ZESZYTY NAUKOWE POLITECHNIKI ŚLĄSKIEJ 1983

Seria: INFORMATYKA z. 5 Nr kol. 759

Adam GACEK
Stanisław KOZIELSKI
Piotr STRÓŻYNA

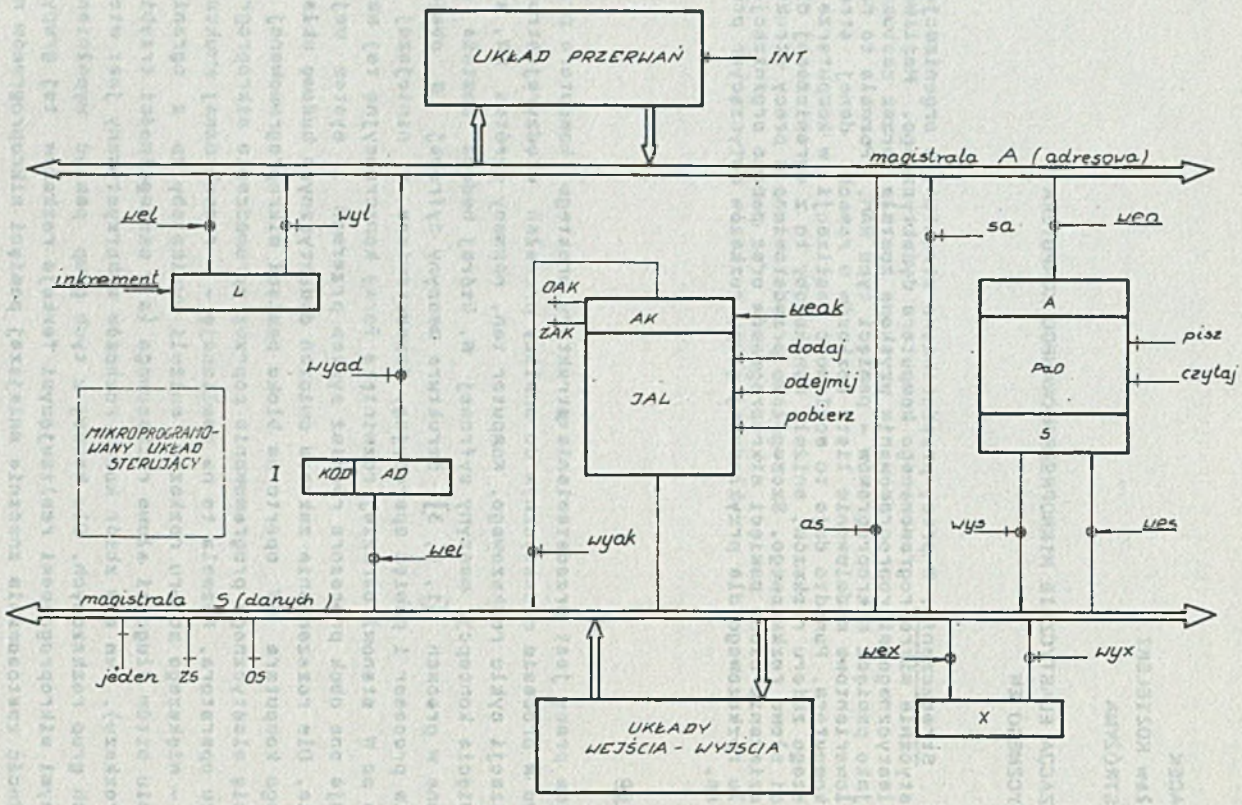
ORGANIZACJA ELASTYCZNIE MIKROPROGRAMOWANEGO KOMPUTERA DYDAKTYCZNEGO EW

Streszczenie. W pracy przedstawiono strukturę i organizację elastycznie mikroprogramowanego komputera dydaktycznego. Możliwość elastycznego mikroprogramowania uzyskana została przez zastosowanie jako pamięci mikroprogramów - pamięci typu RAM. Pozwala to na wielowariantowe modelowanie listy rozkazów w ramach danej struktury komputera. Ponadto daje to możliwość realizacji w komputerze większego zbioru rozkazów, aniżeli wynikałoby to z ograniczonej długości słowa rozkazowego. Szczegółowo przedstawiono w pracy strukturę i działanie bloku pamięci mikroprogramów oraz podano organizację cyklu rozkazowego dla przykładowej grupy rozkazów dotyczących procesora.

1. Wstęp

Celem pracy jest przedstawienie struktury prostego komputera przeznaczonego w procesie dydaktycznym do analizy przesłań międzyrejestrowych i organizacji cyklu rozkazowego. Komputer ten, nazwany skrótem EW, stanowi rozwinięcie koncepcji maszyny cyfrowej W, której budowa została przedstawiona w pracach [1, 2, 3]. Struktura maszyny cyfrowej W obejmowała jedynie procesor i pamięć operacyjną. Przedstawiona w niniejszej pracy wersja mc W stanowi bardziej rozwiniętą formę konstrukcyjną tej maszyny. Obejmuje ona obok procesora również system przerwań i system wejście - wyjścia. Dla rozszerzenia zakresu ćwiczeń dydaktycznych budowę układu sterującego komputera EW oparto na bloku pamięci mikroprogramowanej z możliwością elastycznego programowania poprzez wprowadzanie mikroprogramów z pulpitu operatora. Pozwala to na realizację - w ramach danej struktury maszyny - większego zbioru rozkazów, aniżeli wynikałoby to z ograniczonej do ośmiu bitów długości słowa rozkazowego (a w szczególności trzybitowego kodu rozkazu). Ten sam zbiór kodów rozkazów wykorzystywany jest wtedy dla różnych grup rozkazowych. Dla każdej z tych grup pamięć wypełniana jest odrębnymi mikroprogramami realizującymi funkcje rozkazów tej grupy. Daje to możność zastosowania znacznie mniejszej pamięci mikroprogramów niż wynikałoby to z potrzeb jednorazowej realizacji pełnej listy rozkazów.

Przyjęta koncepcja mc EW odpowiada również wymaganiom związanym z organizacją zajęć dydaktycznych. Podział zajęć na jednostki dydaktyczne na-



Rys. 1. Architektura jednostki centralnej EW

rzuca bowiem konieczność oddzielnej analizy poszczególnych bloków funkcjonalnych, a więc również oddzielnego rozpatrywania odpowiednich grup rozkazów dotyczących tych bloków.

W niniejszej pracy skupiono uwagę na strukturze samego procesora mc EW oraz organizacji cyklu rozkazowego dla przykładowej grupy rozkazów dotyczących procesora. Szczegółowo została przy tym przedstawiona struktura i działanie bloku pamięci mikroprogramów decydującego o własnościach i walorach dydaktycznych komputera EW.

2. Struktura maszyny cyfrowej "EW"

2.1. Architektura jednostki centralnej

Architektura jednostki centralnej komputera "EW" przedstawiona została na rys. 1.

