S Y S T E M M E R A - 100

P O D R Ę C Z N I K P R O G R A M O W A N I A

ZAKŁADY MECHANICZO-PRECYZYJNE "MERA-BŁONIE"

05-870 BŁONIE, UL. GRODZISKA 15

1. Wprowadzenie

2. Przedstawiamy MERĘ - 100

   a/ Format danych

   b/ Format instrukcji

   c/ Konfiguracja

   d/ Rejestry procesora

   e/ Pamięć

3. Programowanie

   a/ Sieć logiczna programu

   b/ Język programowania

4. Instrukcje

   a/ Instrukcje, realizowane w procesorze

   b/ Instrukcje, dotyczące pamięci kasetowej PK-1

   c/ Rozkazy, dotyczące pracy monitora ekranowego VDU lub CRT

   d/ Rozkazy, dotyczące pamięci na dyskach elastycznych /FD/

   e/ Rozkazy, dotyczące jednostki, sterującej transmisją danych

   f/ Rozkazy, dotyczące konwertera

5. Rutyny

   a/ Rutyny, których działanie dotyczy procesora, drukarki i klawiatury

   b/ Rutyny, wykorzystywane przy pracy z PK-1

   c/ Rutyny, wykorzystywane przy pracy z pamięcią na dyskach elastycznych

6. Tablice:

   a/ Tablica 1. Rozkazy MERY-100

   b/ Tablica 2. Kody ASCII /klawiatury, drukarki, monitora ekranowego i transmisji/

C-34
44-43
WAT 50-61

Wprowadzenie

Arytmetyka istnieje od czasów, kiedy zaistniała potrzeba liczenia. Rozwój arytmetyki i metod numerycznych pozwolił na osiągnięcie dzisiejszego poziomu cywilizacji technicznej. Od zarania dziejów arytmetyki ludzie starali się ułatwić sobie żmudny proces liczenia. Powstał abak, później arytmometr. Dziś są komputery.

Podstawową jednostką informacji jest bit - rozróżnienie dwóch stanów: TAK lub NIE, 1 lub 0, jest sygnał lub go nie ma. W układach TTL stan 1 jest utożsamiany z obecnością dodatniego napięcia około +5V, zaś stan 0 z napięciem 0V. Ponieważ działanie urządzeń cyfrowych opiera się właśnie na takim dwuwartościowym przedstawieniu informacji, to ich arytmetykę nazywa się arytmetyką binarną. Operacje są wykonywane nie nad pojedyńczymi bitami, ale nad ich grupami. Najczęściej spotykane są grupy 8-bitowe, zwane bajtami. Za pomocą 8 bitów można przedstawić 256 kombinacji, co odpowiada wartości 2^8. Ilość możliwych kombinacji z n binarnych elementów jest równa 2^n.

Prawa arytmetyki binarnej są analogiczne do praw arytmetyki dziesiętnej. Różnica polega na tym, że przeniesienie do starszej pozycji pojawia się nie po cyfrze 9, a po cyfrze 1:

7	111
+ 7	+ 111
14	1110

Analogia występuje również w interpretacji pozycji cyfr w liczbie:

$$14_{10} = 1 \cdot 10^1 + 4 \cdot 10^0;$$

$$1110_2 = 1 \cdot 2^3 + 1 \cdot 2^2 + 1 \cdot 2^1 + 0 \cdot 2^0 = 14_{10}.$$

Waga pozycji rośnie z prawej strony do lewej. Przyjęto też pozycje bitów w bajcie numerować zgodnie z potęgami dwójki, reprezentowanymi przez te pozycje:

0	0	0	1	1	1	0	1

pozycja: 7 6 5 4 3 2 1 0

Postać binarna informacji, choć czytelna dla maszyny, jest mało czytelna dla człowieka. Dlatego też jest stosowany system heksadecymalny: binarna tetrada /pół bajtu/ jest zastępowana przez jedną cyfrę heksadecymalną. Cały bajt jest często przedstawiany w postaci liczby dziesiętnej, jaką reprezentuje.

