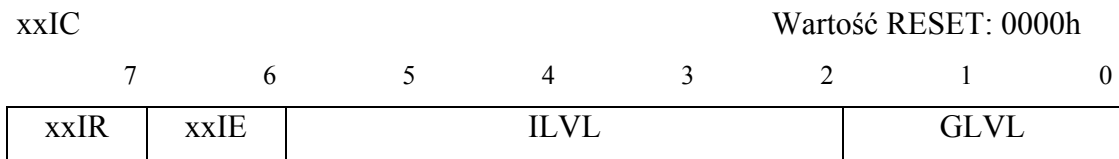
3.9.3.1 Opis rejestru systemu przerw.

Przetwarzanie przerwania jest globalnie kontrolowane bezpośrednio przez PSW i umożliwiające przez bit IEN oraz pole pierwszeństwa ILVL. Dodatkowo inne źródła przerw są kontrolowane indywidualnie przez ich odpowiednie rejestry kontroli. Dlatego wybór przyjęcia odpowiedniego przerwania do obsługi jest określane przez CPU na podstawie indywidualnego rejestru kontroli przerw i PSW. Dla usługi PEC przeznaczony jest dodatkowy rejestr i zawsze muszą być programowane dwa wskaźniki, ażeby móc wyszczególnić zadanie, które ma być wykonywane przez poszczególne kanały usług PEC.

3.9.3.2 Rejestr kontroli przerw.

Wszystkie rejestry kontroli przerw są organizowane identycznie. Rejestr kontroli przerw jest 8-mio bitowy i zawiera wszystkie informacje o źródle, które są wymagane do określenia pierwszeństwa. Wszystko w rejestrze kontroli przerw jest bitowo – adresowalne i wszystkie bity mogą być czytane i ustawiane przez program. Pozwala to każdemu ze źródeł być programowanym przy użyciu jednej instrukcji.

Na rysunku 18 przedstawiono przykład rejestru kontroli przerw xxIC mikrokontrolera SAB 80C166, gdzie xx zastępuje mnemoniczny zapis konkretnego źródła. Każdy rejestr kontroli przerw z jego nazwą i adresem jest wyraźnie pokazany. Funkcje poszczególnych bitów lub pól bitowych zostaną szczegółowo opisane w następnych punktach.

Rysunek 18. Rejestr Kontroli Przerwań dla źródła xx.**Tabela 17.** Bity Rejestru Kontroli Przerwań.

<i>Symbol</i>	<i>Pozycja</i>	<i>Funkcje</i>
GLVL	xxIC [1..0]	Grupa priorytetu przerwań GLVL = 3: najwyższa grupa priorytetu GLVL = 0: najniższa grupa priorytetu
ILVL	xxIC [5..2]	Poziom pierwszeństwa przerwań ILVL = Fh: najwyższy poziom pierwszeństwa ILVL = 0: żądanie nie będzie obsłużone
xxIE	xxIC.6	Bit kontroli odblokowania przerwań xxIE = 0: zablokowanie przerwań xxIE = 1: odblokowanie przerwań
xxIR	xxIC.7	Flaga żądania przerwania xxIR = 0: brak żądania przerwania xxIR = 1: żądanie przerwania

□ **xxIR** – *Flaga Żądania Przerwania.*

Ten bit jest ustawiany sprzętowo w momencie pojawienia się zgłoszenia zdarzenia ze źródła xx. Flaga żądania przerwania jest automatycznie zerowana przy wejściu do obsługi procedury przerwania lub przy usłudze PEC. W przypadku usługi PEC pozostała część flagi żądania przerwania jest ustawiana jeżeli pole COUNT wyboru kanału PEC dąży do zera. Pozwala to na normalną odpowiedź CPU na przerwanie i pełne przenoszenie bloków przy pomocy PEC. Programowe modyfikowanie flagi żądania przerwania powoduje te same skutki, co ustawianie lub zerowanie przez sprzęt.

□ **xxIE** – *Flaga Odblokowania Przerwań.*

Ten bit jest indywidualnie używany do umożliwiania lub uniemożliwiania obsługi żądania przerwania.

□ **ILVL** – Pole Poziomów Pierwszeństwa Przerwań, *xxIC* [5..2]

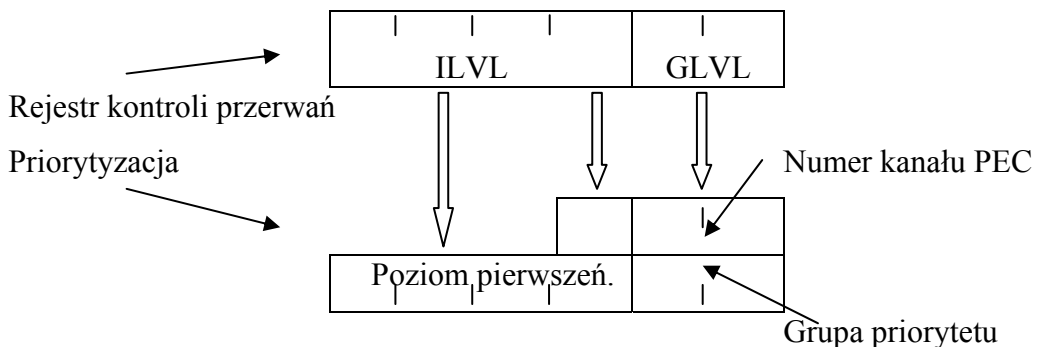
Te cztery bity ściśle określają poziom pierwszeństwa obsługi żądania. Wartości od 0h do Fh wyszczególnione w tym polu przedstawiają poziom pierwszeństwa. Wartość Fh przedstawia najwyższy poziom pierwszeństwa.

Żądania przerwań programowane do poziomów 15 albo 14 będą obsługiwane przez PEC jeżeli pole COUNT dołączane do kanału PEC zawiera zero. W tym przypadku żądanie będzie obsługiwane przez normalne przetwarzanie przerwania. Żądania przerwania, które są programowane od 13 do 1 poziomów zawsze będą obsługiwane przez normalne przetwarzanie przerwania.

Dla sygnałów przerwań, które są wybierane dla usługi PEC przy użyciu metody opisanej powyżej, część LSB z ILVL przedstawia część MSB dołączoną do numeru kanału PEC. W innych słowach programujących źródło na poziomie 15 (ILVL = 1111b) może być wybrany kanał 7 do 4. Przy programowaniu źródła na poziomie pierwszeństwa 14 (ILVL = 1110b) mogą być wybrane kanały od 3 do 0. Faktycznie numer kanału PEC jest określany w polu grupy pierwszeństwa GLVL.

Następujący rysunek przedstawia odwzorowanie pól ILVL i GLVL i ich interpretację podczas ciągłej priorytyzacji.

Rysunek 19. Odwzorowanie pól ILVL i GLVL dla rejestru kontroli przerwań.



Podczas procesu priorytyzacji pola ILVL wszystkich źródeł zgłaszających przerwanie są porównywane z aktualnym poziomem pierwszeństwa CPU, które jest zawarte

w polu ILVL rejestru PSW. Zgłoszenie przerwania o wyższym priorytecie niż aktualne w CPU może spowodować przerwanie wykonywania procesu.

Przy wejściu do obsługi przerwania zgłoszonego przez źródło, które wygrało arbitraż następuje kopiowanie pola ILVL do pól PSW po wysłaniu na stos starego słowa PSW.

System przerwania SAB 80C166 pozwala na zagnieżdżanie do 15 różnych poziomów pierwszeństwa.

Uwaga ! Źródło przerwania, które jest programowane do poziomu pierwszeństwa 0 nigdy nie będzie obsługane przez CPU, ponieważ jego poziom pierwszeństwa nigdy nie będzie wyższy niż poziom CPU.

□ **GLVL** – *Pole grupy priorytetu przerwania.*

Te dwa bity są interpretowane jako względne pierwszeństwo w granicach grupy równoczesnych zgłoszeń od różnych źródeł będących na tym samym poziomie pierwszeństwa. Dla źródła programowanego w ich polach ILVL dla usług PEC, dwa bity GLVL przedstawiają 2 części LSB numeru dołączanego kanału PEC.

Pola grupy pierwszeństwa są szczególnie ważne dla przypadków równoczesnych zgłoszeń od kilku źródeł będących na tym samym poziomie. Istnieje możliwość zaprogramowania 4 źródeł o tym samym poziomie pierwszeństwa. Są one priorytyzowane stosownie do ich grupy pierwszeństwa, gdzie 3 określa grupę o najwyższym priorytecie. Źródła, które zgłosiły równoczesną prośbę obsługi PEC są priorytyzowane stosownie do numeru kanału PEC. Kanał PEC z najwyższym numerem ma najwyższy poziom pierwszeństwa.

Uwaga ! Wszystkie źródła przerwania, które są zaprogramowane na ten sam poziom muszą być przyporządkowane do różnych grup priorytetu. Inaczej dojdzie do generowania niepoprawnego wektora przerwania.

Tabela 18. Przykłady możliwych konfiguracji w rejestrze kontroli przerw.

Pole		Typ usługi (licznik: pole licznika PEC transferu, wybieranego PEC kanału)
ILVL	GLVL	
1111	11	Jeżeli licznik \neq 0: PEC usługa, kanał 7
1111	11	Jeżeli licznik = 0: CPU przerwania, poziom przerw 15, grupa 3
1111	10	Jeżeli licznik \neq 0: PEC usługa, kanał 6
1111	10	Jeżeli licznik = 0: CPU przerwania, poziom przerw 15, grupa 2
1110	11	Jeżeli licznik \neq 0: PEC usługa, kanał 3
1110	11	Jeżeli licznik = 0: CPU przerwania, poziom przerw 14, grupa 3
1110	00	Jeżeli licznik \neq 0: PEC usługa, kanał 0
1110	00	Jeżeli licznik = 0: CPU przerwania, poziom przerw 14, grupa 0
1101	11	CPU przerwania, poziom przerw 13, grupa priorytetu 3
1101	10	CPU przerwania, poziom przerw 13, grupa priorytetu 2
1101	01	CPU przerwania, poziom przerw 13, grupa priorytetu 1
1101	00	CPU przerwania, poziom przerw 13, grupa priorytetu 0
0001	00	CPU przerwania, poziom przerw 1, grupa priorytetu 0
0000	11	Bez usługi
0000	10	Bez usługi
0000	01	Bez usługi
0000	00	Bez usługi

3.9.3.3 Funkcje kontroli przerw w PSW.

Słowo stanu procesora (PSW) funkcjonalnie jest dzielone na 2 części: młodszy bajt PSW przedstawia stan arytmetyczny CPU, starsza część bajtu PSW przedstawia kontrolę systemu przerw SAB 80C166. Organizację PSW przedstawia następujący rysunek:

Rysunek 20. Funkcje kontroli przerw w rejestrze PSW.

PSW (FF10h/88h)				Wartość RESET: 0000h			
15	14	13	12	11	10	9	8
ILVL				IEN	-	-	-
7	6	5	4	3	2	1	0
-	USR0	MULIP	E	Z	V	C	N

□ **ILVL – Pole priorytetu CPU, PSW [15..12]**

Te cztery bity przedstawiają poziom pierwszeństwa jaki w danym czasie posiada CPU. W momencie RESETU pole pierwszeństwa CPU jest ustawiane na najniższym poziomie (tj. poziomie 0). Przy wejściu do usługi przerwania cztery bity z pola poziomu pierwszeństwa ILVL źródła są kopiowane do pola czterech bitów PSW po uprzednim wysłaniu na stos poprzedniej zawartości PSW.

