

Mikrokontrolery serii PIC16C5x

O mikrokontrolerach PIC firmy Microchip pisaliśmy w rubryce „Wizytówka miesiąca” (EP 9/94).

Również Elektor Elektronik zainteresował się tymi bardzo wydajnymi i wdzięcznymi do programowania mikroprocesorami.

Układy te są warte szerszego upowszechnienia z uwagi na ich możliwości, cenę oraz - co najważniejsze - ich podstawowy obszar aplikacji: drobne, o małej liczbie funkcji szybkie sterowniki.

Równoległe z Elektorem także i w EP rozpoczynamy publikacje o tych układach.



Mikrokontrolery serii PIC16C5X są jedną z trzech linii rozwojowych rodziny PIC wytwarzanej przez firmę Arizona Microchip Technology Inc. W skład tej linii wchodzi pięć typów kontrolerów:

- PIC16C54,
- PIC16C56,
- PIC16C58,
- PIC16C55,
- PIC16C57.

Nie bez powodu wymieniono te typy w takiej właśnie kolejności. Procesory te bowiem grupami są zamykane w tych samych obu-

dowach i mogą być wymiennie stosowane w niektórych układach. Układ wyprowadzeń jest identyczny w poszczególnych grupach (rys. 1). Jeśli jeszcze przyjrzymy się wielkości pamięci EPROM, to okaże się, że ze względu na numerację są one kompatybilne w dół (tabela 1), czyli program, który jest stworzony dla wariantu uboższego będzie pracował w procesorze bogatszym w pamięć.

Mikroprocesory wykonywane są w technologii CMOS o wymiarze charakterystycznym 0.9um. Są one w pełni statyczne, co oznacza, że pracują z zegarem o dowolnej częstotliwości nie większej od granicznej i mogą być zasilane napięciem z przedziału od 3V do 6.5V. Ze względu na wielkość częstotliwości granicznej i rodzaj oscylatora wyróżnia się w każdym typie podstawowym cztery odmiany:

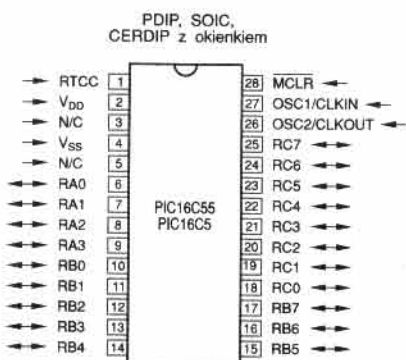
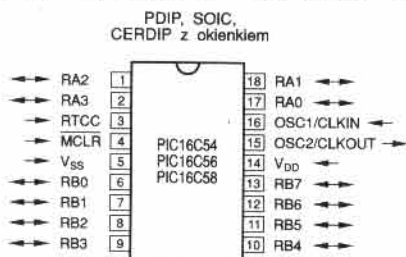
- LP - odmiana o małym poborze mocy, oscylator kwarcowy o częstotliwości do 200kHz;
- RC - odmiana z oscylatorem RC o częstotliwości do ok. 4MHz;
- XT - odmiana z oscylatorem kwarcowym lub ceramicznym o częstotliwości do 4MHz;
- HS - odmiana z oscylatorem kwarcowym lub ceramicznym o częstotliwości do 20MHz.

Wybór odmiany jest programowany dla układów zamykanych w obudowach do wielokrotnego kasowania (z okienkiem), a wersje OTP mają już ustawioną ją na stałe, co jest ukazane na dalszych pozycjach oznaczenia produktu. Odmiana oscylatora i obudowa silnie wpływają na cenę kostki.

Oprócz wyżej wymienionych typów podstawowych produkowane są również typy ze zredukowanym poborem prądu (poniżej 1uA przy 3V zasilania, w trybie uśpienia i wyłączonym układem watchdoga) oznakowane dodatkową literą A, zaś wśród nich wyodrębnia się jeszcze wersję z rozszerzonym zakresem napięć oznaczone PIC16LC5XA. Ponadto w celu wyróżnienia odmiany z pamięcią ROM programowaną przez producenta wprowadza się do oznaczenia literę R, np. PIC16(L)CR57(A).