Przedstawiamy MERA-100

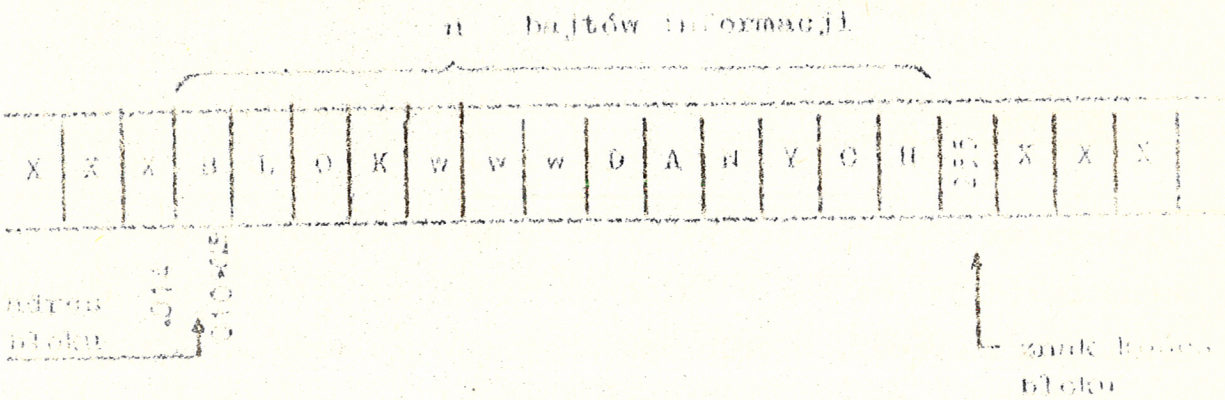
Minisystem MERA-100 został opracowany przez doświadczonych konstruktorów, na bazie najnowszych światowych osiągnięć techniki komputerowej, w celu zaspokojenia wielorakich potrzeb użytkowników. Szereg oryginalnych, nowoczesnych rozwiązań pozwoliło na osiągnięcie dużej elastyczności i wysokiej efektywności tego minikomputera przy stosunkowo niskiej jego cenie. Dużą elastyczność minisystemu uzyskano dzięki rozszerzeniu listy instrukcji przez makroinstrukcje /rutyny/, przechowywane w pamięci typu PROM. Rutyny mogą zajmować do 2 kB pamięci. Ich zestaw jest dobierany w zależności od potrzeb konkretnego użytkownika. Pamięć operacyjna jest zbudowana na układach scalonych średniej lub dużej skali integracji.

Cały proces przetwarzania danych, łącznie z arytmetyką dziesiętną, opiera się na bajtowej strukturze informacji.

Format danych

W minisystemie MERA-100 operacje wykonywane są na bajtach, reprezentujących liczby binarne z zakresu 0 ÷ 255, lub na blokach, reprezentujących teksty lub liczby dziesiętne /każdy bajt w bloku odpowiada jednej cyfrze dziesiętnej, przedstawionej w kodzie ASCII/.

Każdy blok jest identyfikowany według adresu pierwszego jego bajtu. Adres ten jest traktowany jako adres bloku. Za ostatnim bajtem informacyjnym występuje znak końca bloku - bajt o kodzie 255:



Operacje na blokach wykonywane są przez rutyny.

Format instrukcji

Minisystem MERA-100 jest programowany przy użyciu instrukcji jedno-, dwu- i trzybajtowych. Trzybajtowe instrukcje zawierają dwubajtowy operand - adres. Pozwalają one ustawiać rejestry adresowe i dokonywać skoków programowych. Dwubajtowe - to instrukcje ładowania rejestrów ośmiobitowych kodem, zawartym w drugim bajcie instrukcji. Jednobajtowe instrukcje, stanowiące większość nie zawierają operandów.

Podstawową konfigurację minisystemu MERA-100 tworzą:

- procesor /CPU/ i pamięć PROM,

- pamięć operacyjna /RAM/,
- klawiatura,
- drukarka mozaikowa DZM-180,
- pamięć kasetowa PK-1.

Znaczne rozszerzenie możliwości minikomputera uzyskać można drogą rozbudowy podstawowej konfiguracji o:

- monitor ekranowy /VDU lub CRT/,
- pamięć na dyskach elastycznych /Floppy disk/,
- pamięć taśmową,
- jednostkę sterowania transmisją danych.

Na rys.1 grubą linią ukazano bazowe podzespoły minikomputera, linią cienką - urządzenia rozszerzające konfigurację, a linią przerywaną - pozostałe możliwe do podłączenia dodatkowe urządzenia peryferyjne.

Rejestry procesora

R1 - 8-bitowy rejestr uniwersalny,
Z - licznik 8-bitowy,
V - 8-bitowy rejestr warunku,
LRC - 8-bitowy licznik modulo 2,
A i B - 4-bitowe rejestry arytmometru,
PADR - 16-bitowy licznik adresowy pamięci operacyjnej,
BADR - 16-bitowy licznik adresowy bufora,
PC - 16-bitowy rozkazowy licznik adresowy,

oraz

Stos o 5 poziomach, przechowujący dwubajtowe adresy,
R0 - tzw. Rejestr Operacyjny - obszar pamięci od adresu 009.000 do 009.255,
8 strona - obszar pamięci od adresu 008.000 do 008.255, w którym przygotowywany jest blok do zapisu na PK. Obszar ten jest wykorzystywany przez rutyny 003.000, 003.003, 003.019, 003.022

W wypadku, gdy programista nie używa w/w rutyn może wykorzystywać 8 stronę jako pamięć danych.
Funkcjonowanie minisystemu poglądowo ilustruje rys.2. Pokazano na nim kierunki bezpośredniego przesyłania informacji między rejestrami i urządzeniami.