Decyzja, które przerwanie będzie obsługiwane, realizowana jest na podstawie ciągłego przeglądania aktualnego poziomu CPU i porównywania go z poziomami wszystkich nierozstrzygniętych przerw. Modyfikowane pole ILVL informuje o możliwościach programowania poziomu pierwszeństwa, poniżej którego praca CPU nie może być przerwana.

Ponieważ transfer danych przy użyciu usługi PEC zabiera tylko jeden cykl rozkazowy to nigdy nie jest przerywany, ponieważ pola pierwszeństwa nie mają wpływu na zmianę kolejności wykonywanych procesów.

Dla pułapek sprzętowych najwyższy poziom pierwszeństwa dla CPU określony w polu ILVL rejestru PSW –15 jest zawsze wykonywany do końca. Żadna usługa PEC lub inne przerwanie nie są w stanie przerwać tej obsługi.

□ **IEN – Bit kontrolny odblokowania przerw, PSW.11**

Bit ten umożliwia wykonywanie usług PEC. Kiedy bit IEN jest wyzerowany żadne zgłoszenie przerwania nie zostanie obsłużone. Natomiast ustawienie bitu IEN w stan '1' powoduje odblokowanie przerw.

Pułapki niemaskowalne nie są blokowane przez ten bit.

3.9.3.4 Rejestr opisu kanału usługi PEC.

Zewnętrzny kontroler zdarzeń w SAB 80C166 dostarcza użytkownikowi ośmiu kanałów PEC. Kanał PEC jest zdolny do przenoszenia między obszarami pamięci pojedynczych bajtów lub całych słów. Każdy kanał posiada rejestr kontroli i parę wskazówek dla źródła i miejsca przeznaczenia danych.

3.9.3.5 Rejestr zliczająco - kontrolny kanału PEC.

Każdy z ośmiu kanałów PEC jest zaopatrywany w rejestr kontroli. Rejestry te nie są adresowalne bitowo. Rejestr określa jakiego typu zadania mogą być obsługiwane przez dany kanał PEC. Kanał PEC jest określany jednoznacznie przez pola ILVL i

GLVL w rejestrze kontroli przerwań. Następująca tabela przedstawia wszystkie kanały PEC i ich opisy.

Tabela 19. Rejestry zliczajaco - kontrolne PEC.

<i>Rejestr kontrolny</i>	<i>Adres fizyczny</i>	<i>8-bitowy adres</i>
PECC0	FEC0h	60h
PECC1	FEC2h	61h
PECC2	FEC4h	62h
PECC3	FEC6h	63h
PECC4	FEC8h	64h
PECC5	FECAh	65h
PECC6	FECCh	66h
PECC7	FECEh	67h

Rysunek 21. Organizacja rejestru zliczajaco – kontrolnego kanału PEC.

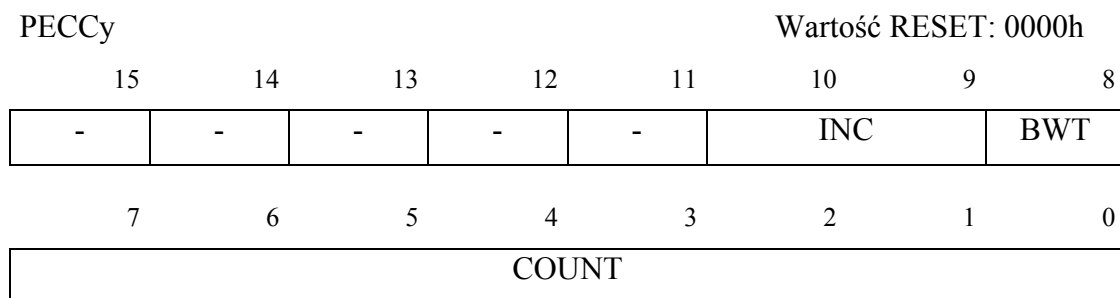


Tabela 20. Bity rejestru zliczająco – kontrolnego kanału PEC.

<i>Symbol</i>	<i>Pozycja</i>	<i>Funkcje</i>
COUNT	PECCy [7..0]	PEC pole licznika transferu: COUNT = FFh tryb kontynuacji transferu. Wartość COUNT nie jest zmniejszana $FE \geq COUNT \geq 1$: Wartość COUNT jest każdorazowo zmniejszana po transferze COUNT = 0: CPU generuje przerwania
BWT	PECCy.8	Bit wyboru rodzaju transferu (bajt lub słowo): BWT = 0: transfer słowa BWT = 1: transfer bajtu
INC	PECCy [10..9]	Zwiększanie pola kontroli: INC = 00b: żaden punkt nie wzrasta INC = 01b: wzrost punktu przeznaczenia INC = 10b: wzrost punktu źródłowego INC = 11b: zarezerwowane
-	PECCy [15..11]	Zarezerwowane

□ **INC – pole kontrolne wzrostu.**

To dwu - bitowe pole informuje, czy wskaźnik źródła lub przeznaczenia kanału PEC będzie zwiększany w momencie przesyłania danych przez PEC. Tylko jeden z tych wskaźników może być zwiększany. W momencie kiedy żaden ze wskaźników nie jest wybierany (INC=00b) to następuje przesyłanie pomiędzy tymi samymi obszarami pamięci.

□ **BWT – bit wyboru rodzaju transferu (słowo lub bajt).**

Bit ten określa w jakiej formie będą przesyłane dane pomiędzy obszarami pamięci.

BWT = 1 to przesyłane będą bajty

BWT = 0 to przesyłane będą całe słowa

W momencie przesyłania bajtu wartość wskaźnika źródła zwiększa się o 1. Gdy przesyłane jest słowo wtedy wartość wskaźnika źródła jest zwiększana o 2.

□ **COUNT – pole licznika transferu PEC.**

To 8-mio bitowe pole używane jest do określania liczby przenoszonych danych przez poszczególne kanały PEC. Można zaprogramować ograniczoną liczbę przesyłań (od 0 do 256) lub nieograniczoną liczbę. Licznik transferu działa na zasadzie 8-mio bitowego licznika w tył. W polu tym dopuszcza się wartości od 0 do FFh.

Gdy wartość COUNT zawiera się pomiędzy FEh a 2, to wartość licznika jest zmniejszana po wykonaniu każdej usługi PEC. Flaga żądania przerwania zostaje wyzerowana.

Gdy wartość COUNT równa się 1 i generowane jest żądanie wykonania usługi PEC to wartość COUNT jest zmniejszana do zera.

Gdy wartość COUNT równa się 0 wtedy żadna usługa PEC nie będzie wykonywana.

3.9.3.6 Budowa poziomów pierwszeństwa.

W mikrokontrolerze SAB 80C166 pierwszeństwo obsługi żądania przerwania lub usługi PEC są całkowicie programowalne. Wszystkie źródła powinny być programowane do różnych poziomów pierwszeństwa. W przypadku gdy źródła programowane są na te same poziomy musi zostać wprowadzony podział na grupy pierwszeństwa. Inaczej mogą istnieć przekłamania w wartości wektora przerwania. Wewnątrz jednego poziomu można przyporządkować przerwania do wszystkich grup pierwszeństwa (od 0 do 4).

Przewagą tego systemu jest to, że procesor może przeglądać przerwania od różnych źródeł na tym samym poziomie, gdzie parametry można zmieniać programowo.

We wszystkich przypadkach źródło z najwyższym poziomem pierwszeństwa w najwyższej grupie jest zawsze obsługiwane jako pierwsze.

System przerwania dostarcza nam 16-tu poziomów przerwania. Ale faktycznie tylko 15 jest efektywnie wykorzystywanych w systemie przerwania, ponieważ poziom 0 nie przerywa pracy procesora.

W rzeczywistości tylko dwa najwyższe poziomy pierwszeństwa (15 i 14) są wykorzystywane przez PEC.

3.9.4 Procedury przerwania.

W przypadku zgłoszenia przerwania o wyższym priorytecie niż priorytet aktualnie obsługiwanego - aktualny stan procesora zostaje wysłany na stos, co zapewnia po-

wrót do miejsca wykonywania, gdzie nastąpiło zgłoszenie przerwania. Informacja przechowywana w systemie stosu zależy od tego, jaka segmentacja jest aktualnie używana. Wskazuje to bit SGTDIS w rejestrze SYSCON.

3.9.4.1 Procedura przerwania z uniemożliwioną segmentacją.

Jeżeli segmentacja jest aktywna to zawartości PSW i IP są wysyłane na stos. Pole określające poziom pierwszeństwa przerwania zostaje skopiowane do pola pierwszeństwa rejestru PSW procesora. Jeżeli przerwanie zostało zgłoszone w momencie wykonywania operacji mnożenia lub dzielenia to bit MULIP w PSW jest ustawiany na '1', flaga źródła przerwania zostaje wyzerowana. CPU przechodzi wówczas do kontroli wektora przerwania. Umieszczona na stosie zawartość IP (adres instrukcji) będzie wskazywała, w które miejsce ma powrócić CPU po obsłudze przerwania.

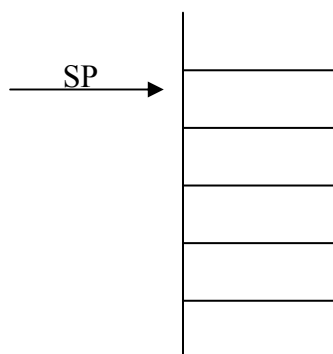
Przy wykonaniu instrukcji RETI (powrót z przerwania) informacje zostają pobierane ze stosu w odwrotnej kolejności niż były tam wysyłane. W ten sposób zostaje przywrócony stan sprzed przerwania. Następujący rysunek pokazuje system stosu podczas przerwania.

Rysunek 22. Procedura przerwania podczas nie-umożliwionej segmentacji.

a). System stosu przed wejściem do procedury obsługi przerwania

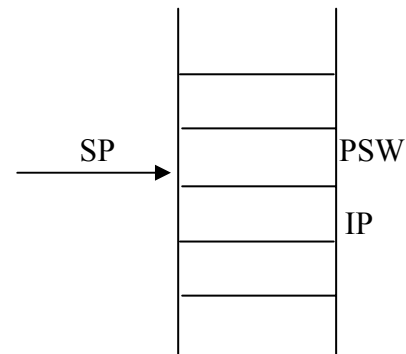
b). System stosu po wejściu do procedury obsługi przerwania

Starszy adres



Młodszy adres

Starszy adres



Młodszy adres

3.9.4.2 Procedura przerwania z umożliwioną segmentacją.

Jeżeli segmentacja jest używana w momencie przyjęcia zgłoszenia przerwania, to wskazówka kodu segmentu (CSP) też jest wysyłana na stos, by zapewnić powrót do poprzedniego segmentu po zakończeniu obsługi przerwania. W pierwszej kolejności na stos wysyłany jest PSW, a zaraz za nim CSP i IP. Tak samo jak w przypadku uniemożliwionej segmentacji tak i w tej sytuacji poziom pierwszeństwa źródła jest kopiowany do pola pierwszeństwa PSW i flaga źródła przerwania zostaje wyzerowana. Jeżeli zgłoszenie przerwania pojawiło się w trakcie wykonywania przez CPU mnożenia lub dzielenia to bit MULIP w PSW jest ustawiany w stan '1'.

Po wykonaniu instrukcji RETI (powrót z przerwania) informacje uprzednio wysłane na stos zostają teraz z niego w odwrotnej kolejności zdejmowane.