Przesyły międzyrejestrowe kontrolowane są przez sygnały mikrosterujące. Sygnały te sterują również pracą jednostki arytmetyczno-logicznej, pamięci operacyjnej oraz pozostałych bloków funkcjonalnych maszyny. Na rys. 1 zaznaczono ważniejsze sygnały mikrosterujące rozróżniając sygnały impulsowe (podkreślone) i statyczne.

Magistrala danych mc EW ma 8 bitów (jest to długość słowa maszynowego). Magistrala adresowa jest 5-bitowa, pozwala to na bezpośrednie adresowanie 32 komórek pamięci operacyjnej.

2.1.1. Procesor

Procesor komputera EW składa się z:

- rejestru rozkazów (I), z wyodrębnionymi podrejestrami:
  - kodu operacji (KOD),
  - argumentu rozkazu (AD),
- licznika rozkazów (L),
- jednostki arytmetyczno-logicznej (JAL),
- rejestru akumulatora (AK),
- rejestru ogólnego przeznaczenia (X).

Ponadto procesor posiada:

- połączenia między magistralami danych i adresową w obu kierunkach,
- układy wyznaczające następujące sygnały stanu maszyny:

a) ØAK (oznaczający zerową zawartość akumulatora),
b) ZAK (oznaczający ujemną zawartość akumulatora),
c) ØS (oznaczający zero na magistrali S),
d) ZS (oznaczający ujemną liczbę na magistrali S),
e) INT (oznaczający żądanie przerwania - por. p. 2.1.3),
f) ORQ (oznaczający zgłoszenie operatora - z klawiatury sterującej).

Sygnały stanu mogą być testowane w układzie sterowania maszyny. Warto również zwrócić uwagę na istnienie sygnału mikrosterującego "jeden" ustawiającego liczbę 1 na magistrali danych.

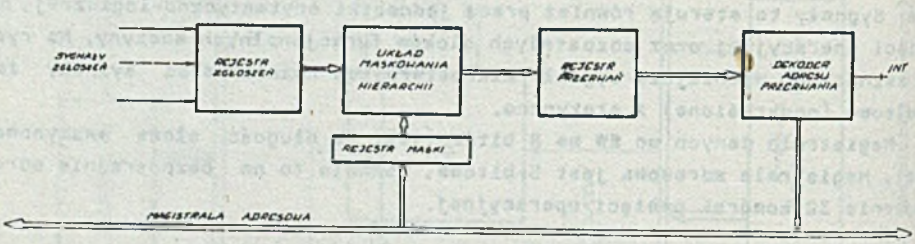
Struktura procesora maszyny pozwala na realizację bardziej złożonych rozkazów będących ilustracją możliwości funkcjonalnych współczesnych komputerów.

2.1.2. Pamięć operacyjna

Pamięć operacyjna maszyny (PaO) ma pojemność 32 słów 8-bitowych. Jest to pamięć półprzewodnikowa bipolarna, typu RAM. Na rys. 1 zaznaczono rejestr adresowy (A) i rejestr danych (S) pamięci.

2.1.3. Układ przerwań

Uproszczony schemat układu przerwań przedstawiono na rys. 2.



Rys. 2. Uproszczony schemat układu przerwań

INT - sygnał przerwania generowany na podstawie sygnałów zgłoszeń, jeden z sygnałów stanu maszyny.

Układ przerwań komputera "EW" charakteryzuje się następującymi cechami:

- zgłoszenia generowane są na 4 poziomach,
- źródłami zgłoszeń są przyciski klawiatury sterującej oraz system we/wy maszyny,
- możliwość maskowania poszczególnych poziomów,
- automatycznie generowany jest adres przerwania (układ wektorowy).

2.1.4. System wyjścia - wyjścia

Zasadnicze cechy systemu we/wy maszyny EW są następujące:

- dwa układy pośredniczące (interfejsu) umożliwiają współpracę komputera z czytnikiem i perforatorem taśmy papierowej,
- obsługa urządzeń zewnętrznych może odbywać się zarówno z wykorzystaniem układu przerwań, jak i bez jego pośrednictwa,
- system we/wy może być w łatwy sposób rozbudowany poprzez dodanie nowych układów interfejsu.

Układ przerwań, system we/wy i problemy związane z komunikacją maszyny "EW" z otoczeniem zostały szczegółowo omówione w [4].

2.2. Tablica synoptyczna

W połączeniu z tablicą synoptyczną klawiatura sterująca pełni funkcje pulpitu operatora maszyny.

Tablica synoptyczna pozwala na bieżącą kontrolę zawartości rejestrów oraz sygnałów mikrosterujących. Jest to nieodzowne dla celów dydaktycznych, gdyż pozwala na krokowe śledzenie realizacji rozkazów.

Funkcję indykatorów stanów rejestrów i sygnałów mikrosterujących pełnią wyświetlacze na diodach elektroluminescencyjnych.

2.3. Klawiatura sterująca

Klawiatura umożliwia sterowanie pracą maszyny "EW", a w szczególności pozwala na:

- wybór rodzaju pracy maszyny:
  1) praca krokowa (mikroinstrukcja po mikroinstrukcji),
  2) praca krokowa (rozkaz po rozkazie),
  3) praca cykliczna,
- inicjację i zatrzymanie pracy,
- zerowanie ogólne układów,
- generację zgłoszeń na dwóch poziomach układu przerwań,
- wpis informacji do pamięci operacyjnej oraz do pamięci mikroprogramów.

Schematycznie wygląd całego zestawu laboratoryjnego komputera "EW" przedstawiono na rys. 3.