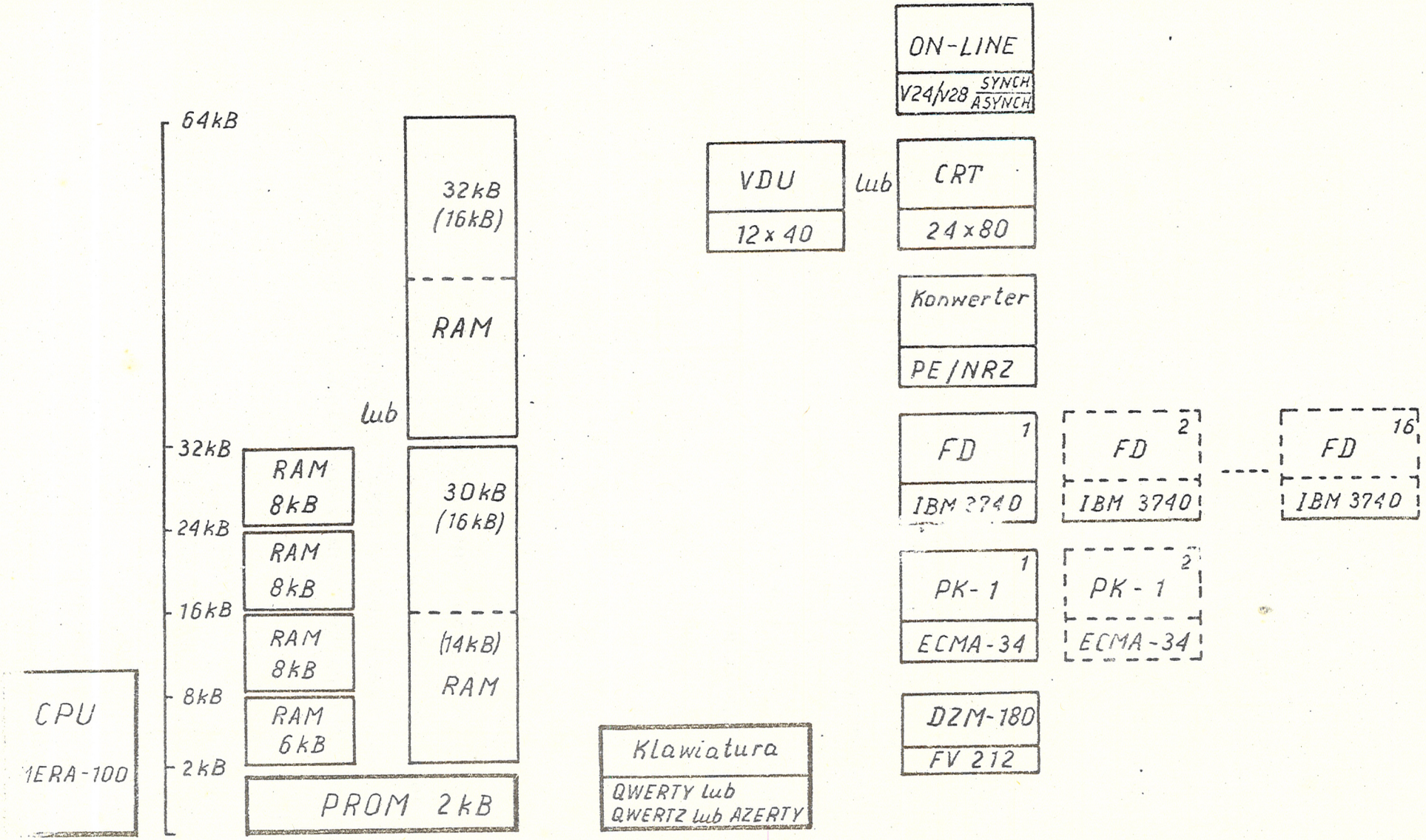
Pamięć

Pamięć operacyjna i pamięć buforowa stanowią jedną całość. Adresowanie pamięci odbywa się za pomocą dwubajtowych adresów. Pozwala to na zaadresowanie 2^{16} = 64 K bajtów.
Organizacja pamięci operacyjnej i buforowej w jednym bloku oraz obecność dwóch niezależnych rejestrów adresowych PADR i BADR pozwala na dostęp do pamięci operacyjnej niejako z dwóch stron. Organizację wewnętrzną pamięci ilustruje rys. 3. na str.7.