Architektura

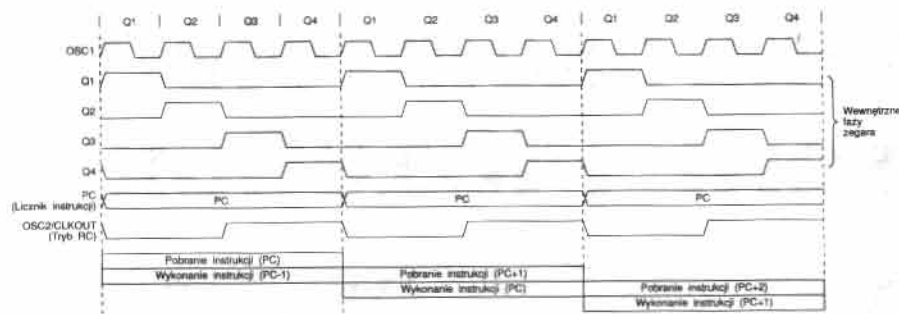
Na podstawie cytowanej tabeli łatwo zauważyć, że długość słowa programu jest inna niż długość słowa danych. Wynika stąd wniosek, że pamięć programu nie może mieć wspólnych obszarów adresowych z pamięcią danych. Istotnie, kontrolery całej rodziny PIC mają mało znaną u nas architekturę hardware'ową, która, w odróżnieniu od architektury automatu von Neumanna (do takiej jes-



Tab. 1. Cechy charakterystyczne procesorów rodziny PIC16C5x

μC	EPROM	RAM	I/O
PIC16C54	512x12	32x8	12
PIC16C56	1024x12	32x8	12
PIC16C58	2048x12	80x8	12
PIC16C55	512x12	32x8	20
PIC16C57	2048x12	80x8	20

Rys. 1. Obudowa i wyprowadzenia procesorów PIC16C5x



Rys. 2. Sposób pobierania i wykonywania rozkazów przez procesory PIC

teśmy przyzwyczajeni przez produkty Intela), mają rozdzielone pamięci programu i danych. Zaletą architektury harwardzkiej jest niezależne kształtowanie długości słowa programu od przeważnie sztywnego słowa danych. Sztywność słowa danych jest związana ze standardami panującymi na świecie, skąd wynika fakt, że nikt nie buduje mikroprocesorów np. 11-bitowych.

Zaletą architektury harward jest to, że nie tracimy możliwości różnego adresowania pamięci danych, możemy natomiast zyskać na skróceniu długości programu, bowiem w wydłużonym słowie (trudno mówić, że to jest bajt) mieści się pełen bajt danej natychmiastowej i jeszcze zostaje kilka bitów na kod rozkazu. Jeśli długość słowa jest niewystarczająca można dolożyć kolejny jeden lub dwa bity. Taka jest właśnie różnica pomiędzy rodziną PIC16C5X i rodziną PIC16CXX. Dodatkowo zyskujemy też na szybkości pracy procesora, bo słowo jest pobierane w jednym cyklu maszynowym, a w następnym cyklu jest ono wykonywane.

W mikrokontrolerach PIC proces pobierania i dekodowania rozkazu oraz wykonywania rozkazu poprzedniego jest realizowany w jednym cyklu. Przypomina to przetwarzanie potokowe spotykane w dużych procesorach (ang. pipelining) - rys. 2. Wyjątek od zasady przetwarzania potokowego stanowią tu rozkazy skoków i odwołania do podprogramu, które z natury rzeczy muszą być dwucyklowe, ponieważ na etapie pobierania instrukcji nie wiadomo jeszcze, jaki będzie wynik operacji jej dekodowania. Producent używa tu terminu „procesor RISCopodobny”. Termin ten właściwie oddaje fakt wykonywania rozkazu w jednym cyklu maszynowym oraz krótką listę dostępnych rozkazów.

Na rysunku 3 pokazano architekturę wewnętrzną mikroprocesorów PIC16C5X. Struktura przedstawiona na rysunku jest uniwersalna, uwzględnia różnice pomiędzy poszczególnymi typami.

Wyróżniamy następujące elementy architektury:

- pamięć programu,
- pamięć danych jako zbiór rejestrów,
- układ watchdog timera i timera RTCC z preskalerem,
- jednostkę arytmetyczno-logiczną ALU z rejestrem roboczym W,
- układy portów we/wy,
- układ oscylatora.

Pamięć programu składa się ze zmiennej, zależnej od numeru procesora liczby 12-bitowych słów.

Zbiór rejestrów to pamięć RAM oraz re-

jestry, których zawartość jest zmieniana samymi rozkazami procesora. Przyjęto oznaczenie adresowalnej pamięci przez podawanie litery f (ang. file - zbiór) przed adresem, czyli np. komórka o adresie 03h ma oznaczenie f03. Generalnie nie używa się liczb innych niż szesnastkowe, choć zapis dziesiętny jest dopuszczalny przez assembler.