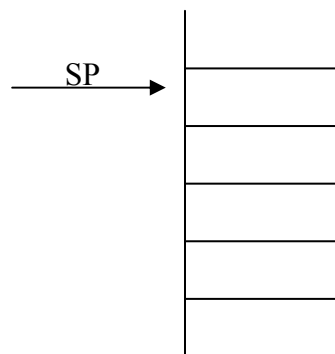
Rysunek 21 przedstawia system stosu w sytuacji, gdy umożliwiona jest segmentacja.

Rysunek 23. Procedura przerwania podczas nieumożliwionej segmentacji.

a). System stosu przed wejściem do procedury obsługi przerwania

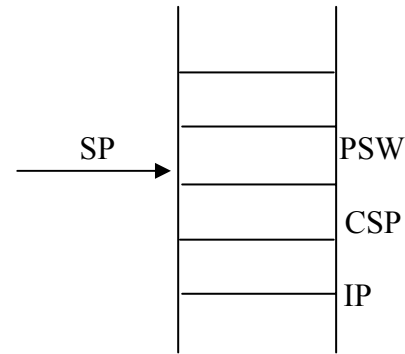
b). System stosu po wejściu do procedury obsługi przerwania

Starszy adres



Młodszy adres

Starszy adres



Młodszy adres

3.9.4.3 Context switching dla usługi przerwania.

Context switching w połączeniu z usługą przerwania pozwala na tworzenie nowych kontekstów w granicach rutynowej usługi przerwania. W tym przypadku dostarczany jest zupełnie inny zestaw rejestrów GPR.

Context switching może być zainicjowany przez wykonanie instrukcji SCXT w granicach procedury usługi przerwania. Np. instrukcją SCXT CP. Nowy używany bank

umieszcza na stosie poprzednią wartość wskaźnika kontekstu CP. Nowy bank jest wyszczególniony natychmiastowo jako argument instrukcji SCXT.

Przed wykonaniem instrukcji RETI przy końcu procedury obsługi przerwania poprzedni wskaźnik kontekstu musi być zdjęty z systemu stosu.

3.9.4.4 Przerwania przetwarzane przez zewnętrzny kontroler zdarzeń PEC.

PEC jest alternatywą do programowego systemu przerwania. Przerwania obsługiwane przez PEC są jakby utajnione ponieważ program wykonywany nie jest informowany o tym co w danym czasie przetwarza PEC. Wszystkie urządzenia peryferyjne w SAB 80C166 są kontrolowane przez SFR-y. System musiałby ich wykorzystywać bardzo dużo, gdyby miały one obsługiwać wszystkie urządzenia. PEC pośredniczy w przenoszeniu danych pomiędzy SFR, a pamięcią w obu kierunkach. Pośredniczy w przechowywaniu wyników z przetwornika A/C lub przechowuje dane z portu szeregowego. Umożliwia także przenoszenie danych pomiędzy różnymi obszarami pamięci wewnątrz jednego segmentu – 0.

Transfer danych PEC nie działa na IP i flagach w PSW. Dlatego żadna informacja o stanie programu podczas wykonywania usługi PEC nie będzie zachowana.

Budowa systemu poziomów przerwania w SAB 80C166 jest tak skonstruowana, że usługa PEC ma pierwszeństwo ponad wszystkimi innymi żądaniami zgłaszanymi do CPU. Wyjątkiem jest tylko sytuacja, gdy CPU posiada poziom 15 lub 14. W momencie, gdy CPU wykonuje procedurę przerwania na poziomie 14 to PEC może jedynie zrealizować przenoszenie danych przez kanały 4,5,6,7. Natomiast w przypadku wykonywania procedury obsługi przerwania na poziomie 15, żadna usługa PEC nie może być zainicjowana.

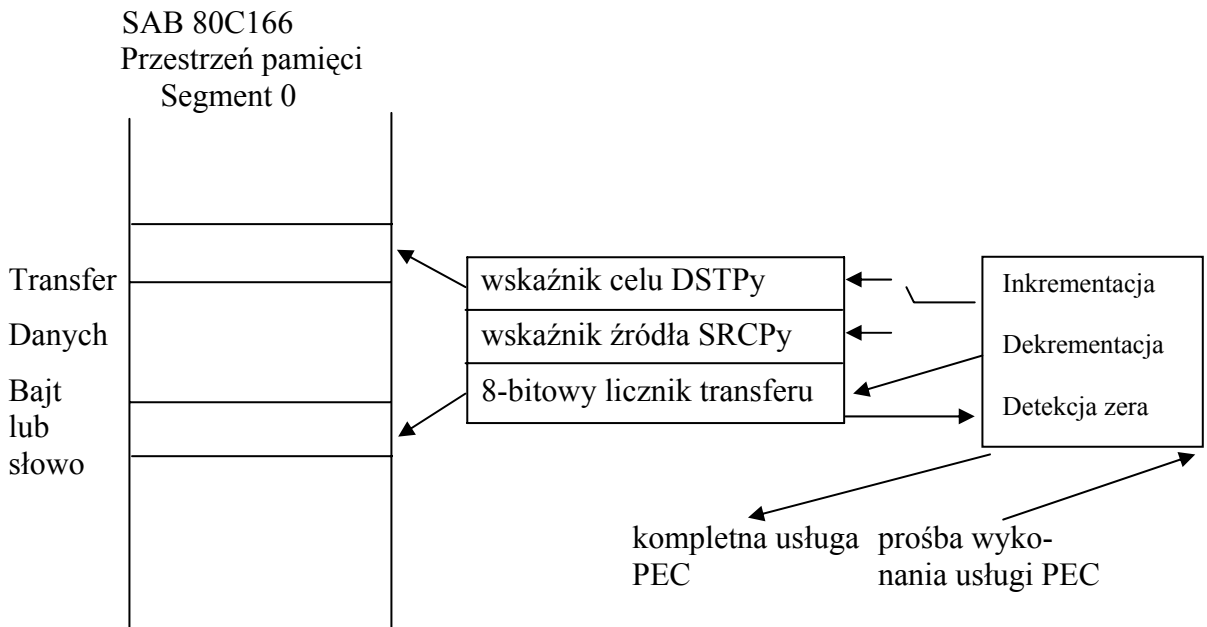
W przypadku, gdy przerwania zaprogramowane dla PEC wybrane są do obwodowej priorytyzacji, to PEC może wykonywać jedynie transfer pojedynczych danych. Typ danych przeznaczonych w tym transferze (bajt lub słowo) jest określony przez bit BWT w rejestrze kontroli PECCy poszczególnego kanału. Źródło i przeznaczenie są określone przez wskaźniki SRCPy (źródło) i DSTPy (przeznaczenie).

Po zakończeniu procedury przenoszenia jeden z dwóch wskaźników musi zwiększyć swoją wartość, a licznik transferu COUNT musi zmniejszyć swoją wartość.

UWAGA! Określone źródła, które będą zgłaszać przerwania do PEC powinny być zaprogramowane do jednego kanału tylko wtedy, gdy istnieje pewność, że nie będą

one zgłaszać się równocześnie. Możliwe są wówczas przekłamania w wektorze przerwania.

Rysunek 23. Procedura usługi PEC.



3.9.4.5 Przerwania zewnętrzne.

W SAB 80C166 dziewiętnaście końcówek portów może służyć jako wejściowe sygnały zgłoszeń przerwania. Opis końcówek znajduje się w tabeli.

Tabela 21. Końcówki portów skonfigurowane jako wejściowe końcówki przerwania.

<i>Pin portu</i>	<i>Symbol alternatywny</i>	<i>Funkcja alternatywna</i>
P2.0	CC0IO	CAPCOM Rejestr 0 Przechwytywanie Wejściowe / Porównywanie Wyjściowe
:	:	:
P2.15	CC15IO	CAPCOM Rejestr 15 Przechwytywanie Wejściowe / Porównywanie Wyjściowe
P3.2	CAPIN	CAPREL Rejestr Przechwytywanie Wejściowe
P3.5	T4IN	Timer 4 Zliczanie / Bramkowanie / Przechwytywanie Wejściowe
P3.7	T2IN	Timer 2 Zliczanie / Bramkowanie / Przechwytywanie Wejściowe

Dla każdej z tych końcówek odpowiednio po zaprogramowaniu stan wysoki albo niski może powodować generację przerwania albo żądanie usługi PEC. Urządzenie peryferyjne musi być przypisane do odpowiedniego trybu przerwania. Pierwszeństwo przerwania jest określone w rejestrze kontroli przerwania dla każdego urządzenia zewnętrznego, a wektor tego przerwania będzie przepisywany do rejestru w przypadku potwierdzenia przerwania.

Aby każda z końcówek opisana w tabeli 21 mogła być używana jako końcówka zgłaszająca przerwanie to bit kontroli kierunku DPx.y odpowiedniego portu w rejestrze kontroli kierunku musi być ustawiony w stan '0'.

3.9.5 Funkcje pułapek.

SAB 80C166 oferuje użytkownikowi dwa różne mechanizmy pułapek. Są to pułapki programowe i sprzętowe. Istnieją funkcje pozwalające ominąć system priorytyzacji, gdzie wymagana jest natychmiastowa reakcja systemu. Funkcje te są niemaskowalne i zawsze mają pierwszeństwo przed żądaniami przerwania przypisanymi do określonych poziomów.

3.9.5.1 Przerwania programowe.

Instrukcja pułapki jest używana do spowodowania wywołania procedury obsługi przerwania. Dołączony do instrukcji pułapki numer może być wyszczególniany w polu argumentu instrukcji. Numer pułapki decyduje o tym jak będzie wyglądał wektor lokalizacji procedury obsługi w przestrzeni adresowej (od 0h do 1FCh). Pokazane jest to w tabeli 16.

Wykonywanie instrukcji pułapki powoduje podobny skutek jak wykonywanie zwykłego przerwania. IP i PSW, a trybie segmentacji również CPS są wysyłane na stos i wykonywany jest skok do odpowiedniego obszaru pamięci (określony przez wektor). W przypadku umożliwionej segmentacji i wykonywanej obsłudze pułapki, CSP jest ustawiany na segment 0. Jeżeli pole pierwszeństwa PSW nie jest modyfikowane to obsługiwane są przerwanie z poziomu, który obowiązywał przy ostatnio wykonywanym przerwaniu. Wykonywanie przerwania może być przerwane tylko i wyłącznie przez przerwanie z wyższym pierwszeństwem. Obsługa przerwania musi zostać zakończona przez instrukcję RETI (powrót z przerwania), która zapewnia poprawny powrót.

3.9.5.2 Przerwania sprzętowe.

Przerwania sprzętowe są używane do identyfikacji defektów albo określonych stanów procesu podczas pracy systemu. Mikrokontroler SAB 80C166 posiada osiem różnych pułapek sprzętowych. Pułapki tego typu są niemaskowalne i mają zawsze pierwszeństwo. Jeżeli wystąpi sytuacja, że zgłoszonych zostanie więcej niż jedna pułapka sprzętowa w tym samym czasie to do obsługi wybierana jest ta, która ma najwyższy poziom pierwszeństwa.