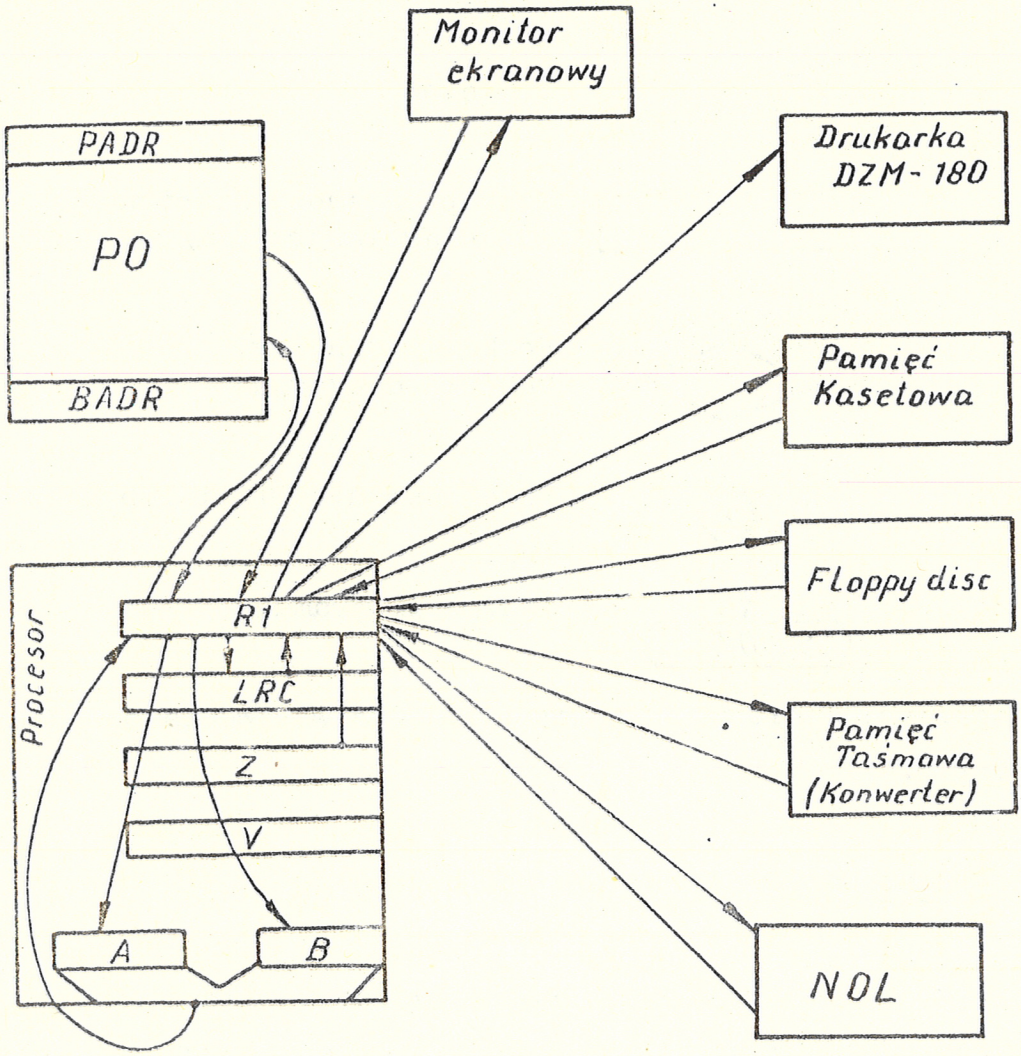
Programowanie

Sieć logiczna programu

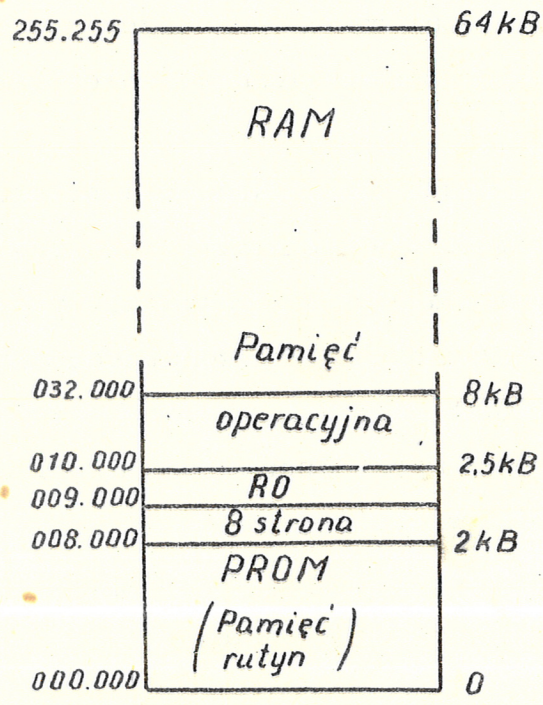
Pisanie programu celowo jest rozpoczynać od jego sieci logicznej. Za jej pomocą przedstawia się kolejność operacji w programie, rozgałęzienia i pętle, czyli strukturę logiczną programu. Proponujemy następujące oznaczenia elementów programu na sieci logicznej:


ON-LINE
V24/V28 SYNCH. ASYNCH.
VDU
12 x 40
lub
CRT
24 x 80
Konwerter
PE/NRZ
FD 1
IBM 3740
FD 2
IBM 3740
FD 16
IBM 3740
PK-1 1
ECMA-34
PK-1 2
ECMA-34
DZM-180
FV 212
64kB
32kB (16kB)
RAM
lub
32kB
RAM 8kB
24kB
RAM 8kB
16kB
RAM 8kB
8kB
RAM 6kB
2kB
30kB (16kB)
(14kB) RAM
PROM 2kB
CPU
1ERA-100
Klawiatura
QWERTY lub QWERTZ lub AZERTY

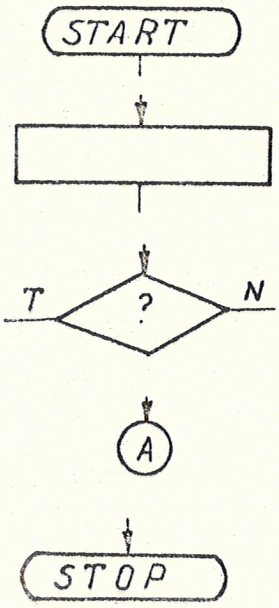
Rys. 1


Monitor ekranowy
PADR
PO
BADR
Drukarka DZM-180
Pamięć Kasetowa
Floppy disc
Pamięć Taśmowa (Konwerter)
NOL
Procesor
R1
LRC
Z
V
A
B