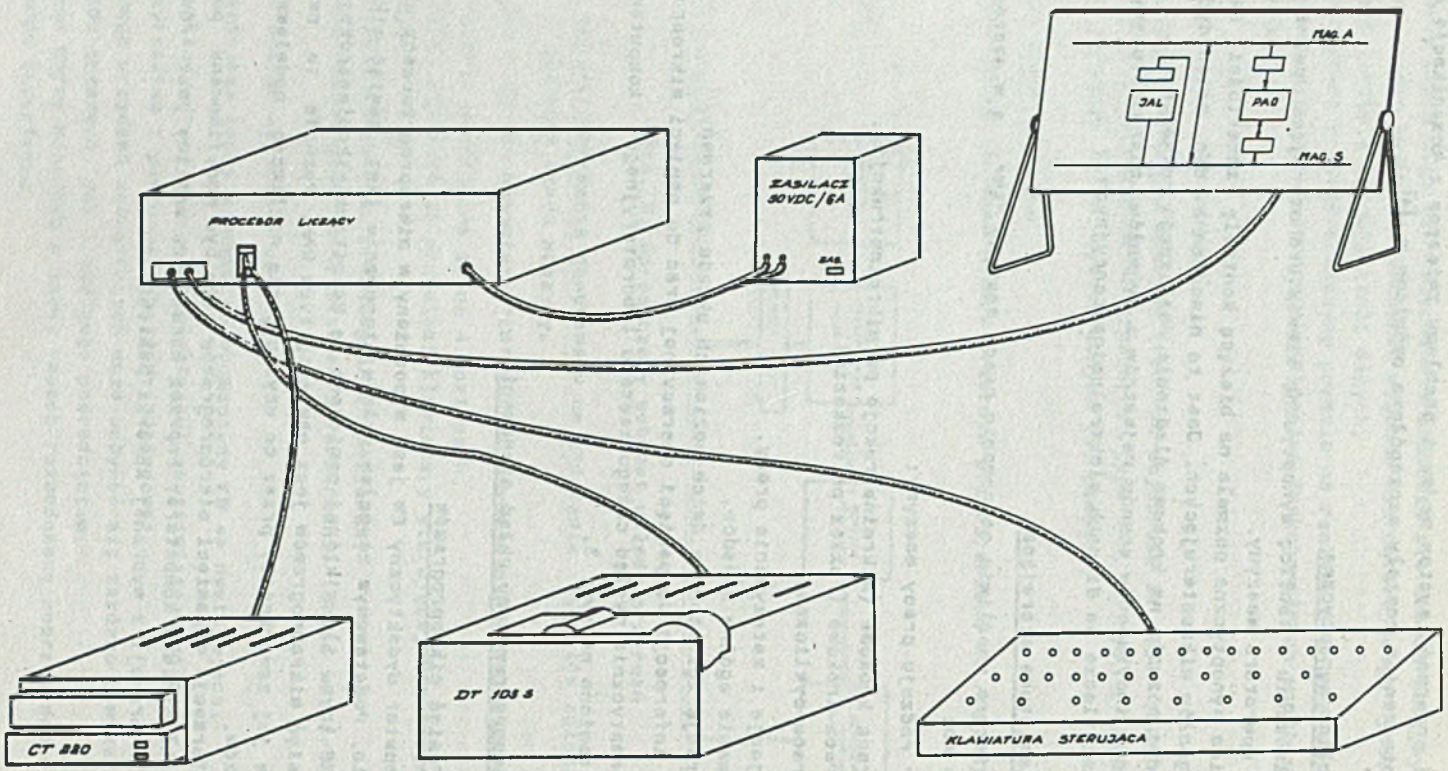
3. Mikroprogramowany układ sterowania

3.1. Pamięć mikroprogramów

Komputer dydaktyczny EW jest wyposażony w mikroprogramowany układ sterowania. Podstawowym zespołem układu sterowania jest pamięć mikroprogramów PMP (rys. 1), w której pamiętane są 24-bitowe mikroinstrukcja.

Pamięć mikroprogramów jest pamięcią typu RAM. Pozwala to na dowolną zmianę jej zawartości, przez co uzyskuje się możliwość modelowania listy rozkazów.

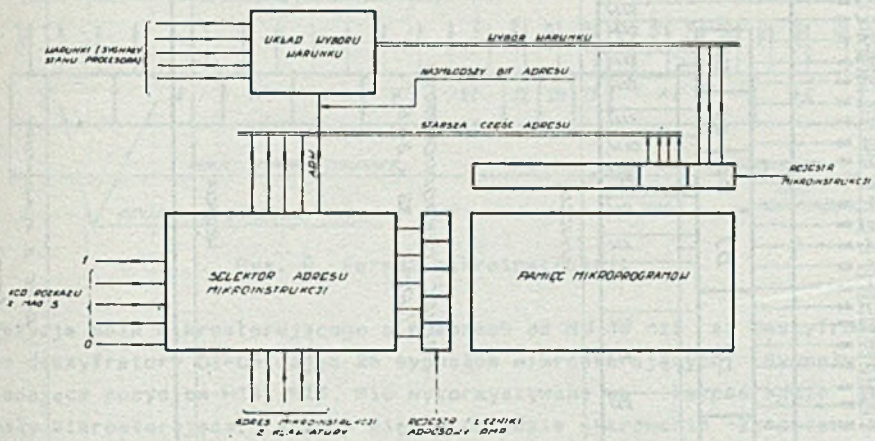
Informacja w pamięci mikroprogramów może być modyfikowana poprzez zapis z klawiatury komputera. Poprzez klawiaturę możliwy jest również odczyt informacji z wybranej komórki pamięci.



Rys. 3. Konfiguracja zestawu laboratoryjnego komputera EW

3.2. Adresacja pamięci mikroprogramów

Sposób adresacji pamięci mikroprogramów przedstawiono na rys. 4.



Rys. 4. Poglądowy schemat tworzenia adresu mikroinstrukcji

Pamięć mikroprogramów adresowana jest zawartością 5-bitowego licznika adresowanego LA (rys. 4, 5).

Adres do licznika LA wpisywany jest z trzech źródeł.

1. Z magistrali S - jako kod rozkazu wyczytanego z pamięci operacyjnej PaO - sygnały $\overline{S5}$, $\overline{S6}$, $\overline{S7}$ (rys. 5). Sygnały te podawane są odpowiednio na pozycje A3, A2, A1 licznika adresowego. Na pozycji A4 i AØ jest wówczas wymuszany stan wysoki odpowiadający "1", a na pozycji AØ stan niski ("0").

2. Z magistrali M - jako adres następnej mikroinstrukcji - mikrosygnały M17 - M20. W tym przypadku stan najmniej znaczącej pozycji licznika adresowego określa sygnały ADW - generowany przez selektor S1.