Ileokroć pojawia się pułapka, PSW, IP, w trybie segmentacji także CSP są zapisywane na stosie. Pole określające pierwszeństwo w PSW dla procedury obsługi pułapki jest ustawiane na najwyższy poziom (15), który umożliwia przyjmowanie zawsze wszystkich przerw. Każda pułapka musi kończyć się wykonaniem instrukcji RETI. Osem pułapek sprzętowych przydzielone jest do dwóch klas: A i B. Pułapki klasy A są niemaskowalne i zgłaszane zewnętrznie przez końcówkę NMI#. Wszystkie pułapki z tej klasy posiadają to samo pierwszeństwo ale każda ma oddzielny wektor adresu.

Pułapki z klasy B są następujące:

- ❖ nieokreślony kod pułapki
- ❖ pułapka błędu ochrony
- ❖ nielegalna postać argumentu dostępu pułapki
- ❖ nielegalny dostęp pułapki do zewnętrznej szyny

Wszystkie te pułapki mają ten sam poziom pierwszeństwa i wektor adresujący.

Aby można było zidentyfikować rodzaj pułapki stosuje się Rejestr Specjalnych Funkcji (SFR). Dostępny jest również Rejestr Flagowy Pułapek (TFR). Rejestr ten przedstawiony został na rysunku 24.

Rysunek 24. Rejestr Flagowy Pułapek (TFR).

TFR (FFACh/D6h)				Wartość RESET: 0000h			
15	14	13	12	11	10	9	8
NMI	STKOF	STKUF	-	-	-	-	-
7	6	5	4	3	2	1	0
UNDOPC	-	-	-	PRTFLT	ILLOPA	ILLINA	ILLBUS

Tabela 21. Bity Rejestru Flagowego Pułapek (TFR).

<i>SYMBOL</i>	<i>POZYCJA</i>	<i>FUNKCJA</i>
NMI	TFR.15	Flaga zewnętrznego niemaskowalnego żądania obsługi pułapki. Ustawiana, gdy wykryty zostaje sygnał na końcówce NMI#. Musi być zerowana programowo.
STKOF	TFR.14	Flaga informująca o przekroczeniu zakresu stosu od góry. Ustawiana, gdy wskaźnik stosu jest mniejszy niż zawartość rejestru góry stosu (STKOV). Musi być programowo zerowana.
STKUF	TFR.13	Flaga informująca o przekroczeniu zakresu stosu od dołu. Ustawiany, gdy wskaźnik stosu jest większy niż zawartość rejestru dołu stosu (STKUV). Musi być programowo zerowana.
UNDOPC	TFR.7	Flaga nieokreślonego kod żądania obsługi pułapki. Pojawia się w momencie podania kodu, który nie może zostać zdekodowany. Musi być programowo zerowana.
PRTFLT	TFR.3	Flaga defektu ochrony żądania obsługi pułapki. Ustawiana, gdy pojawia się nielegalny format chronionej instrukcji. Musi być programowo zerowana.
ILLOPA	TFR.2	Flaga nielegalnej postaci argumentu dostępu do obsługi pułapki. Musi być programowo zerowana.
ILLINA	TFR.1	Flaga nielegalnej postaci instrukcji dostępu do obsługi pułapki. Musi być programowo zerowana.
ILLBUS	TFR.0	Flaga nielegalnego dostępu do zewnętrznej magistrali. Musi być programowo zerowana.
-	-	Zarezerwowane.

Każda pułapka posiada inną flagę żądania. Kiedy pojawia się pułapka sprzętowa odpowiednia flaga w rejestrze TFR ustawiana jest w stan '1'. Musi zostać ona następnie programowo wyzerowana.

Pułapki z klasy A mogą przerywać obsługę pułapek z klasy B. Pułapka z końcówki NMI# posiada najwyższy poziom.

Obsługa pułapek z klasy B może być przerywana przez pułapki z klasy A i w takim przypadku ich obsługa będzie dokończona po zakończeniu obsługi pułapki z klasy A. Odkąd wszystkie pułapki z klasy B posiadają identyczne wektory, pierwszeństwo obsługi zgłaszanych jednocześnie przerw z tej klasy jest rozstrzygane programowo.

Klasa pułapek A pojawiających się w trakcie obsługi pułapki z klasy B będzie natychmiastowo obsłużona. Natomiast pojawienie się sygnału żądania obsługi pułapki z klasy B podczas trwania procedury obsługi pułapki z klasy A nie powoduje zmiany procesu. A zatem w tym przypadku pojawienie się pułapki klasy B jest zapamiętywane w rejestrze TFR, ale wartość IP zostaje utracona.

W przypadku, gdy np. pojawi się pułapka nieokreślonego kodu operacji wraz z pułapką z końcówki NMI#, zarówno NMI# jak i flaga UNDOPC są ustawiane, IP z nieokreślonym kodem instrukcji jest umieszczany na stosie, a pułapka NMI jest wykonywana. Następnie wykonywany jest powrót z procedury NMI, IP pobiera ze stosu wartość i natychmiast umieszcza tą wartość z powrotem na stosie z powodu pułapki o nazwie UNDOPC.

3.9.5.3 Zewnętrzna pułapka NMI.

Kiedy tylko pojawia się przerwanie na końcówce NMI#, ustawiana jest flaga w rejestrze TFR i CPU wejdzie do obsługi procedury pułapki NMI. Wartość IP jest wysyłana na stos (w IP znajduje się adres następnej instrukcji, która byłaby wykonywana, gdyby nie pojawienie się pułapki NMI).

3.9.5.4 Pułapka przekroczonego zakresu stosu.

Ileokroć wartość wskaźnika jest zmniejszana i osiąga wartość mniejszą niż wartość rejestru STKOV, flaga STKOF jest ustawiana w rejestrze TFR. Wartość IP będzie przesłana na stos, który oddziałuje na SP poprzez zmniejszanie jego wartości. Podczas ukrytej dekrementacji SP przez wykonywanie instrukcji CALL lub realizację pułapki wartość IP jest umieszczana na stosie jako adres następnej instrukcji.

Dla pewności odzyskiwania informacji ze stosu musi być pewność, że istnieje tam odpowiednia przestrzeń dla zachowywania aktualnego stanu systemu (PSW, IP, CSP). W przeciwnym wypadku powinien być generowany sygnał RESET.

3.9.5.5 Pułapka niedopełnienia stosu.

Ileokroć wskaźnik stosu jest zwiększany i przekroczy wartość rejestru STKUN, flaga STKUF w rejestrze TFR jest ustawiana. Wartość IP będzie przesłana na stos, który oddziałuje na wartość SP. Tak samo, jak w poprzednim przypadku ukryty wzrost wartości SP powoduje przesłanie na stos wartości IP, która wskazuje adres następnej wykonywanej instrukcji.

3.9.5.6 Pułapka nieokreślonego kodu instrukcji.

Ilekoć aktualnie dekodowana instrukcja posiada nieokreślony kod, flaga UNDOPC jest ustawiana w rejestrze TFR i CPU realizuje pułapkę nieokreślonego kodu operacji. Wartość IP umieszczona na stosie zawiera adres instrukcji powodującej pułapkę. Ażeby możliwe było dalsze podejmowanie przetwarzania złożona na stos wartość IP musi zostać zwiększona przez rozmiar nieokreślonej instrukcji, co jest ustalone przez użytkownika, zanim nastąpi instrukcja RETI.

3.9.5.7 Pułapka ochrony defektu.

Ilekoć jedna ze specjalnych chronionych instrukcji jest wykonywana, gdzie kod tej instrukcji nie jest drugi raz powtarzany w drugim słowie instrukcji i bajt następnego kodu operacji nie jest uzupełnieniem kodu, flaga PRTFLT w rejestrze TFR jest ustawiana i następuje obsługa pułapki. Chronione instrukcje włączają DISWDT, EINIT, IDLE, PWRND, SRST, SRVWDT.

IP umieszcza na stosie adres następnej instrukcji.

3.9.5.8 Pułapka nielegalnego słowa argumentu dostępu.

W tym przypadku flaga ILLOPA jest ustawiana i następuje realizacja pułapki. Wartość IP jest umieszczana na stosie, zawiera ona adres następnej instrukcji.

3.9.5.9 Pułapka nielegalnej instrukcji dostępu.

Flaga ILLINA jest ustawiana w rejestrze TFR w momencie wchodzenia do procedury obsługi pułapki. Wartość IP zawierająca adres następnej instrukcji jest przesyłana na stos.

3.9.5.10 Pułapka nielegalnego zewnętrznego dostępu do szyny.

W tym przypadku jeżeli realizowane jest np. ładowanie do pamięci i żadna konfiguracja zewnętrznej szyny nie była wyszczególniona w polu BTYP rejestru SYSCON, flaga ILLBUS zostaje ustawiona w rejestrze TFR i CPU przechodzi do procedury obsługi pułapki. Wartość IP zostaje umieszczona na stosie.

3.10 Układy wejścia - wyjścia.

3.10.1 Porty równoległe.

SAB 80C166 może używać do 76 wejściowo - wyjściowych linii pogrupowanych w pięć portów wejścia - wyjścia i jeden port wyjściowy. Każda linia może być indywidualnie adresowana oraz ustawiana jako wejściowa lub wyjściowa. Linie wejścia - wyjścia są prawdziwymi liniami dwukierunkowymi, które po przełączeniu w stan wejściowy znajdują się w stanie wysokiej impedancji. Po sygnale RESET wszystkie linie pracują jako wejściowe.

Wszystkie porty posiadają także alternatywne wejściowe lub wyjściowe funkcje. *Port 0* i *Port 1* mogą pełnić rolę sygnałów adresowych lub danych przy pracy z zewnętrzną pamięcią. *Port 4* pełni rolę najstarszej części magistrali adresowej - sygnały A16 i A17. *Port 2* może być skojarzony z funkcjami przechwytywania lub porównywania modułu CAPCOM albo ze sterowaniem dostępu do zewnętrznej magistrali sygnałami (BREQ#, HLDA#, HOLD#). *Port 3* zawiera także funkcje powiązane z timerami oraz sygnały sterujące WR#, BHE#, READY# i wyjście zegara systemowego CLKOUT. *Port 5* może być użyty jako kanał wejść analogowych dla przetwornika A/C. Wszystkie linie, które nie są użyte do alternatywnych funkcji mogą pracować jako zwykłe linie wejścia / wyjścia.

3.10.2 Porty Szeregowe.

Komunikacja szeregową z innymi mikrokontrolerami, procesorami, terminalami, lub urządzeniami zewnętrznymi jest oparta na dwóch identycznych interfejsach szeregowych: ASC0 i ASC1.

Są one kompatybilne z portami szeregowymi układów SAB 8051x i ich rodziną. Mogą one pracować z prędkościami do 2.5Mbitów / s przy transmisji synchronicznej i do 625Kbitów / s przy transmisji asynchronicznej.

Dwa dedykowane generatory pozwalają na ustawienie wszystkich standardowych prędkości transmisji danych. Dla nadawania i obioru danych oraz powiadamiania o błędach służą trzy oddzielne przerwania dla każdego z kanałów szeregowych.

W trybie asynchronicznym transmitowane może być 8 lub 9 bitów danych z 1, 1.5 i 2 bitami stopu. Dla komunikacji wieloprocessorowej przeznaczony jest mechanizm oddzielania adresu urządzenia od danych (8-bitów danych + bit pobudzenia).

W transmisji szeregowej dane są transmitowane wraz z sygnałem zegarowym generowanym przez procesor.

Dostępnych jest wiele mechanizmów wykrywania błędów. Bit parzystości jest automatycznie generowany przy nadawaniu i automatycznie sprawdzany przy odbieraniu danych. Błąd ramki jest generowany gdy nie zostanie przesłany bit stopu. Błąd przepełnienia jest generowany gdy bufor odbiorczy nie został odczytany, a następny znak został już w całości przysłany.