Rys. 2


255.255
64kB
RAM
Pamięć
032.000
operacyjna
8kB
010.000
2,5kB
009.000
R0
008.000
8 strona
2kB
PROM
(Pamięć rutyn)
000.000
0

Rys. 3



Początek programu

Krok programowy /instrukcja lub ciąg instrukcji/

Rozkaz warunkowy, komórka decyzyjna

Przeniesienie linii lub łącznik

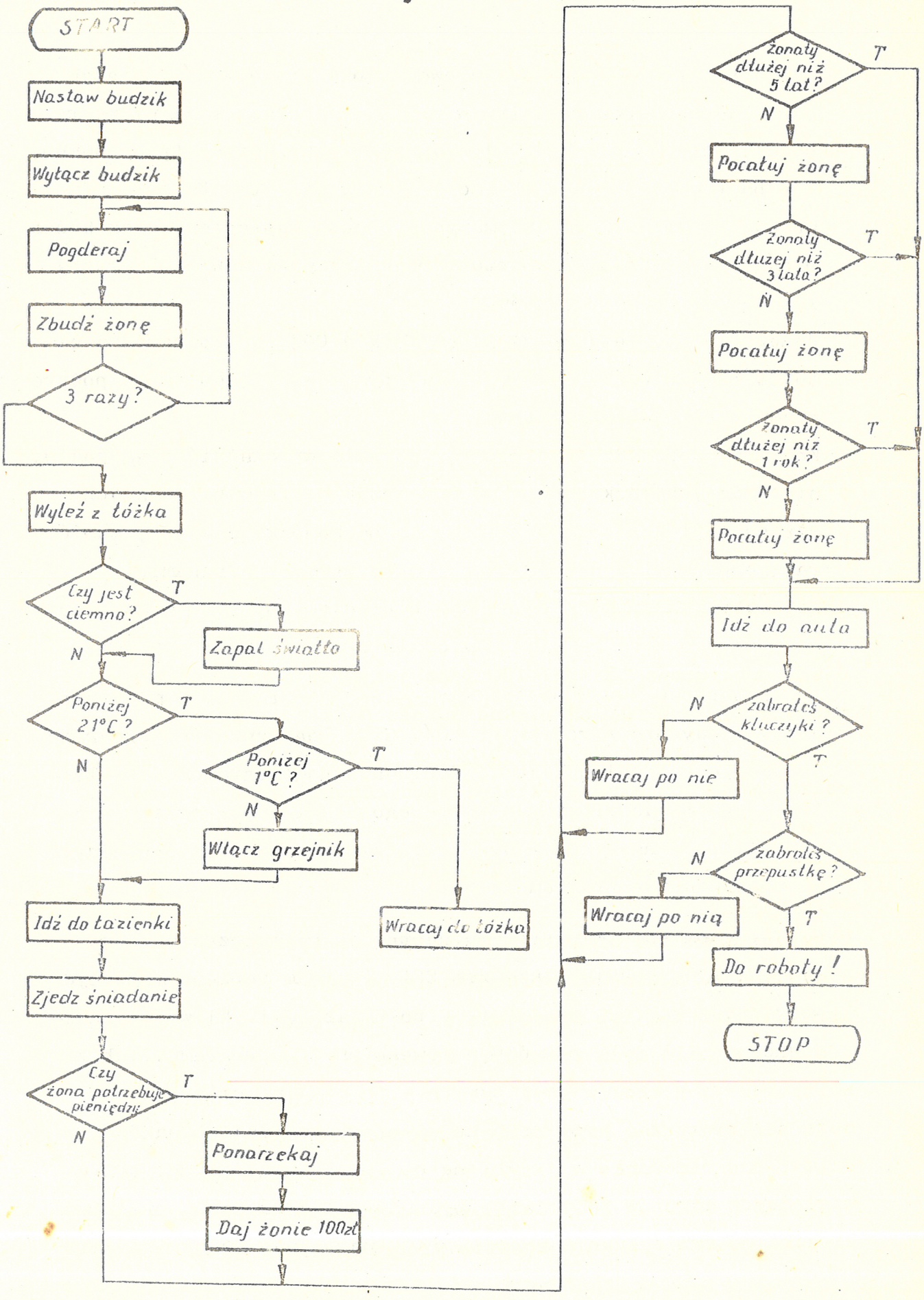
Koniec programu.

Przykład sieci logicznej programu rannego wstawania pokazany jest na rys.4.

Język programowania

Język programowania MERY-100 jest językiem o silnej orientacji maszynowej. Rozkazy są pisane w kodach dziesiętnych, które procesor interpretuje jako binarne kody wewnętrzne. Każdy rozkaz może posiadać etykietę. Wprowadzenie etykiet pozwala na wykorzystanie w programach zarówno absolutnych adresów, jak i adresów względnych, reprezentowanych właśnie przez etykiety.

Etykiety znacznie ułatwiają pisanie i testowanie programów. W wypadku konieczności wprowadzenia poprawki w programie drogą ustawienia lub wykasowania jakiegoś rozkazu, zmieniają się adresy kolejnych rozkazów. Korekcja absolutnych adresów w instrukcjach skoków itp. jest zajęciem pracochłonnym i może się powtarzać przy każdej wprowadzonej poprawce. Jak widzimy, programowanie w języku wejściowym z wykorzystaniem względnych adresów /etykiet/ jest pierwszym i najważniejszym krokiem do zmniejszenia kosztów programowania.