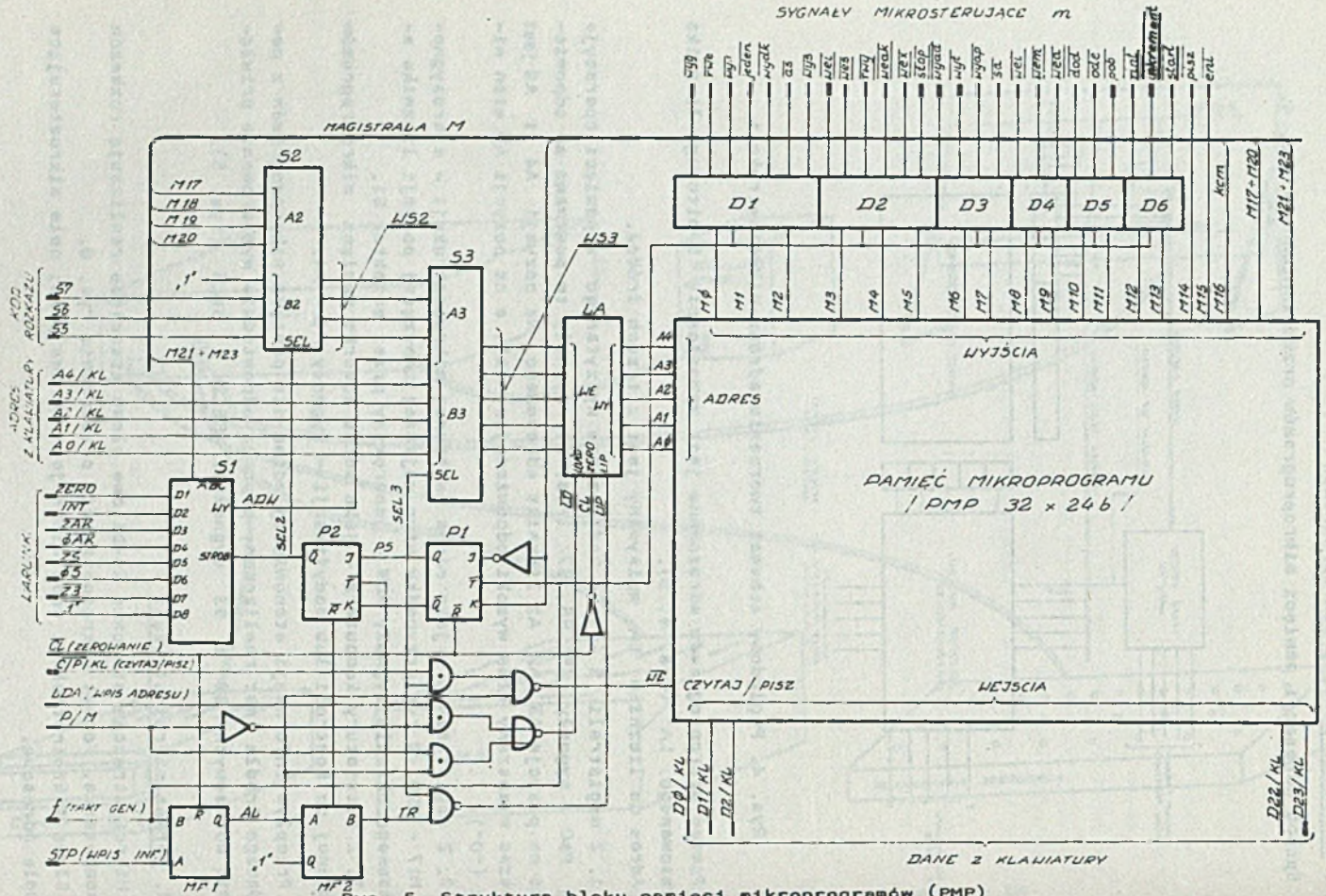
3. Z klawiatury komputera - jako adres komórki pamięci mikroprogramów wybranej do kontroli lub modyfikacji - sygnały AØ - A4.

Przepływ informacji stanowiącej adres dla pamięci mikroprogramów z pożądanego źródła jest realizowany poprzez odpowiednie wysterowanie przełącznic adresowych S2 i S3 sygnałami SEL2 i SEL3 (rys. 5).

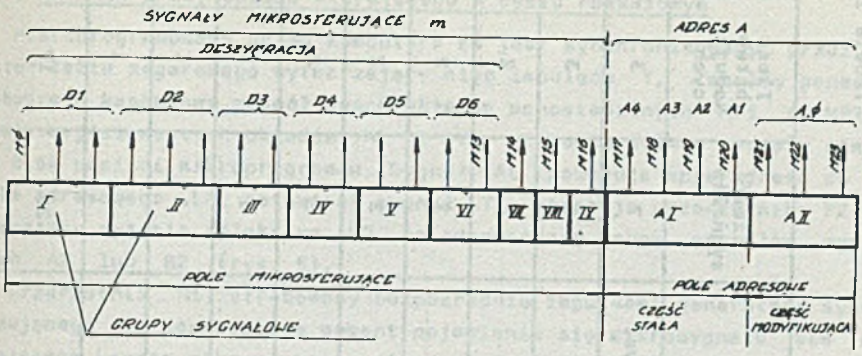
3.3. Format mikroinstrukcji

Mikroinstrukcje stanowią 24-bitowe słowa sterujące realizacją rozkazów w komputerze. Format mikroinstrukcji przedstawia rys. 6.

Słowo mikroinstrukcji podzielone jest na dwa pola; pole mikrosterujące i pole adresowe.



Rys. 5. Struktura bloku pamięci mikroprogramów (PMP)



Rys. 6. Format mikroinstrukcji

Pozycja pola mikrosterującego o numerach od M0 do M13 są deszyfrowane przez deszyfratory D1-D6 dając 26 sygnałów mikrosterujących. Sygnały odpowiadające pozycjom M14, M15, M16 wykorzystywane są bezpośrednio jako sygnały mikrosterujące. Łącznie więc w układzie sterowania komputera wytwarzanych jest 28 mikrosygnałów. Funkcje poszczególnych grup sygnałowych oraz opis symboliczny przyporządkowanych im mikrosygnałów przedstawia tabela 1.

Najmniej znaczące pozycje słowa mikroinstrukcji M17-M23 są wykorzystywane do formowania adresu następnej mikroinstrukcji w mikroprogramie realizującym funkcje danego rozkazu. Pole adresowe mikroinstrukcji jest podzielone na dwie części, z których pierwsza (M17, M18, M19, M20) tworzy bezpośrednio 4 bardziej znaczące bity adresowe A4, A3, A2, A1, natomiast druga część pola dresowego (M21, M22, M23) jest wykorzystywana do adresowania selektora S1, którego sygnał wyjściowy ADW tworzy piąty najmniej znaczący bit adresu A0. Sten tego bitu odpowiada sygnałowi warunku wybranego w selektorze S1.

Sygnałami warunków są następujące sygnały stanu procesora:

0AK - sygnalizacja zerowej zawartości rejestru AK,
ZAK - sygnalizacja ujemnej zawartości rejestru AK,
0S - sygnalizacja zerowej informacji na mag. S,
ZS - sygnalizacja ujemnej inf. na mag. S,
INT - sumaryczny sygnał przerwania,
ORQ - sygnał zgłoszenia operatora,
"1", "0" - wymuszone stany sygnałów warunków.

Wykorzystując sygnały warunków, uzyskuje się możliwość modyfikacji adresu następnej mikroinstrukcji, co pozwala na realizację rozgałęzień w mikroprogramach.

Tabela 1

Tabela sygnałów mikrosterujących

Grupa sygnałów	Funkcja grupy sygnałów	Symbole sygnałów	Ilość sygnałów	Ilość bitów kodujących
I	Wyjście na mag. S	wyg, rwe, wyx, jeden, wysk, as, wye.	7	3
II	Wejście z mag. S	wej, wes, rwy, wesk, wex	5	3
III	Wyjście na mag. A	wysd, wyl, wysp, sa	4	2
IV	Wejście z mag. A	wel, wem, wea	3	2
V	Funkcje JAL	dodaj, odejmij, pobierz	3	2
VI	Inne sygnały ster.	stop, inkrement, start	3	2
VII	Zerowanie przerwania	rint	1	1
VIII	Przyjęcie zgłoszenie	ani	1	1
IX	Koniec cyklu rozkaz.	kcm	1	1
		R a z e m	28	17

3.4. Analiza pracy układu sterującego w cyklu rozkazowym

Mikroprogramowany układ komputera EW jest synchronizowany przez generator taktu zegarowego wytwarzający ciąg impulsów f. Impulsy generatora pobudzają kaskadowy zespół dwóch układów monostabilnych MF1 i MF2. Sygnały wyjściowe tych układów AL i TR stanowią główne sygnały sterujące blok pamięci mikroprogramów. Sygnał AL powoduje wpis adresu do licznika adresowego LA, natomiast sygnał TR strobuje przerzutnik P2 przełączający wyjścia selektora S2 na odpowiednią grupę sygnałów wyjściowych A2 lub B2 (rys. 5).