Zbiór rejestrów jest podzielony na trzy grupy:

- rejestry operacyjne,
- rejestry portów,
- rejestry ogólne.

Rejestry operacyjne to :

- wskaźnik adresowania pośredniego INDF,
- rejestr adresowania pośredniego FSR,
- rejestr timera RTCC,
- licznik rozkazów PC,
- rejestr stosu STACK,
- rejestr stanu STATUS.

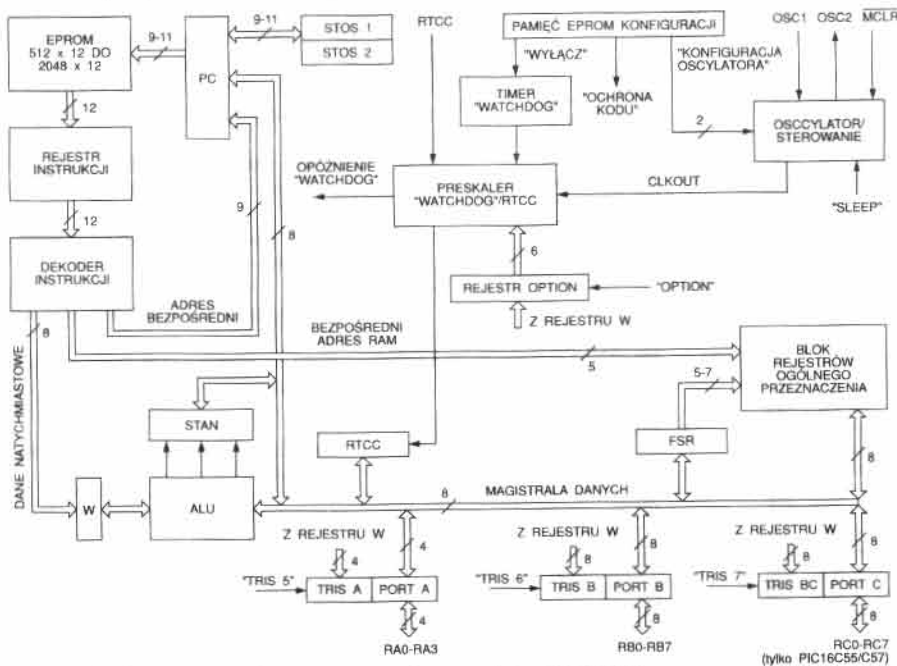
Wskaźnik adresowania pośredniego INDF (Indirect Data Adressing) fizycznie nie istnieje. Jest to tylko termin występujący w grupie rozkazów adresowania pośredniego wskazujący na źródło adresu, którym jest zawsze zawartość rejestru adresowania pośredniego FSR. Na przykład operacja ADDWF INDF,W spowoduje dodanie do zawartości rejestru W zawartości rejestru zaadresowanego zawartością rejestru FSR i umieszczenie wyniku w rejestrze W. Pomimo specyfiki rejestru INDF daje się on zaadresować (adres f00),

czyli jest komórką pamięci RAM. Pośredni odczyt tego rejestru (czyli dla FSR=00) daje wartość zero, natomiast pośredni zapis do niego wykonuje się jako rozkaz NOP, co świadczy o jego fizycznym nieistnieniu.

Rejestr FSR wchodzi w skład pamięci RAM i ma adres f04. Służy on do pośredniego adresowania pamięci RAM.

Licznik rozkazów PC jest fragmentem zbioru rejestrów z tą różnicą, że jako jedyny z nich ma długość większą niż 8 bitów, zależnie od tego, jaką pojemność ma pamięć EPROM. Bit A9 i dalsze są umieszczone w rejestrze stanu STATUS. Pamięć programu została podzielona na 512-sładowe strony. Pamięć procesorów 16C54/55 mieści się w jednej stronie. Skoki bezwarunkowe GOTO mogą odbywać się w ramach jednej strony. W rozkazie umieszczanych jest 9 bitów, czyli również nie pamiętany nigdzie A8. Odwołania do podprogramów CALL odbywają się zawsze w ramach dolnej połowy strony. Bit A8 jest wtedy zawsze wyzerowany. Bit ten będzie wyzerowany także w przypadku modyfikacji przez program licznika rozkazów PC. Wynika z tego, że ta połowa strony jest przeznaczona do umieszczania podprogramów, zaś w górnej połowie powinien znaleźć się program główny. Temu podziałowi podporządkowano zachowanie się licznika rozkazów podczas restartu, który ma wtedy ustawione wszystkie swoje bity, co oznacza, że pierwszym wykonanym rozkazem będzie kod zawarty w ostatnim słowie pamięci programu. Tym rozkazem jest zazwyczaj rozkaz skoku bezwarunkowego.