3.11 Jednostka wychwytywania / porównania – CAPCOM.

Jednostka CAPCOM generuje i kontroluje czas oraz kolejność dostępu do 16-tu kanałów z maksymalnym rozkładem 400 ns (dot. Zegara 20 MHz). Jednostka CAPCOM typowo wykorzystuje szybkie zadania I/O takie jak impuls i generacja impulsów, impulsy z modulacją (PMW), konwersja cyfrowo – analogowa (C / A), czas systemowy lub czas zapisu dla zewnętrznych zdarzeń.

Dwa 16-to bitowe timery (T0 / T1) na bieżąco ładują rejestry, dostarczając dwóch niezależnych czasów dla wychwytywania / porównywania tablicy rejestrów.

Wejściowy zegar dla timerów jest programowalny do kilku wartości zegara CPU, lub może być wyprowadzany z timera T6 w module GPT2, pod warunkiem, że szeroki zakres zmian czasu dla timera pozwala dokładnie przystosować się do specyfikacji wymagań. W dodatku liczba zewnętrznych wejść dla CAPCOM timera T0 pozwala wpisywać zdarzenia na listę wychwytywania / porównywania rejestrów względem zewnętrznych zdarzeń.

Rejestr wychwytywania / porównywania zawiera podwójną tablicę w celu wychwytywania / porównywania rejestrów, z których każdy może być indywidualnie umieszczony w jednym z timerów w CAPCOM: T0 lub T1 i zaprogramowany dla funkcji wychwytywania lub porównywania. Każdy rejestr posiada jeden port – pin wspólny z wejściowym pinem dla wywoływania funkcji wychwytywania lub jako wyjściowy pin sygnalizujący zdarzenia z porównania zdarzeń.

Kiedy rejestr wychwytywania / porównywania będzie wybrany w trybie wychwytywania, aktualna zawartość timera alokacji będzie zatraskiwana w rejestrze wychwytywania / porównywania w odpowiedzi na zewnętrzne zdarzenie na porcie, które jest dołączane z tym rejestrem. W dodatku odpowiednie żądanie przerwania jest genero-

wane dla rejestru wychwytywania / porównywania. Zawartości wszystkich rejestrów, które były wybierane dla jednego z pięciu trybów porównania są ciągle porównywane z zawartością licznika przydziału. Kiedy dorównanie wystąpi pomiędzy wartością timera, a wartością rejestru wychwytywania / porównania będzie wybrany tryb porównywania.

Tryb 0	Przerwanie jedynie w trybie porównania.
Tryb 1	Przełączanie pomiędzy każdym porównaniem.
Tryb 2	Przerwanie jedynie w trybie porównania.
Tryb 3	Porównywanie w czasie „overflow”.
Tryb podwójnego rejestru	Możliwość porównywania kilku zdarzeń przez timera.

3.12 Ogólny moduł Timerów – GPT.

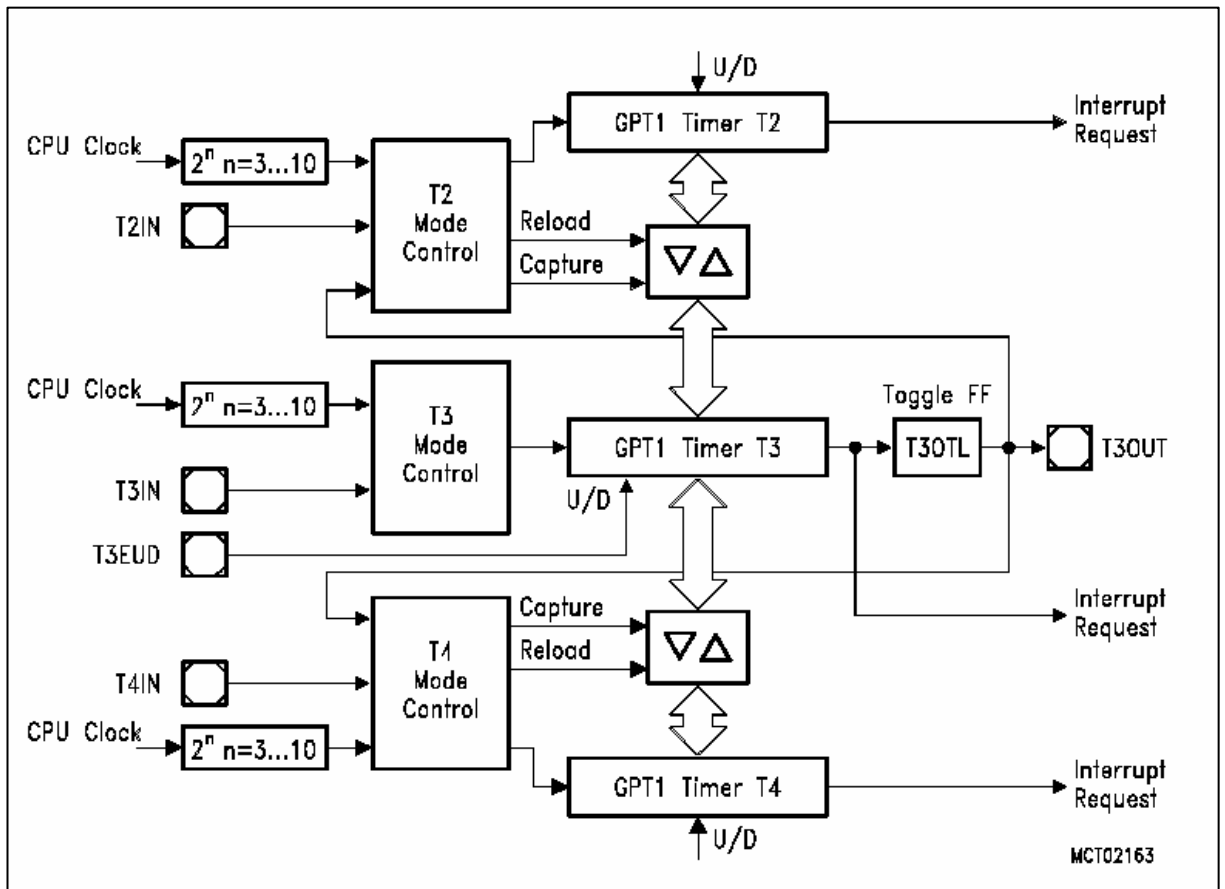
Moduł GPT (od ang. General Purpose Timer) jest wielofunkcyjną strukturą timerów i liczników, które mogą być używane do różnych celów jak np. odmierzenie czasu, pulsowanie z szerokością impulsu, lub jako generator pulsacji.

Moduł GPT składa się z pięciu 16-to bitowych timerów, które podzielone są na dwa osobne bloki: GPT1 i GPT2. Każdy timer, w każdym bloku może posiadać niezależne tryby pracy, lub może być połączony z innymi timerami w tej samej lub innej grupie.

Każdy z trzech timerów T2,T3,T4 modułu GPT1 może być skonfigurowany indywidualnie w jednym z trzech podstawowych trybach pracy: jako timer, bramkowany timer, licznik. W trybie timera wejściowy sygnał zegarowy otrzymywany jest z zegara systemowego podzielonego przez odpowiedni dzielnik, w trybie pracy jako licznik timer jest taktowany wystąpieniem zewnętrznych zdarzeń.

Pulsowanie odbywa się w trybie z bramkowanym timerem, wtedy, gdy zliczanie timera jest blokowane przez odpowiednią końcówkę (TxIN), która blokuje wejście zegarowe timera.

Maksymalna rozdzielczość timerów w module GPT1 wynosi 400 ns (dla 20 MHz zegara CPU).

Rysunek 25. Schemat blokowy modułu GPT1.

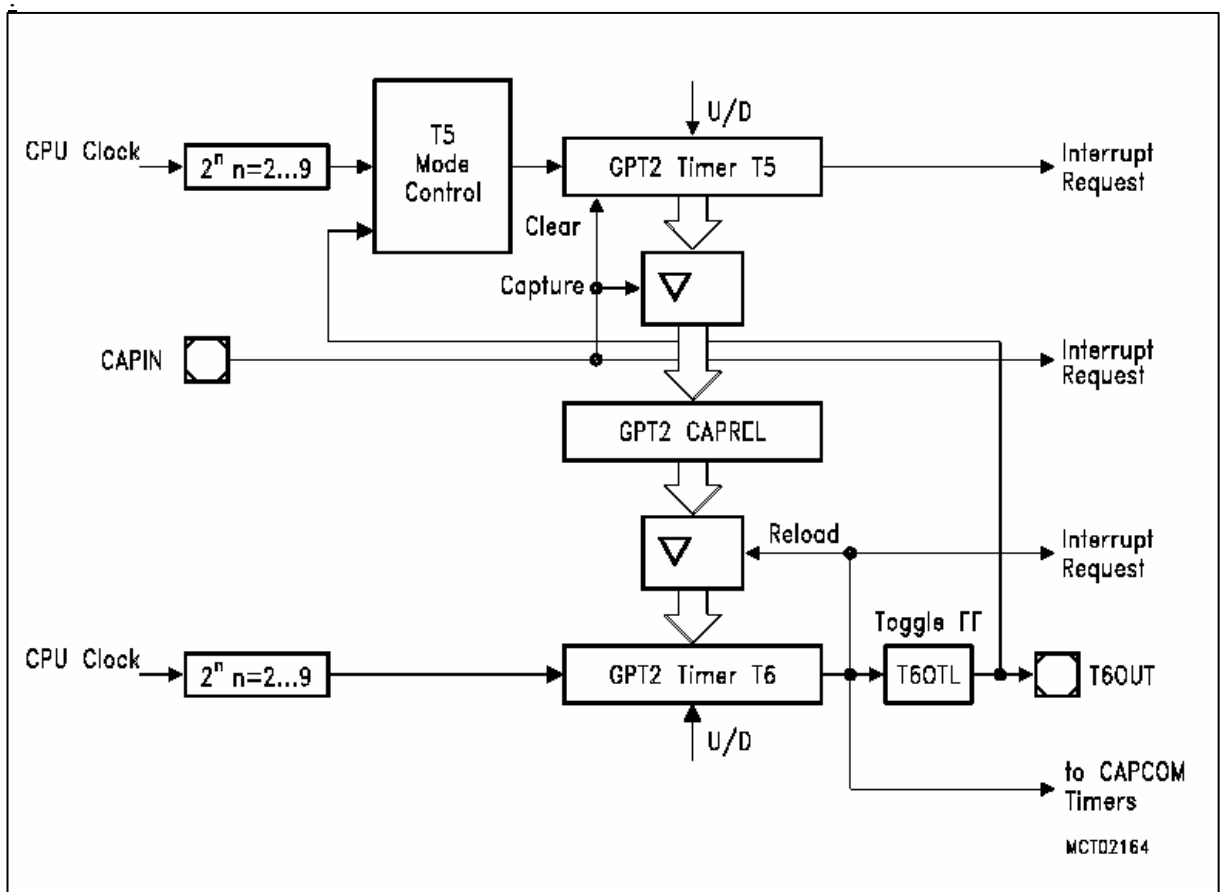
Kierunek zliczania (w górę / w dół) dla każdego timera jest ustawiany programowo. Dla Timera T3 kierunek zliczania może być dynamicznie zmieniany zewnętrznym sygnałem poprzez końcówkę (T3EUD).