Przerzutnik P1 strobowany bezpośrednio impulsami generatora synchronizującego f zapamiętuje moment pojawienia się mikrosygnału kcm oznaczającego koniec mikroprogramu dla danego rozkazu.

W każdym cyklu rozkazowym składającym się z n faz wyróżnia się dwie charakterystyczne mikroinstrukcje oznaczające początek i koniec mikroprogramu danego rozkazu.

Mikroinstrukcja początkowa realizuje pobranie rozkazu z pamięci operacyjnej komputera do rejestru rozkazów. W trakcie tej mikroinstrukcji do licznika adresowego LA jest wpisywany kod rozkazu, dla którego wykonywany będzie określony mikroprogram. Kod rozkazu określa początkowy adres mikroprogramu dla danego rozkazu. Mikroinstrukcja początkowa może być umieszczana w pamięci mikroprogramów pod dowolnym adresem.

Charakterystyczny stan dla tej mikroinstrukcji opisuje relacja:

$$SEL2 \cdot \overline{PS} \cdot GT = 1$$

Mikroinstrukcja końcowa odznacza się zakodowaniem w niej sygnału "końca mikroprogramu" kcm. Sygnał ten powoduje dla następnej mikroinstrukcji (początkowej dla kolejnego rozkazu) wysterowanie przełącznicy S2 na wejście B2, co umożliwia wpis kodu rozkazu do licznika adresowego LA.

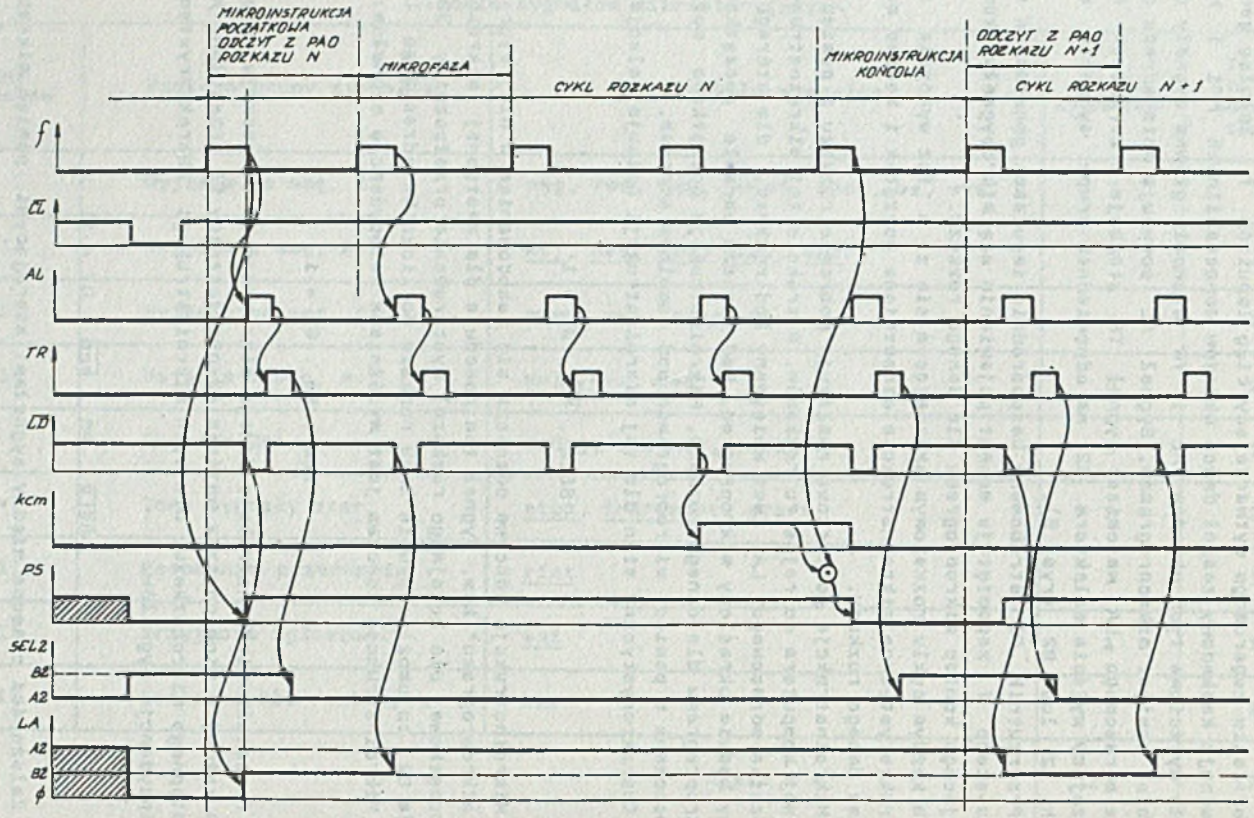
Mikroinstrukcja końcowa jest wyróżniana koincydencją sygnałów:

$$kcm \cdot PS \cdot GT = 1$$

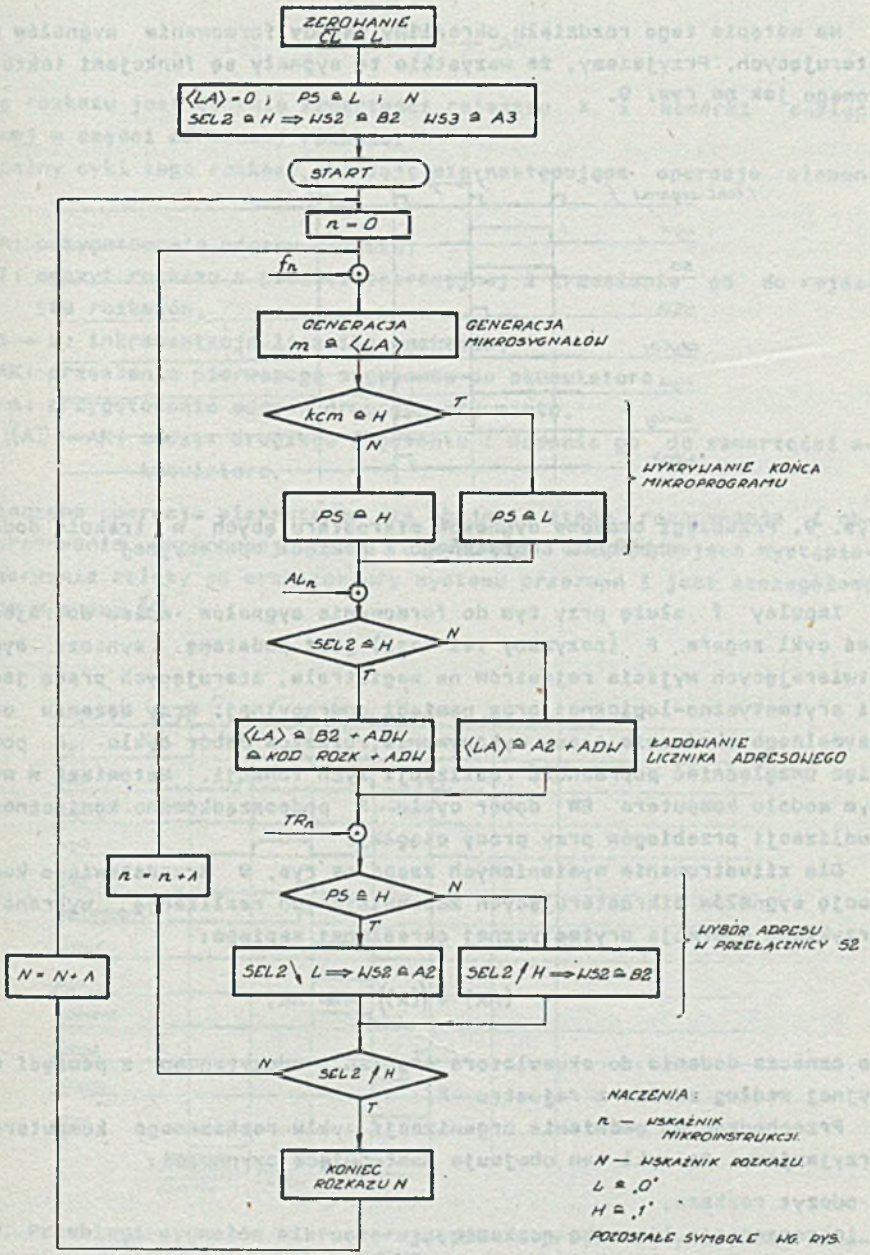
Pozostałe mikroinstrukcje danego mikroprogramu odznaczają się tym, że dla każdej z nich poprzez wejścia A2 przełącznicy S3 przesyłany jest adres następnego mikrorozkazu. Dla tych mikroinstrukcji charakterystyczna jest koincydencja sygnałów:

$$\overline{SEL2} \cdot PS \cdot \overline{kcm} \cdot GT = 1$$

Zależności czasowe między sygnałami sterującymi pamięć mikroprogramów w cyklu rozkazowym przedstawiono na rys. 7. Natomiast na rys. 8 przedstawiono schemat blokowy działania układu sterującego pamięć mikroprogramów.



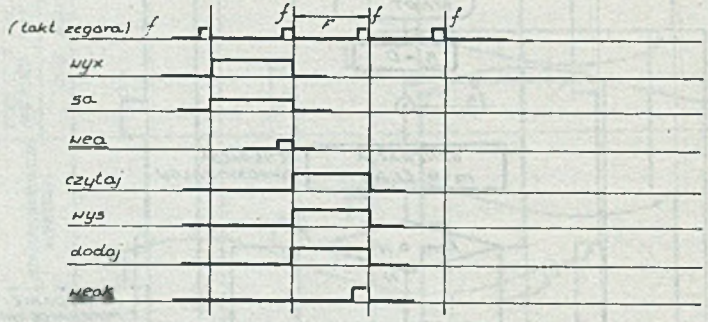
Rys. 7. Diagram zależności czasowych sygnałów sterujących PMP w cyklu rozkazowym



Rys. 8. Schemat blokowy działania układu sterującego PMP w cyklu rozkazowym

4. Zasady organizacji cyklu rozkazowego omawianego komputera

Na wstępie tego rozdziału określimy zasady formowania sygnałów mikrosterujących. Przyjmiemy, że wszystkie te sygnały są funkcjami taktu zegarowego jak na rys. 9.



Rys. 9. Przebiegi czasowe sygnałów mikrosterujących w trakcie dodawania argumentu odczytanego z pamięci operacyjnej

Impulsy f służą przy tym do formowania sygnałów wpisu do rejestrów, zaś cykl zegara F (nazywany też fazą) jest podstawą syntezy sygnałów otwierających wyjścia rejestrów na magistrale, sterujących pracą jednostki arytmetyczno-logicznej oraz pamięci operacyjnej. Przy dążeniu do maksymalnego skrócenia czasu wykonywania rozkazów dobór cyklu F powinien więc uwzględniać poprawność realizacji tych funkcji. Natomiast w wykonanym modelu komputera EW dobór cyklu F podporządkowano konieczności wizualizacji przebiegów przy pracy ciągłej.

Dla zilustrowania wymienionych zasad na rys. 9 przedstawiono konfigurację sygnałów mikrosterujących zapewniających realizację wybranej dla przykładu operacji arytmetycznej określonej zapisem:

$$(AK) + ((X)) \longrightarrow AK,$$

co oznacza dodanie do akumulatora argumentu odczytanego z pamięci operacyjnej według adresu z rejestru X.

Przechodząc do omówienia organizacji cyklu rozkazowego komputera EW, przyjmujemy, że cykl ten obejmuje następujące czynności:

- odczyt rozkazu,
- inkrementacja licznika rozkazów,
- wykonanie funkcji rozkazu,
- rozpoznanie i obsługa przerwania,
- przygotowanie adresu następnego rozkazu.

Dla przykładu rozpatrzmy szczegółowo organizację cyklu rozkazowego dla rozkazu DOX o funkcji określonej zapisem:

$$(X) + ((AD)) \rightarrow AK$$

Treścią rozkazu jest dodanie zawartości rejestru X i komórki pamięci wskazanej w części adresowej rozkazu.

Na pełny cykl tego rozkazu składają się następujące operacje elementarne:

(L) → A: przygotowanie adresu rozkazu,

((A)) → I: odczyt rozkazu z pamięci operacyjnej i przesłanie go do rejestru rozkazów,

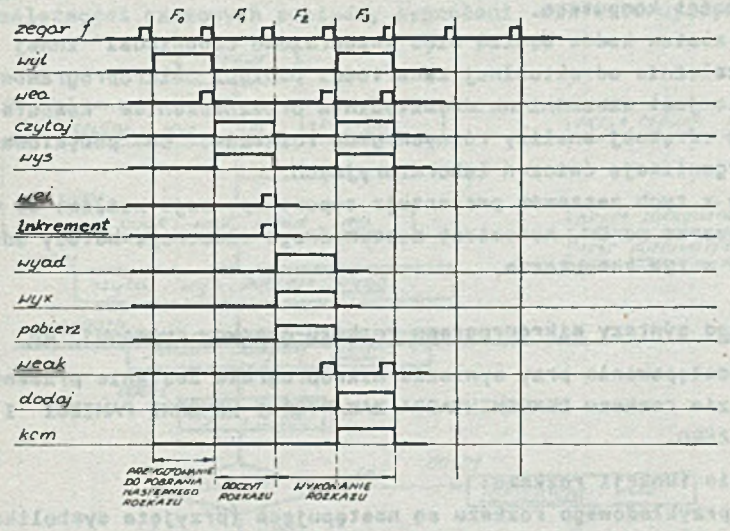
(L) + 1 → L: inkrementacja licznika rozkazów,

(X) → AK: przesłanie pierwszego argumentu do akumulatora,

(AD) → A: przygotowanie adresu drugiego argumentu,

(AK) + ((A)) → AK: odczyt drugiego argumentu i dodanie go do zawartości akumulatora.