START
Nastaw budzik
Wyłącz budzik
Pogderaj
Zbudź żonę
3 razy?
Wyleź z łóżka
Czy jest ciemno?
T
Zapal światło
N
Poniżej 21°C?
T
Poniżej 1°C?
T
N
Włącz grzejnik
N
Idź do łazienki
Wracaj do łóżka
Zjedz śniadanie
Czy żona potrzebuje pieniędzy
T
N
Ponarzekaj
Daj żonie 100zł
Żonaty dłużej niż 5 lat?
T
N
Pocałuj żonę
Żonaty dłużej niż 3 lata?
T
N
Pocałuj żonę
Żonaty dłużej niż 1 rok?
T
N
Pocałuj żonę
Idź do auta
N
Zabrałeś kluczyki?
T
Wracaj po nie
N
Zabrałeś przepustkę?
T
Wracaj po nią
Do roboty!
STOP

Rys. 4

W programie można deklarować bloki /rejestry/. Za etykietą, określającą adres rejestru w pamięci /i służącą za nazwę rejestru/, wpisuje się kod 001, za nim można zapisać tyle znaków, ile bajtów powinien zawierać tworzony rejestr. Za ostatnim wprowadzonym znakiem automatycznie jest zapisywany znak końca bloku /kod 255/. Tak tworzone rejestry mogą zawierać liczby lub teksty /do 100 znaków/.
Jeżeli za etykietą wpisany będzie kod 003, to wpisany za nim tekst będzie w czasie kompilacji traktowany jako nazwa podprogramu z biblioteki systemu.
Kod 006, poprzedzony etykietą, pozwala wprowadzić ciąg kodów, nie będących rozkazami, lub ciąg etykiet, symbolizujących adresy. W ten sposób organizowane są komórki pamięci, zawierające parametry niezbędne np. przy pracy z adresami.
Rejestry, nazwy podprogramów z biblioteki systemu oraz komórki pamięci z parametrami należy deklarować na końcu programu.

Program w języku wejściowym pisany jest przy pomocy programu - edytora /program nr 50/. Sieć logiczna tego programu podana jest w załączniku 1. Dopisanie podprogramów z biblioteki systemu do programu źródłowego realizuje program nr 52, a kompilację na program wynikowy /zamiana adresów względnych na absolutne/ - program nr 51.

Uruchomienie minisystemu następuje po włączeniu go do sieci i wyzerowaniu klawiszem SEL. Program roboczy może być wczytany z pamięci kasetowej - po naciśnięciu klawisza funkcyjnego L i wybraniu dwuznakowego numeru programu, lub z pamięci na dyskach elastycznych - po naciśnięciu klawisza funkcyjnego LK , wybraniu nazwy programu /do 10 znaków/ i naciśnięciu klawisza funkcyjnego A . Prawidłowe wczytanie programu jest sygnalizowane wydrukowaniem za nazwą programu znaków "OK". Uruchomienie programu następuje po naciśnięciu klawisza funkcyjnego A .

INSTRUKCJE

Instrukcje, realizowane w procesorze

TIM/254. Opóźnienie

Format:

Kod op.	Operand
254	———

1	1	1	1	1	1	1	0

Realizacja: Rozkaz powoduje zawieszenie pracy systemu na czas 1 ms.

LDR/232. Ustawienie rejestru R1

Format:

Kod op.	Operand
232	ustawiany kod

1	1	1	0	1	0	0	0	x	x	x	x	x	x	x	x

Realizacja: Kod, zawarty w drugim bajcie rozkazu, jest ładowany do rejestru R1. Poprzednia zawartość rejestru R1 jest zacierana.

przed operacją

R1:

087

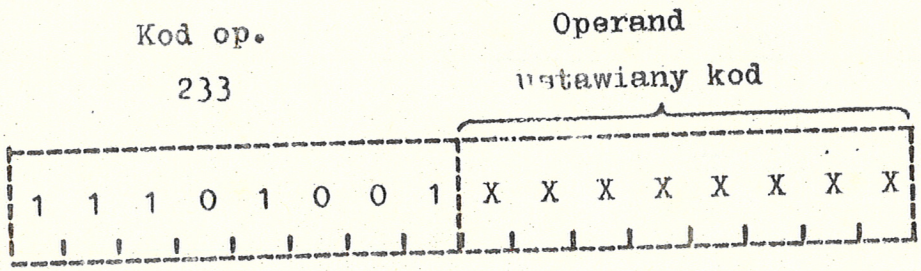
Po wykonaniu rozkazu LDR; 245:

R1:

245

LDV/233. Ustawienie rejestru V

Format:

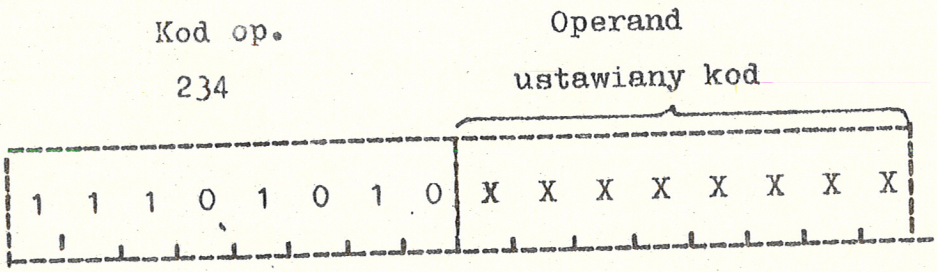


Realizacja: Kod, zawarty w drugim bajcie rozkazu, jest ładowany do rejestru V. Poprzednia zawartość rejestru V jest zacierana.