Timer T3 posiada także zatraskujące wyjście (T3OTL) które zmienia swój stan po każdym przepełnieniu timera lub dojściu do zera. Stan tego wyjścia może być wyprowadzony także na wyjście (T3OUT) w celu kontroli czasów generowanych przez zewnętrzne urządzenia. Timer T3 może być także użyty wraz z timerami T2 i T4 do mierzenia czasów o długich okresach z wysoką rozdzielczością.

Dodatkowo timery T2 i T4 mogą być użyte do przeładowania lub przechwycenia rejestru timera T3. Kiedy używamy ich jako rejestry do przechwycenia lub przeładowania to timery T2 i T4 są zatrzymywane. Zawartość timera T3 jest przechwytywana przez timery T2 lub T4 w odpowiedzi na odpowiedni sygnał wejściowy (TxIN). Timer T3 jest przeładowywany do timera T2 lub T4 zarówno na zewnętrzny sygnał jak i wybrany stan zatrasku T3OTL. Kiedy oba T2 i T4 są skonfigurowane

alternatywnie do przeładowania timera T3 w przeciwny stan od T3OTL z niskim lub wysokim czasem sygnału PWM, ten sygnał może być generowany bez udziału programu.

Rysunek 26. Schemat blokowy modułu GPT2.



Z maksymalną rozdzielczością 200ns (dla zegara 20 MHz), blok GPT2 nadaje się do precyzyjnej kontroli zdarzeń i upływającego czasu. W jego skład wchodzi dwa timery T5 i T6 oraz przejmujący, przeładowujący rejestr (CAPREL). Oba timery mogą być taktowane wejściem zegarowym podzielonym przez programowy dzielnik. Kierunek zliczania dla każdego timera jest ustawiany programowo na zliczający w dół lub w górę. Powiązanie tych timerów jest możliwe poprzez wyjściowy przerzutnik zatraskujący T6OTL należący do timera T6, który zmienia stan przy każdym przepełnieniu lub dojściu do zera.

Stan tego zatrzasku może być użyty jako zegar timera T5, lub może być sygnałem wyjściowym na końcówce T6OUT układu. Przepiętnienie / dojście do zera może być także użyte jako zegar dla CAPCOM timerów T0 i T1 lub też jako przyczyna przeładowania z rejestru CAPREL. Rejestr CAPREL może przejąć wartość timera T5 na zewnętrzny sygnał z końcówki CAPIN, oraz T5 może być opcjonalnie kasowany po operacji przepisania.

3.13 Przetwornik analogowo - cyfrowy.

Do pomiarów analogowych przeznaczony jest 10-bitowy przetwornik analogowo-cyfrowy z dziesięcioma multipleksowanymi wejściami wbudowanymi w układ. Używa on metody sukcesywnej aproksymacji. Pojedyncze przetwarzanie może maksymalnie trwać do 9.7 μ s dla zegara 20 MHz .

Przetwornik posiada wykrywanie błędów i ochronę, które można sprawdzić w rejestrze rezultatu przetwarzania ADDAT : przerwanie jest generowane jeżeli rezultat poprzedniego przetwarzania nie został jeszcze odczytany, a upłynął już czas potrzebny na następną konwersję.

Dla aplikacji, które potrzebują mniej niż 10-ciu pomiarów analogowych pozostałe wejścia portu 5 można wykorzystać jako wejścia cyfrowe.

Przetwornik A/C wbudowany w SAB 80C166 posiada cztery różne tryby konwersji. W trybie „Standardowy Pojedynczy Kanał” (z ang. Single Mode) analogowa wartość określonego kanału jest przetwarzana na wartość cyfrową . W trybie „Ciągły Pojedynczy Kanał” (z ang. Single Channel Continuous) wartość analogowa jest ciągle przetwarzana bez udziału programu. W trybie „Automatyczne Przetwarzanie” (z ang. Auto Scan) wartość analogowa z wybranych kanałów jest próbkowana i przetwarzana. W trybie „Ciągłym” (z ang. Auto Scan Continuous) wybrane kanały są bezustannie próbkowane i przetwarzane.

Obwodowy kontroler zdarzeń może być wykorzystany do automatycznego zapamiętywania rezultatów przetwarzania i umieszczania w tablicy lub pamięci dla dalszych obliczeń, bez potrzeby wchodzenia i wychodzenia z procedury przerwań dla każdego transfery danych.

3.14 Watchdog Timer.

Watchdog jest specjalnym mechanizmem zabezpieczającym działanie programu.

Jest to specjalny Timer który po przepelnieniu generuje sygnał RESET. Aby się nie przepelniał powinien być okresowo kasowany przez program. Jeśli program się zapętlili i nie skasuje Watchdoga to system zostanie wyzerowany.

Watchdog jest zawsze aktywny po sygnale RESET i można go wyłączyć tylko w czasie trwania inicjacji systemu. W czasie trwania wewnętrznego sygnału RESET wywołanego przez Watchdoga pojawia się także sygnał zewnętrzny RSTOUT, który powinien kasować pozostałe urządzenia podłączone do procesora.

Timer Watchdoga jest 16-bitowy, taktowany z zegara systemowego podzielonego przez wartość od 2 do 128. Starsza część rejestru Watchdoga może być użyta do wpisania wartości przeładowania (rejestr WDREL), dla ustalania różnych czasów oczekiwania.

Czas na wyłączenie watchdoga po sygnale RESET jest standardowo ustalony na 6 ms.

3.15 Ładowanie początkowe - Bootstrap Loader.

SAB 80C166 ma wbudowany specjalny tryb ładowania początkowego (od ang. Bootstrap Loader - BSL), który pozwala na wczytanie krótkiego, 32-u bajtowego programu do wewnętrznej pamięci RAM i uruchomienie go. Wczytanie programu odbywa się przez interfejs szeregowy ASC0. Nie potrzeba do tego wewnętrznego ROM-u ani zewnętrznej pamięci RAM. Uruchomienie BSL następuje gdy sygnał ALE jest w stanie wysokim pod koniec trwania zewnętrznego sygnału RESET, a bezpośrednio po nim następuje przerwanie NMI. Po uruchomieniu BSL SAB 80C166 skanuje linie RxD0, aż do przyjęcia zerowego bajtu wraz z bitem startu i stopu. W czasie trwania tego bajtu ustawiana jest prędkość transmisji, inicjowany jest port ASC0. Używając obliczonej prędkości procesor zwraca identyfikującą go wartość (dla modelu 80C166W jest to wartość 55H). Następnie wysyłane są kolejno 32 bajty do wewnętrznej pamięci RAM i umieszczane od adresu 0FA40h do 0FA5Fh oraz uruchamianie programu od adresu 0FA50h. Program BSL służy zazwyczaj do załadowania właściwego programu. Opuszczając tryb BSL należy wykonać progra-

mowy RESET, ignorowany jest wtedy stan linii ALE, lub sprzętowy RESET lecz należy pamiętać wtedy o odłączeniu sygnału ALE. Program BSL jest umieszczony w specjalnej pamięci ROM. Jest on niewidzialny i niedostępny w czasie normalnej pracy.

3.16 Interfejs szeregowy.

Komunikacja szeregową z innymi mikrokontrolerami, procesorami, terminalami, lub urządzeniami zewnętrznymi jest oparta na dwóch identycznych interfejsach szeregowych ASC0 i ASC1. Operacje na portach szeregowych ASC0 i ASC1 są kontrolowane przez adresowane bitowo rejestry S0CON i S1CON.

Rysunek 27. Rejestry S0CON i S1CON.

S0CON (FFB0h/D8h)						Wartość RESET: 0000h	
15	14	13	12	11	10	9	8
S0R	S0LB	-	-	-	S0OE	S0FE	S0PE
7	6	5	4	3	2	1	0
S0DEN	S0FEN	S0PEN	S0REN	S0SPT	S0M		

S1CON (FFB8h/DCh)						Wartość RESET: 0000h	
15	14	13	12	11	10	9	8
S1R	S1LB	-	-	-	S1OE	S1FE	S1PE
7	6	5	4	3	2	1	0
S1DEN	S1FEN	S1PEN	S1REN	S1SPT	S1M		

Tabela 22. Bity Rejestrów S0CON i S1CON.

<i>Symbol</i>	<i>Pozycja</i>	<i>Funkcja</i>
SxM	SxCON[2..0]	ASCx wybór trybu pracy .
SxSTP	SxCON.3	Wybór liczby bitów stopu. SxSTP=0 : Jeden bit stopu. SxSTP=1 :Dwa bity stopu.
SxREN	SxCON.4	Zezwolenie na odbiór znaków. Kasowane sprzętowo po każdym bajcie w transmisji synchronicznej. SxREN = 0; Odbieranie zabronione. SxREN = 1; Odbieranie dozwolone.
SxPEN	SxCON.5	Sprawdzanie bitu parzystości SxPEN = 0 ; Sprawdzanie bitu parzystości wyłączone. SxPEN = 1 ; Sprawdzanie bitu parzystości włączone.
SxFEN	SxCON.5	Sprawdzanie ramki SxFEN = 0; Sprawdzanie ramki wyłączone. SxFEN = 1; Sprawdzanie ramki włączone.
SxOEN	SxCON.7	Sprawdzanie przedłużania SxOEN = 0; Przedłużanie wyłączone. SxOEN = 1; Przedłużanie włączone.
SxPE	SxCON.8	Błąd parzystości. Ustawiany sprzętowo po wystąpieniu błędu parzystości (gdy SxPEN=1). Musi być kasowany programowo.
SxFE	SxCON.9	Błąd ramki. Ustawiany sprzętowo po wystąpieniu błędu ramki (gdy SxFEN=1). Musi być kasowany programowo.
SxOE	SxCON.10	Błąd przedłużania. Ustawiany sprzętowo.
SxLB	SxCON.14	Tryb pracy w pracy w pętli zamkniętej. SxR = 0; wyłączony. SxR = 1; włączony.
SxR	SxCON.15	Generator włączony. SxR = 0 ;Generator zatrzymany. SxR = 1;Generator załączony.
-	-	Zarezerwowane.

Tryb pracy można ustawić niezależnie dla każdego portu poprzez słowo sterujące S0M (S1M) w rejestrze S0CON (S1CON) według tabeli 23.

Tabela 23. Bit y Słów sterujących S0M (S1M).

<i>S0M / S1M</i>			<i>Tryb</i>	
(2)	(1)	(0)		
0	0	1	8 bitowe dane	Transmisja asynchroniczna
0	1	1	7 bitowe dane + bit parzystości	Transmisja asynchroniczna
1	0	0	9 bitowe dane	Transmisja asynchroniczna
1	0	1	8 bitowe dane + bit pobudzenia	Transmisja asynchroniczna
1	1	1	8 bitowe dane + bit parzystości	Transmisja asynchroniczna
X	1	0	8 bitowe dane	Transmisja synchroniczna

□ **Ramka.**

Każda dana przesyłana przy transmisji asynchronicznej przesyłana jest w specjalnej ramce, która składa się z:

- Jednego bitu startu.
- 8 albo 9 bitów danych, wybieranie przez S0M / S1M.
- Jednego lub dwóch bitów stopu, wybór przez bit S0STP / S1STP w rejestrze S0CON / S1CON.

Rysunek 28 przedstawia ramkę z 8-ma bitami danych. Bit D7 może służyć jako ósmy bit danych lub jako bit parzystości (w zależności od trybu pracy).

Rysunek 28. Ramka dla 8-miu bitów danych.