Wymienione operacje elementarne nie obejmują etapu rozpoznania i obsługi przerwania. Organizacja cyklu rozkazowego uwzględniająca wystąpienie przerwania zależy od architektury systemu przerwań i jest szczegółowo omówiona w pracy [4].



Rys. 10. Przebiegi sygnałów mikrosterujących w przykładowym cyklu rozkazowym

Przebiegi sygnałów mikrosterujących dla tego rozkazu przedstawia rysunek 10.

Dla ułatwienia analizy kolejne cykle zegarowe (fazy) zostały ponumerowane. Zauważmy, że w fazie F_0 następuje przesłanie zawartości licznika rozkazów do rejestru adresowego pamięci operacyjnej. Jest to przygotowania do pobrania rozkazu i można tę operację wykonać w ostatniej fazie poprzednio wykonywanego rozkazu. Zgodnie z tą zasadą w fazie F_3 umieszczono sygnały "wyl" i "wea" realizujące przygotowanie do pobrania następnego rozkazu. Przy takim rozwiązaniu można zrezygnować z wyodrębniania fazy F_0.

Dla poprawnej realizacji rozkazu konieczne jest wygenerowanie w ostatniej fazie danego rozkazu (tzn. F_3 dla rys. 10) sygnału "kom" umożliwiającego w fazie następnej (tzn. F_1) wybór właściwego mikroprogramu wykonania rozkazu.

5. Modelowania listy rozkazów w maszynie cyfrowej EW

5.1. Ogólne zasady realizacji listy rozkazów

Strukturą maszyny cyfrowej EW pozwala na realizację większości typowych rozkazów komputerowych. Ośmiobitowa długość słowa rozkazowego ogranicza jednak istotnie liczbę rozkazów, którym można przyporządkować różne kody.

Wykorzystując łatwość programowania pamięci mikroprogramów, można jednak zmieniać zestaw rozkazów wybierając do prezentacji tylko określoną część własności komputera.

Ten sam zestaw kodów będzie więc każdorazowo odpowiadał innej grupie rozkazów, zależnie od aktualnej zawartości pamięci mikroprogramów. Takie postępowanie jest uzasadnione dydaktycznym przeznaczeniem komputera EW, wymagającym odrębnej analizy różnych grup rozkazów, co podyktowane jest przyjętą organizację ćwiczeń laboratoryjnych.

Pierwszy z tych zestawów prezentuje typowe operacje możliwe do wykonania w procesorze mc EW, natomiast zestaw drugi ilustruje metody adresacji realizowane w tym komputerze.

5.2. Przykład syntezy mikroprogramu rozkazu maszyny cyfrowej "EW"

Sposób postępowania przy syntezie mikroprogramu zostanie przedstawiony na przykładzie rozkazu DEKREMENTACJA ZAWARTOŚCI KOMÓRKI PAMIECI I PRZESKOK JEŚLI ZERO.

a) Określenie funkcji rozkazu:

Funkcje przykładowego rozkazu są następujące (przyjęta symbolika odpowiada oznaczeniom z rys. 1):

- pobranie rozkazu z PaO do rejestru I: $((A)) \rightarrow I$,
- inkrementacja licznika rozkazów: $(L) + 1 \rightarrow L$,
- przesłanie zawartości AK do X: $(AK) \rightarrow X$,

- pobranie zawartości komórki PaO do AK: ((AD)) → AK,
- dekrementacja zawartości AK: (AK) - 1 → AK,
- przesłanie nowej wartości AK do PaO: (AK) → (A),
- jeżeli (AK) = ∅, to inkrementacja L: (L) + 1 → L,
- odtworzenie pierwotnej zawartości AK: (X) → AK,
- przygotowanie adresu następnego rozkazu (L) → A.

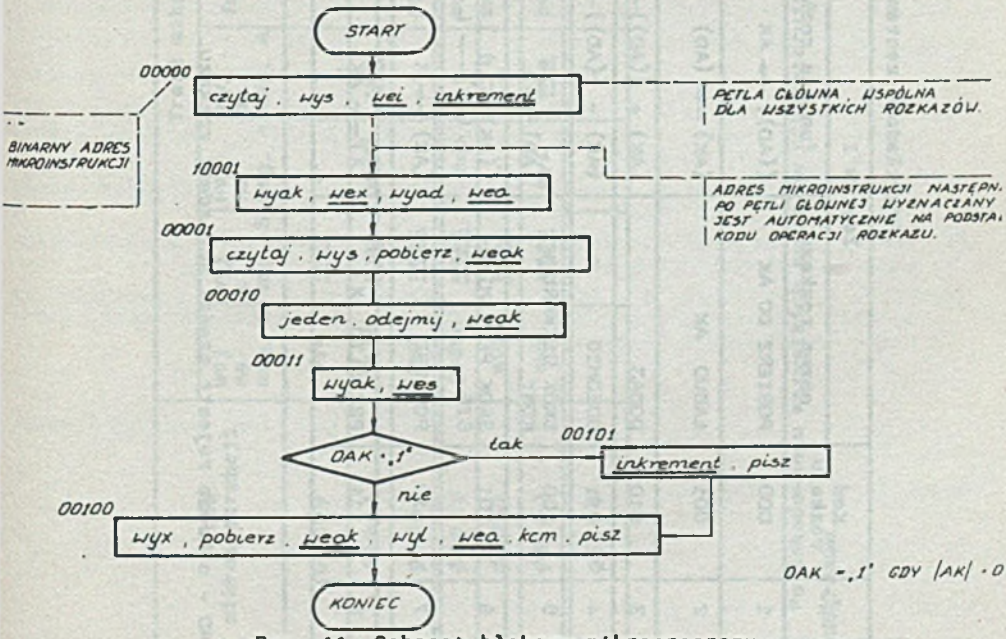
Z przeprowadzonej analizy wynika, że rozkaz niszczy zawartość rejestru X.

b) Określenie sygnałów mikrosterujących realizujących poszczególne funkcje:

- ((A)) → I: czytaj, wys, wei
- (L) + 1 → L: inkrement
- (AK) → X: wysk, wex
- ((AD)) → AK: wysd, wea,
  czytaj, wys, pobierz, weak
- (AK) - 1 → AK: jeden, odejmij, weak
- (AK) → (A): wysk, wea, pisz
- (AK) = ∅ ⟹ (L) + 1 → L: inkrement
- (X) → AK: wyx, pobierz, weak
- (L) → A: wyl, wea.