Przykład - analogiczny, jak dla rozkazu LDR.

LDZ/234. Ustawienie rejestru Z

Format:



Realizacja: Kod, zawarty w drugim bajcie rozkazu, jest ładowany do rejestru Z.
Poprzednia zawartość rejestru /licznika/ Z jest zacierana.

Przykład - analogicznie, jak dla rozkazu LDR.

INZ/235. Zwiększenie zawartości licznika Z o 1.

Format:

Kod op.	operand
235	———

1	1	1	0	1	0	1	1

Realizacja: Zawartość licznika Z jest zwiększona o 1. Ponieważ licznik Z jest licznikiem 8-bitowym, więc może liczyć do 255 /Po dodaniu 1 do liczby 255 w liczniku Z będzie kod 000/.

Przykład:

Przed operacją: 047

\+ 1

Po wykonaniu rozkazu INZ: 048

MZR/236. Poniesienie zawartości licznika Z do rejestru R1

Format:

Kod op.	Operand
236	———

1	1	1	0	1	1	0	0

Realizacja: Bajt informacji, znajdujący się w liczniku zostaje przeniesiony do rejestru R1. Poprzednia zawartość rejestru R1 jest zacierana, natomiast zawartość licznika Z nie ulega zmianie

Przykład:

Przed operacją:

Z: 015

R1: 219

Po wykonaniu rozkazu MZR:

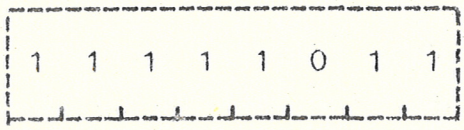
Z: 015

R1: 015

ADS/251. Ustawienie sumatora w stan dodawania

Format:

Kod op.	Operand
251	———

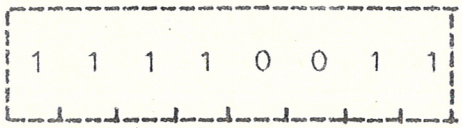


Realizacja: Rozkaz powoduje wyzerowanie bitu przeniesienia międzypozycyjnego oraz ustawienie sumatora w stan dodawania.

SBS/243. Ustawienie sumatora w stan odejmowania

Format:

Kod op.	Operand
243	———



Realizacja: Rozkaz powoduje wyzerowanie bitu przeniesienia międzypozycyjnego oraz ustawienie sumatora w stan odejmowania. Odejmowany będzie kod, zawarty w rejestrze B od kodu, zawartego w rejestrze A.

PRA/247. Przeniesienie młodszej tetrady z rejestru R1 do rejestru A sumatora

Format:

Kod op.	Operand
247	

1	1	1	1	0	1	1	1

Realizacja: Cztery młodsze bity zawartości rejestru R1 są inwertowane i w zanegowanej formie ładowane do rejestru A sumatora

Przykład:

przed operacją:			Po wykonaniu rozkazu PRA		
R1:	200	Kod cyfry 7	R1:	200	kod cyfry
A:	0 0 0 0	kod "0"	A:	0 1 1 1	kod "7"

PRB/255. Przeniesienie młodszej tetrady z rejestru R1 do rejestru B sumatora

Format:

Kod op.	Operand
255	

1	1	1	1	1	1	1	1

Realizacja: Cztery młodsze bity zawartości rejestru R1 są inwertowane i w zanegowanej formie ładowane do rejestru B sumatora.

Przykład - analogiczny, jak dla rozkazu PRA.

PKR/253. Przeniesienie wyniku operacji arytmetycznej do rejestru R1

Format:

Kod op.	Operand
253	———

1	1	1	1	1	1	0	1

Realizacja: Czterobitowy wynik operacji arytmetycznej, wykonanej w sumatorze, jest inwestowany, uzupełniany starszą tetradą postaci 1100 i ładowany do rejestru R1.

Przykład:

Przed operacją:

Sumator: | 1 | 0 | 0 | 1 |

R1: | 188 |

Po wykonaniu rozkazu PKR:

Sumator: | 1 | 0 | 0 | 1 | /kod "9"/

R1: | 198 | kod cyfry 9

AGB/228. Skok pod warunkiem A > B.

Format:

Kod op.	Operand adres	
228	Adres strony	Adres komórki

1	1	1	0	0	1	0	0	Adres strony	Adres komórki

Realizacja: Porównywane są czterobitowe kody binarne cyfr 0 ÷ 9, zawarte w rejestrach A i B sumatora. Jeśli jest spełniony warunek A > B, to rozkazowy licznik adresowy /RLA/ jest ustawiany na adres strony, zawarty w drugim bajcie rozkazu AGB, i adres komórki, zawarty w trzecim bajcie. Program jest kontynuowany od instrukcji, znajdującej się pod tym adresem. Jeśli warunek nie jest spełniony, to program jest kontynuowany od kolejnej instrukcji za rozkazem AGB.