Rodzina PIC16C5X posiada dwa rejestry stosu o długości takiej samej, jak licznik rozkazów PC. Rejestry te, o nazwach STACK1 i STACK2, służą wyłącznie do pamiętania adresu powrotu z podprogramu i nie są fragmentem pamięci RAM. Producent przewidział więc tylko dwukrotnie zagnieżdżanie procedur. Taka ilość wydaje się wystarczająca, jeśli uświadomimy sobie, że żaden z tych procesorów nie może pracować z zewnętrzną pamięcią programu, zaś wielkość



Rys. 3. Budowa wewnętrzna procesorów z rodziny PIC16C5x

pamięci wewnętrznej nie jest imponująca.

Wywołanie podprogramu CALL powoduje zapamiętanie w rejestrze STACK1 adresu powrotu, przedtem poprzednia zawartość STACK1 jest przepisywana do STACK2. W razie trzykrotnego odwołania się do podprogramu adres powrotu z pierwszej procedury ulegnie wymazaniu. Na szczęście producent przewidział taką możliwość w firmowym symulatorze tych procesorów, który w takiej sytuacji zgłosi błąd. Rozkaz powrotu z podprogramu RETLW działa odwrotnie: przepisuje ze STACK1 do PC, a STACK2 do STACK1. Trzecie wykonanie RETLW spowoduje skok do adresu ze STACK2.

Rejestr stanu STATUS jest jedną z komórek pamięci RAM o adresie f03. Jest to zbiór jednobitowych znaczników od najmłodszego licząc:

- C - wskaźnik przeniesienia/pożyczki;
- DC - wskaźnik przeniesienia pomiędzy tetradami;
- Z - wskaźnik wyniku zerowego operacji;
- PD - wskaźnik stanu obniżonego poboru mocy;
- TO - wskaźnik faktu zerowania procesora przez układ watchdoga;
- PA0, PA1 - bity wyboru strony pamięci programu;
- PA2 - na razie jest to bit dla użytkownika, do zastosowań specjalnych w kolejnych wersjach.

Zestaw rejestrów każdego portu składa się z dwóch rejestrów:

- rejestru PORTx, który jest fragmentem pamięci RAM,
- rejestru sterującego TRISx, gdzie x jest oznaczeniem portu.

Rejestry TRIS odpowiadają za kierunek przesyłania danych przez konkretną linię portu. Stan wysoki odpowiedniego bitu tego rejestru oznacza linię wejściową, zaś niski - wyjściową. Dane wyjściowe na liniach są zatraskiwane, zaś dane wejściowe - tylko buforowane. Rejestry TRIS są rejestrkami tylko do zapisu, nie stanowią fragmentu pamięci RAM, zapis polega na przesłaniu danych z rejestru W za pomocą rozkazu TRIS.

Port A jest 4-bitowy i służy przede wszystkim jako zestaw swobodnych linii we-wy. Port B jest ośmiobitowy. Rejestr portu C w procesorach o numerze parzystym jest rejestrem ogólnym, zaś TRISC nie występuje, ponieważ port C nie istnieje.

Rejestry ogólne to pozostałe komórki pamięci RAM, przeznaczone do przechowywania danych przez użytkownika. Wielkość pamięci RAM zależy od typu procesora. Rejestry o adresie od f01 do f1F włącznie mogą być adresowane bezpośrednio i pośrednio, pozostałe zaś tylko pośrednio. PIC16C5X mają 32 bajty pamięci RAM adresowanej w całości bezpośrednio i pośrednio. W PIC16C57/58 występuje stronicowanie pamięci. W tych procesorach pozostała pamięć RAM jest adresowana tylko pośrednio (za pomocą rejestru FSR). O dostępie do danego banku pamięci decydują dwa bity rejestru FSR (piąty i szósty). Aby mieć dostęp do banków w ogóle, należy ustawić czwarty bit tego rejestru, w przeciwnym razie zawsze zaadresujemy pośrednio najmłodsze 16 bajtów pamięci RAM (tej, w której znajdują się m.in. rejestry operacyjne), niezależnie od stanu bitów następnym. Starsze, nieużywane bity rejestru