c) Schemat blokowy mikroprogramu

Dla rozkazów bardziej złożonych, szczególnie gdy istotna jest dokładna analiza zależności czasowych pomiędzy sygnałami mikrosterującymi, przy-



Rys. 11. Schemat blokowy mikroprogramu

Tabela 2

Przykładowe zestawy rozkazów maszyny cyfrowej EW

Lp.	Kod rozkazu	Zestaw I		Zestaw II	
		nazwa rozkazu	funkcja rozkazu	nazwa rozkazu	funkcja rozkazu
1	000	POBIERZ DO AK	((AD)) → AK	SKOK BEZWARUNKOWY POŚREDNI	((AD)) → L
2	001	ŁADUJ AK	(AK) → (AD)	SKOK WARUNKOWY BEZPOŚREDNI	((AK) = ∅) ⇒ (AD) → L
3	010	DODAJ	(AK) + ((AD)) → AK	POBIERZ DO AK INDEKSOWO	((AD) + (X)) → AK
4	011	ODEJMIJ	(AK) - ((AD)) → AK	POBIERZ DO AK POŚREDNIO	(((AD))) → AK
5	100	SKOK BEZWARUNKOWY	(AD) → L	POBIERZ DO AK INDEKSOWO POŚREDNIO	(((AD)) + (X)) → AK
6	101	SKOK PRZY MINUSIE	((AK) < ∅) ⇒ ⇒ (AD) → L	SKOK WZGLĘDNY	(L) + (AD) → L
7	110	POBIERZ X	(AK) → X	POBIERZ DO X BEZPOŚREDNIO	((AD)) → X
8	111	PRZEŚLIJ X	(X) → AK	POBIERZ X NATYCHMIASTOWO	(AD) → X

AD - oznacza rejestr części adresowej rozkazu.

Tabela 3

Adres mikroinstrukcji	Treść mikroinstrukcji									Część adresowa	
	wej na mag. S	wyj z mag. S	wej na mag. A	wyj z mag. A	funkcja JAL	sygn. inne	eni	rint	kcm	część stała adresu	modyfikator
ØØØØØ	wys	wsi	-	-	-	inkrement	-	-	-	dowolna wartość	
1ØØØ1	wysk	wsx	wysd	wsa	-	-	-	-	-	ØØØØ	1
ØØØØ1	wys	wsak	-	-	pobierz	-	-	-	-	ØØØ1	Ø
ØØØ1Ø	jeden	wsak	-	-	odejmij	-	-	-	-	ØØØ1	1
ØØØ11	wysk	wsa	-	-	-	-	-	-	-	ØØ1Ø	ØAK
ØØ1ØØ	wyx	wsak	wyl	wsa	pobierz	-	-	-	kcm	ØØØØ	Ø
ØØ1Ø1	-	-	-	-	-	inkrement	-	-	-	ØØ1Ø	Ø

Po zakodowaniu wszystkich sygnałów mikrosterujących tabela jest podstawą do wypełnienia pamięci mikroprogramów.

datne jest sporządzenie wykresów czasowych jako etap pośredni między (b) i (c). Często jednak można przystąpić bezpośrednio po etapie (b) do budowy schematu blokowego mikroprogramu. Schemat mikroprogramu przykładowego rozkazu przedstawia rys. 11.

d) Rozmieszczenie mikroprogramu w pamięci

Tabela 3 przedstawia rozmieszczenie mikroprogramu w pamięci PMP. Pola kodowe (zgodnie z formatem mikroinstrukcji) zostały dla przejrzystości wyodrębnione.

Mikroinstrukcję pętli głównej umieszczono w komórce pamięci mikroprogramów o adresie Ø . Adres pierwszej mikroinstrukcji właściwego mikroprogramu zależy od kodu operacji. Przyjmując (całkowicie arbitralnie), że kod przykładowego rozkazu będzie równy $ØØØ_2$, adres pierwszej mikroinstrukcji będzie równy $1ØØØ1_2$.

Zakończenie

Przedstawiona w pracy maszyna cyfrowa EW została praktycznie zrealizowana i jest wykorzystywana w laboratoriach Instytutu Informatyki Czasu Rzeczywistego. Aktualnie wykorzystywane są dwa modele tej maszyny. Dotychczasowe doświadczenia związane z eksploatacją maszyny potwierdziły użyteczność przedstawionej koncepcji komputera EW i jego możliwości funkcjonalnych jako maszyny cyfrowej specjalizowanej do zastosowań dydaktycznych.

LITERATURA

[1] Węgrzyn S., Konsek M.: Przesyły międzyrejestrowe w programowanych maszynach cyfrowych o adresowanych rejestrach pamięci. Podstawy sterowania, 1975, t. 5, z. 4.

[2] Węgrzyn S., Konsek M., Mrózek A.: Mikrosterowanie i mikroprogramowanie. Podstawy Sterowania, 1976, t. 6, z. 1.

[3] Konsek M.: Organizacja cyklu rozkazowego jako regulamin obsługi zgłoszeń. Podstawy Sterowania, 1976, t. 6, z. 1.

[4] Gacek A., Kozielski S., Stróżyna P.: Modelowanie listy rozkazów dla potrzeb komunikacji komputera dydaktycznego z otoczeniem. Zeszyty Naukowe Politechniki Śląskiej, Informatyka, z. 5.

Recenzent: Prof. dr hab. inż. Andrzej Grzywak

Wpłynęło do Redakcji 15.09.1982 r.

ОРГАНИЗАЦИЯ ЭЛАСТИЧНО МИКРОПРОГРАММИРОВАННОЙ ДИДАКТИЧЕСКОЙ ЭВМ ТИПА EW

Р е з ю м е

В статье предоставлено структуру эластично микропрограммированной дидактической ЭВМ. Возможность эластического микропрограммирования полученно применяя запоминающее устройство с произвольным доступом (RAM) для запоминания микропрограмм. Благодаря этому можна проводить многовариантное моделирование набора команд в рамках данной структуры ЭВМ и реализовать множество команд больше чем это вытекает из ограниченной длины слова команд. В статье подробно предоставлено структуру и принцип действия блока запоминающего устройства для микропрограмм а также организацию цикла команд для выбранной группы машинных команд связанных с процессором.

ORGANIZATION OF EW FLEXIBLY MICROPROGRAMMED DIDACTICAL COMPUTER

S u m m a r y

The structure and organization of flexibly microprogrammed didactical computer is presented in the paper. The ability of flexible microprogramming has been accomplished by using RAM as microprogram memory. This enables to achieve multi-variant modelling of instruction set for a given computer structure. Moreover it gives a facility of performing more instructions than limited length of instruction word would allow. The structure and functioning of microprogram memory are described in detail. Organization of instruction cycle for a group of instructions for the computer processor is presented as an example.