Bit Startu	D0	D1	D2	D3	D4	D5	D6	D7	1-szy Bit Stopu	2-gi Bit Stopu
------------	----	----	----	----	----	----	----	----	-----------------	----------------

Następny rysunek przedstawia ramkę dla dziewięciu bitów danych. Bity D0 – D7 są bitami danych. Bit D8 może być skonfigurowany jako dziewiąty bit danych, bit parzystości lub bit pobudzenia w specjalnym trybie komunikacji wieloprocessorowej.

Rysunek 29. Ramka dla 9-ciu bitów danych.

Bit										1-szy	2-gi
Start	D0	D1	D2	D3	D4	D5	D6	D7	D8	Bit	Bit
										Stop	Stop

D8 – bit parzystości lub pobudzania.

□ **Asynchroniczne nadawanie.**

Nadawanie zostaje zapoczątkowane po wpisaniu danej do bufora nadawania S0TBUF (S1TBUF). (Należy pamiętać o wcześniejszym włączeniu zegara, bit S0R=1 (S1R=1)). Rejestry te są tylko do zapisu i nie są adresowalne bitowo. Po transmisji bufor zostaje kasowany do wartości 0000h. O zakończeniu transmisji informuje ustawiony bit S0TIR (S1TIR), który może wywoływać przerwanie. Przed wysłaniem kolejnego znaku należy go programowo skasować (bit S0TIR (S1TIR)). Bit parzystości jest ustawiany automatycznie.

□ **Asynchroniczne odbieranie.**

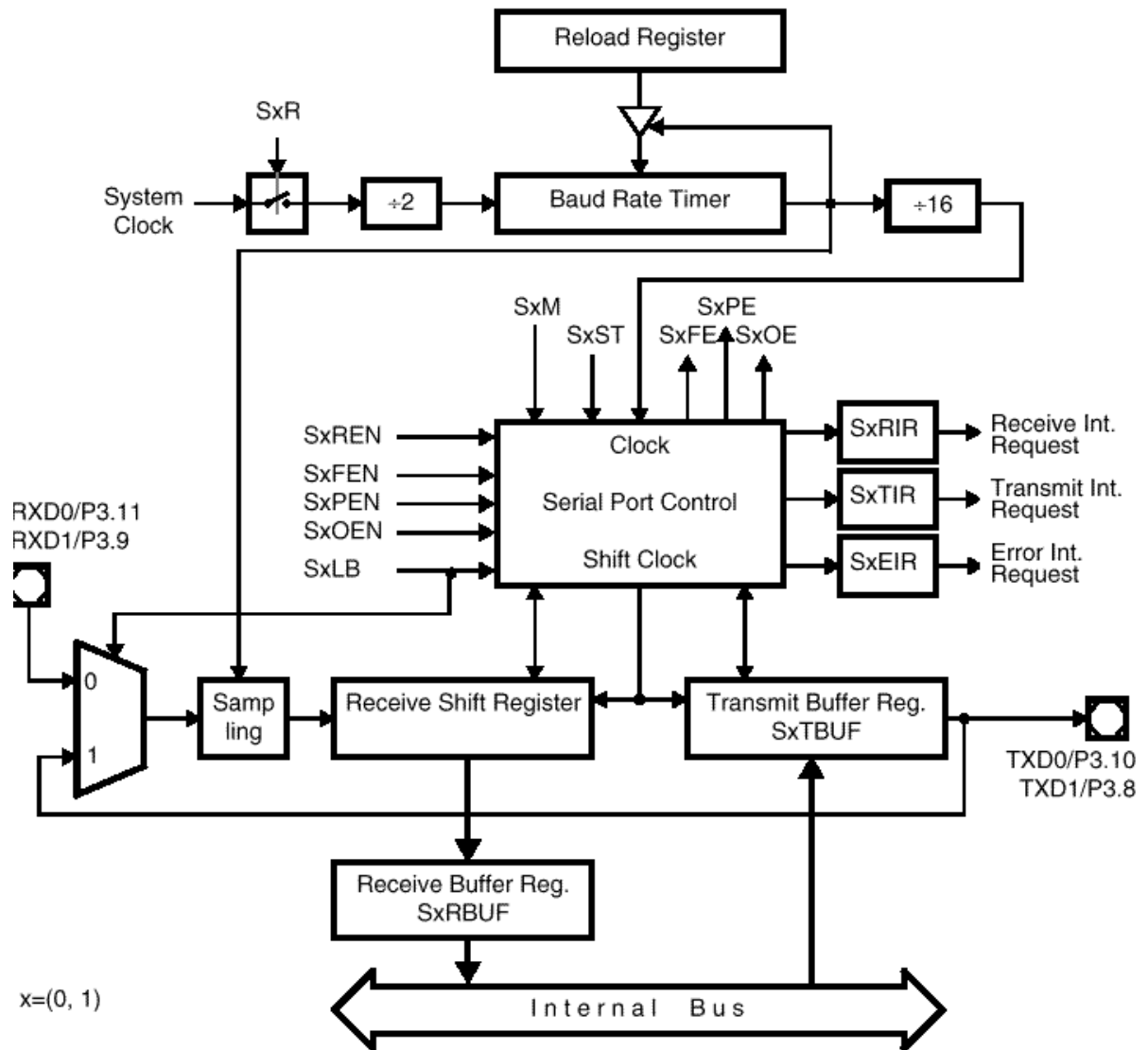
Odbiór danych jest możliwy gdy bit S0REN (S1REN) jest ustawiony (oraz włączony jest zegar) . O przyjściu znaku informuje ustawienie bitu S0RIR (S1RIR), który może wywoływać przerwanie. Bit ten musi być skasowany programowo przed przyjęciem następnego znaku. Bufor odbiorczy jest buforem podwójnym, możliwe jest więc odbieranie kolejnego znaku przed odczytaniem pierwszego. We wszystkich trybach informacje o błędach w transmisji są dostępne w odpowiednich bitach rejestru S0CON (S1CON). Dodatkowo jeśli ustawiony jest bit S0EIR (S1EIR) to przy wystąpieniu jakiegokolwiek błędu zostanie wygenerowane przerwanie. W trybie

pracy z bitem pobudzenia przyjmowane są tylko dane, w których ostatni - najstarszy bit ma wartość '1'. Jest to tryb wykorzystywany przy komunikacji wieloprocesorowej.

□ **Transmisja asynchroniczna.**

Transmisji asynchroniczna jest w pełni dwukierunkowa. Każdy z dwóch portów transmisji szeregowej zajmuje końcówki odpowiednio RXD0/P3.11 i TXD0/P3.10 dla portu ASC0 oraz RXD1/P3.9 i TXD1/P3.8 dla portu ASC1. Rysunek 30 pokazuje schemat blokowy jednego portu szeregowego.

UWAGA!!! Aby umożliwić działanie portów szeregowych należy ustawić odpowiednie piny jako wejścia i wyjścia. Piny RXD0/P3.11 i RXD1/P3.9 muszą być ustawione jako wejścia w odpowiednim rejestrze kontrolnym, bit DP3.11 i DP3.9 musi być ustawiony w stan zera logicznego. Natomiast piny TXD0/P3.10 i TXD1/P3.8 muszą być ustawione jako wyjścia. W rejestrze sterującym portem P3 należy ustawić bity DP3.10 i DP3.8 .

Rysunek 30. Schemat blokowy portu szeregowego.

□ 8 Bitowe dane.

Tryb ten jest wybierany przez wpisanie do słowa S0M (S1M) w rejestrze S0CON (S1CON) wartości '001b'. Dane będą nadawane i / lub odbierane jako słowa 8-mio bitowe. Po odebraniu danej starszy bajt rejestru S0RBUF ma wpisaną wartość '0', a młodsza część odebraną daną. Kontrola parzystości w tym trybie jest wyłączona niezależnie od ustawienia bitów S0PEN (S1PEN).

□ **7-bitowe dane + bit parzystości.**

Aby ustawić ten tryb należy wpisać do słowa S0M (S1M) wartość '011b'. Dane nadawane i / lub odbierane będą posiadały siedem bitów danych i jeden bit parzystości. W trybie tym dostępne są wszystkie kontrole błędów. W czasie transmisji bit parzystości jest automatycznie generowany i ustawiany na najmłodszej pozycji w ramce (ale przed bitem / bitami stopu). Bit parzystości jest ustawiony jeśli suma modulo 2 siedmiu bitów danych jest '1' i kasowany jeśli jest '0'. W czasie odbioru obliczana jest parzystość 7-bitów danych i porównywana z 8-mym bitem parzystości. Jeśli włączona jest kontrola parzystości ustawiany jest znacznik błędu i znacznik przerwania S0EIE (S1EIE) dla odpowiedniego kanału. Po odebraniu danej starszy bajt bufora odbioru S0RBUF (S1RBUF) jest ustawiany w stan '0', bit D7 jest bitem parzystości, a pozostałe D0-D6 są bitami danych.

□ **9-bitowe dane.**

Tryb ten jest ustawiany przez wpisanie do słowa S0M (S1M) wartości '100b'. Dane do transmisji umieszcza się na najmłodszych 9-ciu bitach rejestru S1TBUF (S1TBUF). W czasie odbioru wszystkie 9 - bitów zostaje wpisane do rejestru S0RBUF (S1RBUF), pozostałe nie używane bity są kasowane. Kontrola parzystości w tym trybie jest niedostępna niezależnie od ustawienia bitów S0PEN (S1PEN). Błąd przepelnienia oraz kontrola ramki mogą być ustawione.

□ **8-bitowe dane + bit pobudzenia.**

Jest to specjalny tryb przeznaczony specjalnie do komunikacji z innymi procesorami. Ustawia się go poprzez wpisanie słowa '101b' do S0M (S1M). Nadawanie jest identyczne jak w poprzednim trybie z 9-bitowymi danymi. Odbiór jest możliwy tylko wtedy, gdy najstarszy bit – dziewiąty jest w stanie '1', jeśli jest on w stanie '0' to dana nie jest przepisywana do bufora odbiorczego, nie są też ustawiane bity odpowiedzialne za odbiór. Kiedy główny procesor zamierza przesłać dane do procesora podrzędnego, wysyła najpierw jego adres (numer, identyfikator). Adres różni się od danych ustawionym 9-tym bitem w stan '1', podczas gdy dane mają 9-ty bit ustawiony w stan '0'. Wszystkie procesory podrzędne odczytują ten adres i porównują ze swoim, wybrany procesor przełącza się na tryb z 9-bitowymi danymi (poprzez

skasowanie bitu SxM.0) i może odczytywać dane. Procesory, które nie zostały wybrane przesyłanych danych wcale nie zauważają.