Tab. 2. Lista rozkazów procesorów PIC16C5x

Kod inst.	Opis	Mnemonic	Symboliczny zapis operacji	Wpływ na wskaźn.
Rozkazy przetwarzania danych bajtowych				
000111dffff	Dodaj W do f	ADDWF f,d	W+f->d	C,DC,Z
000101dffff	Iloczyn logiczny W i f	ANDWF f,d	W .AND. f->d	Z
0000011ffff	Zeruj f	CLRF f	0->f	Z
000001000000	Zeruj W	CLRWF	0->W	Z
001001dffff	Inwersja f	COMF f,d	!f->d	Z
000011dffff	Zmniejsz f o 1	DECF f,d	f-1->d	Z
001011dffff	Zmniejsz f o 1; omiń, gdy d=0	DECFSZ f,d	f-1->d;	-
001010dffff	Zwiększ f o 1	INCF f,d	f+1->d	Z
001111dffff	Zwiększ f o 1; omiń, gdy d=0	INCFSZ f,d	f+1->d;	-
000100dffff	Suma logiczna w i f	IORWF f,d	W .OR. f->d	Z
001000dffff	Prześlij f	MOVF f,d	f->d	Z
0000001ffff	Prześlij W do f	MOVWF f	W->d	-
000000000000	Nic nie rób	NOP	-	-
001101dffff	Przesuń cyklicznie w lewo razem z C	RLF f,d	f(n)->d(n+1); C->d(0); f(7)->C	C
001100dffff	Przesuń cyklicznie w prawo razem z C	RRF f,d	f(n)->d(n-1); C->d(7); f(0)->C	C
000010dffff	Od f odejmij W	SUBWF f,d	f-W->d	C,DC,Z
001110dffff	Zamień tetrady miejscami	SWAP f,d	f<0:3><->f<4:7>->d	-
000110dffff	Exclusiv OR W i f	XORWF f,d	W .XOR. f->d	Z
Rozkazy działań na bitach				
0100bbffff	Zeruj bit w f	BCF f,b	0->f(b)	-
0101bbffff	Ustaw bit w f	BSF f,b	1->f(b)	-
0110bbffff	Testuj bit w f; BTFSC f,b omiń, gdy zerowy	BTFSC f,b	f(b)=0 =>PC<-PC+2 f(b)=1 =>PC<-PC+1	-
0111bbffff	Testuj bit w f; omiń, gdy ustawiony	BTFSS f,b	f(b)=1 =>PC<-PC+2 f(b)=0 =>PC<-PC+1	-
Pozostałe rozkazy				
1110kkkkkkkk	Iloczyn logiczny W i litera ^u	ANDLW k	W .AND. k -> W	Z
1001kkkkkkkk	Skok do podprogramu	CALL k	PC+1->STACK; k->P	-
00000000100	Zeruj watchdog	CLRWDT	0->WDT	TO,PD
101kkkkkkkk	Skok bezwarunkowy	GOTO k	k->PC<0:9>	-
1101kkkkkkkk	Suma logiczna W i litera ^u	IORLW k	W .OR. k -> W	Z
00000000010	Ładuj rejestr OPTION	OPTION	W -> OPTION	-
1000kkkkkkkk	Powrót z podprogramu	RETLW k	k->W; STACK->PC	-
00000000011	Przejdźcie w stan uśpienia	SLEEP	0->WDT; CLK=0	TO,PD
00000000fff	Ładowanie rejestru TRISf	TRIS f	W->TRISf	-
1111kkkkkkkk	Exclusiv OR W i litera ^u	XORLW k	W .XOR. k -> W	Z

FSR mogą być tylko odczytane i ich wartość wynosi 1. W dodatkowych bankach pamięci istnieje tylko tylko 16 starszych komórek RAM, młodsze są fizycznie komórkami z pierwszego banku.

Układ watchdoga jest licznikiem i służy do odnajdowania zagubionej ścieżki logicznej programu w warunkach pojawienia się dużych zakłóceń w otoczeniu procesora. Jego

działanie polega na okresowym generowaniu restartu jednostki centralnej w momencie przepelnienia tego licznika. Unikając takiego zdarzenia można poprzez dodanie do programu rozkazu zerowania watchdoga. Program z zagubioną ścieżką logiczną może nie znaleźć na czas tego rozkazu. Watchdog ma własny oscylator RC, który jako mało stabilny zapewnia przepelnienie co około 18ms.