□ **8-bitowe dane + bit parzystości.**

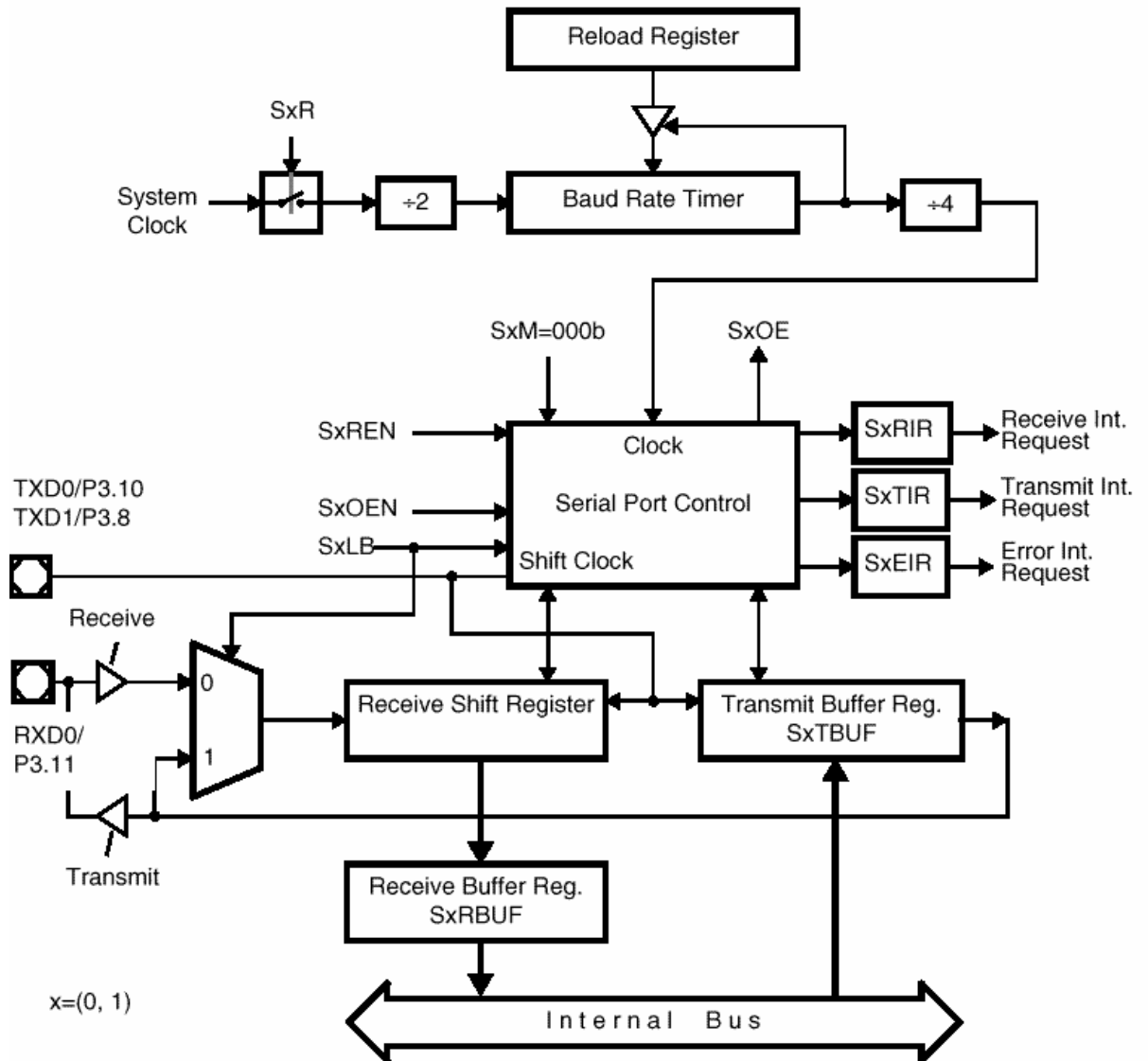
Tryb ten jest ustawiany przez wpisanie słowa '111' do S0M lub S1M. Dane w ramce są nadawane i / lub obierane jako 8-bitowe z bitem parzystości. W trybie tym dostępne są wszystkie kontrole błędów. W czasie transmisji bit parzystości jest automatycznie generowany na podstawie 8-miu bitów danych i wstawiany jako najstarszy bit w ramce. W czasie odbioru bit parzystości jest automatycznie generowany (z pierwszych ośmiu bitów) i porównywany z 9-tym bitem odebranego słowa. Jeśli porównanie się nie powiedzie ustawiany jest znacznik błędu oraz znacznik przerwania SxEIE.

□ **Transmisja synchroniczna.**

Ten tryb pracy portu szeregowego charakteryzuje się pół- dwukierunkowym kierunkiem transmisji i przeznaczony jest dla prostych urządzeń wejścia / wyjścia posiadających rejestr przesuwany. Osiem bitów danych jest wysyłanych bądź odbieranych w takt zegara synchronizującego. Sygnał z zegara synchronizującego jest obecny tylko w czasie przesyłania danych. Transmisję synchroniczną można ustawić wpisując do S0M (S1M) wartość '000b'.

Rysunek 31 pokazuje schemat blokowy portu szeregowego synchronicznego.

Rysunek 31. Schemat blokowy portu szeregowego synchronicznego.



W trakcie transmisji szeregowej pin TXD0/P3.10 jest używany jako wyjście zegarowe (DP3.10 musi być ustawiony w stan '1'), a pin RXD0 służy do transmisji zarówno zapisu jak i odczytu dla portu ASC0. Analogiczne ustawienia dla portu ASC1.

□ **Wysyłanie w trybie synchronicznym.**

Aby wysłać bajt należy załadować go do bufora nadawczego S0TBUF. Jeśli ustawimy bity S0R=1 (S1R=1) i S0REL=0 (S1REL=0) to po czterech cyklach zegarowych od tej operacji zapisu zostanie rozpoczęte nadawanie. Dana jest wysyłana wraz z taktami zegarowymi. Po wysłaniu 8-mego - ostatniego bitu ustawiony zostaje znacznik transmisji S0TIR (S1TIR) i następuje zakończenie wysyłania.

Jeśli w czasie wysyłania danej wpisujemy do bufora kolejną wartość, to poprzednie wysyłanie zostanie przerwane i wysłana zostanie nowa wartość bufora. Ustawienie bitu S0REN (S1REN) w trakcie wysyłania może prowadzić do błędów. Dla tej konfiguracji TXD0/P3.10 i TXD1/P3.8 są wyjściami sygnału zegarowego i należy ustawić te piny jako wyjścia (DP3.10 i DP3.8 = 1). Piny RXD0/P3.11 (RXD1/P3.9) też są wyjściami, a więc należy je także ustawić jako wyjścia (DP3.11 i DP3.9 = 1).

□ **Odbiór w trybie synchronicznym.**

Odbiór jest inicjowany przez ustawienie bitu S0REN=1 (S1REN=1). Jeśli generator jest włączony (S0R=1 ;S1R=1) to poprzez pin RXD0 (RXD1) wczytywany jest bajt w takt zegara taktującego wyprowadzonego na pin TXD0 (TXD1). Po pobraniu wszystkich ośmiu bitów dana zostaje przepisana do bufora odbiorczego oraz zostaje ustawiony znacznik przerwania S0RIR (S1RIR). Natomiast bit zezwalający na odbiór zostaje skasowany S0REN=0 (S1REN=0). Końcówki RXD0/P3.11 i RXD1/P3.9 muszą być skonfigurowane jako wejścia (DP3.11 = 0 i DP3.9 = 0).

Raz zapoczątkowany odbiór nie może być w żaden sposób zatrzymany, ani przez ustawienie SxREN w stan '0', ani przez wpisanie danej do bufora nadawczego.

W transmisji synchronicznej młodszy bajt bufora odbiorczego reprezentuje daną, podczas gdy starszy jest zawsze równy zero. Jeśli bufor odbiorczy nie został odczytany do czasu przyścia następnego znaku, pojawi się błąd przepełnienia S0OE (S1OE), i ustawiony zostanie znacznik przerwania S0EIR (S1EIR).

□ Tryb pętli zamkniętej.

W celu testowania został wprowadzony specjalny tryb, który umożliwia testowanie każdego z porów bez używania odpowiadających mu pinów. Dane są przepisywane bezpośrednio z bufora nadawczego do odbiorczego. Transmisje w pętli zamkniętej wywołuje się ustawiając bity: S0LB=1 , S0REN=1 (S1LB=1,S1REN=1), zarówno dla transmisji synchronicznej jak i asynchronicznej.

□ Generator.

Każdy z porów szeregowych posiada swój własny generator z własnym 13-to bitowym licznikiem. Oba liczniki są taktowane przez zegar systemowy podzielony przez 2. Timery te mogą być wystartowane i zatrzymane przez bity S0R i S1R w rejestrze S0CON i S1CON. Każde przepełnienie licznika powoduje jeden takt dla portu szeregowego. Każdy z liczników jest automatycznie przeładowywany przez własny 13-bitowy rejestr przeładowania. Rejestry S0BG i S1BG posiadają podwójne funkcje. Odczytując je otrzymujemy wartość Timera, a zapisując przeładowujemy nową wartością. Skasowanie bitów S0R, S1R powoduje zablokowanie przeładowywania, natomiast ich ustawienie powoduje pierwsze przeładowanie dopiero po czasie jednego cyklu rozkazowego.

Podczas transmisji asynchronicznej częstotliwość zegara określa się na podstawie wzoru:

$$B_{asyc0} = \frac{f_{osc}}{64 * (<S0BRL> + 1)} \qquad B_{asyc1} = \frac{f_{osc}}{64 * (<S1BRL> + 1)}$$

gdzie <S0BRL> i <S1BRL> reprezentują 13-bitową wartość przeładowyującą licznik, a f_{osc} – częstotliwość zegara systemowego.

Maksymalną możliwą do otrzymania prędkością transmisji asynchronicznej jest 625 Kbodów. Tabela poniżej pokazuje wartości rejestrów S0BRL i S1BRL dla standardowych prędkości transmisji.

<i>Prędkość transmisji</i>	<i>Częstotliwość zegara</i>	<i>Wartość przeladowywana</i>
652 000	40 MHz	0000h
19 200	39,3216 MHz	001Fh
9 600	39,3216 MHz	003Fh
4 800	39,3216 MHz	007Fh
2 400	39,3216 MHz	00FFh
1 200	39,3216 MHz	01FFh
600	39,3216 MHz	03FFh
75	39,3216 MHz	1FFFh

Dla transmisji synchronicznej wzór jest podobny, z tym że zegar systemowy jest dzielony przez 16 a nie przez 64.

Maksymalna prędkość przy transmisji synchronicznej wynosi 2.5 MBodów/s (dla zegara 40MHz)

□ **System przerwania portów szeregowych.**

Do każdego z portów szeregowych przypisane są trzy adresowane bitowo rejestry kontroli przerwania. Rejestry S0TIC i S1TIC są odpowiedzialne za przerwanie przy nadawaniu, rejestry S0RIC i S1RIC za przerwanie przy odbiorze, natomiast rejestry S0EIC i S1EIC za przerwanie wywołane błędami w transmisji. Każde przerwanie ma swój własny dedykowany wektor przerwania oznaczany jako S0TINT, S1TINT dla przerwania nadawania, S0RINT, S1RINT dla odbioru i E0EINT, E1EINT dla wektora przerwania spowodowanego błędem.

Przerwanie spowodowane błędem transmisji nie mówi o tym jaki błąd wystąpił. Aby się tego dowiedzieć należy sprawdzić odpowiednie bity w słowie sterującym portem szeregowym S0CON (S1CON) – bity (S0FE, S0PE, S0OE lub S1FE, S1PE, S0OE). Każdy z wymienionych bitów musi być skasowany programowo przed dalszą transmisją.

7	6	5	4	3	2	1	0
S0TIR	S0TIE		ILVL		GLVL		

S0TIR – znacznik przyjęcia przerwania

S0TIE – maskowanie przerwania (1-przerwanie aktywne, 0-przerwanie nieaktywne)

ILVL - priorytet przerwania (0-15, 0-przerwanie się nie wykonuje)

GLVL - priorytet grupy

Pozostałe rejestry przerwania są analogiczne.