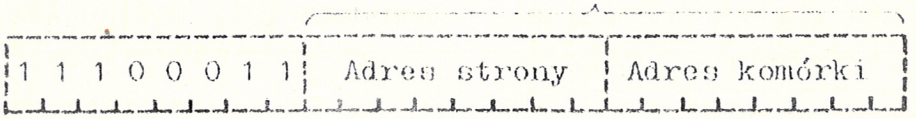
Przykład: Rejestry A i B zawierają kody 0111 i 0101. Badamy, czy możliwe jest wykonanie odejmowania:

AGB; SB	228; SB
JMP; IN	228; IN
SB: SBS	SB: 243
PKR	253
....	....
IN:	IN: zamieniamy kody w rejestrach A i B

ALB/227. Skok pod warunkiem A < B.

Format:

Kod op.	Operand
227	Adres



Realizacja: Porównywane są czterobitowe kody binarne cyfr 0 ÷ 9, zawarte w rejestrach A i B sumatora. Jeśli jest spełniony warunek A < B, to rozkazowy licznik adresowy jest ustawiany na adres strony, zawarty w drugim bajcie rozkazu ALB i adres komórki, zawarty w trzecim bajcie. Program jest kontynuowany od instrukcji, znajdującej się pod tym adresem. Jeśli warunek nie jest spełniony, to program jest kontynuowany od kolejnej instrukcji za rozkazem ALB.

Przykład: Analogicznie, jak dla rozkazu 228/AGB

REV/226. Skok przy równości danych w rejestrach R1 i V

Format:

Kod op.	Operand	
226	adres	
1 1 1 0 0 0 1 0	adres strony	adres komórki

Realizacja: Porównywane są bajty informacji, zawarte w rejestrach R1 i V. Jeśli informacje są identyczne, to rozkazowy licznik adresowy jest ustawiany na adres strony, zawarty w drugim bajcie rozkazu REV, i adres komórki, zawarty w trzecim bajcie. Program jest kontynuowany od instrukcji, znajdującej się pod tym adresem.

Jeśli informacje w rejestrach R1 i V są różne, to program jest kontynuowany od kolejnej instrukcji za rozkazem REV.

Przykład: Należy odnaleźć koniec bloku, zapisanego w pamięci od określonego adresu.

Sprawdzamy, czy kolejny odczytany bajt zawiera kod 255 /znak końca bloku/:

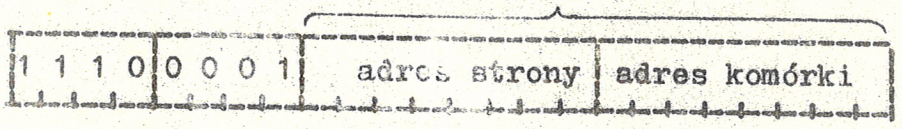
```
     ....
     BUD; adres
     BUF
     LDV; 255
ZN:  INB
     MPR
     REV; KC
     JMP; ZN
KC:  ....
```

```
     ...
     209; adres
     208
     233; 255
IN:  210
     221
     226; KC
     224; ZN
KC:  ....
```

AEB/225. Skok przy równości danych w rejestrach A i B sumatora

Format:

Kod op.	Operand	
225	adres	



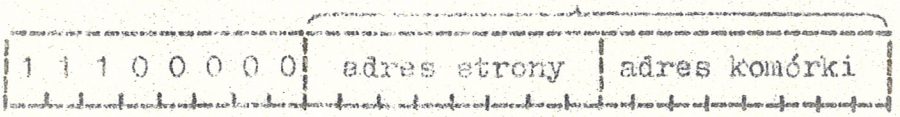
Realizacja: Porównywane są czterobitowe kody binarne cyfr 0 ÷ 9, zawarte w rejestrach A i B sumatora. Jeśli jest spełniony warunek A=B, to rozkazowy licznik adresowy jest ustawiany na adres strony, zawarty w drugim bajcie rozkazu A[illegible] i adres komórki, zawarty w trzecim bajcie. Program jest kontynuowany do instrukcji, znajdującej się [illegible]m adresem. Jeśli warunek nie jest spełniony, to program jest kontynuowany od kolejnej instrukcji za rozkazem AEB.

Przykład: analogiczny, jak w innych instrukcjach skoku warunkowego

JMK/224. Skok bezwarunkowy

Format:

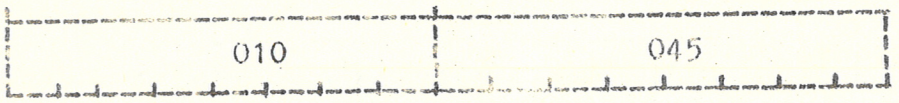
Kod op.	Operand	
224	adres	



Realizacja: Rozkazowy licznik adresowy jest ustawiany na adres, zawarty w drugim i trzecim bajcie rozkazu skoku. Program jest realizowany od instrukcji, znajdującej się pod tym adresem.

Przykłady:

1. Przed operacją:



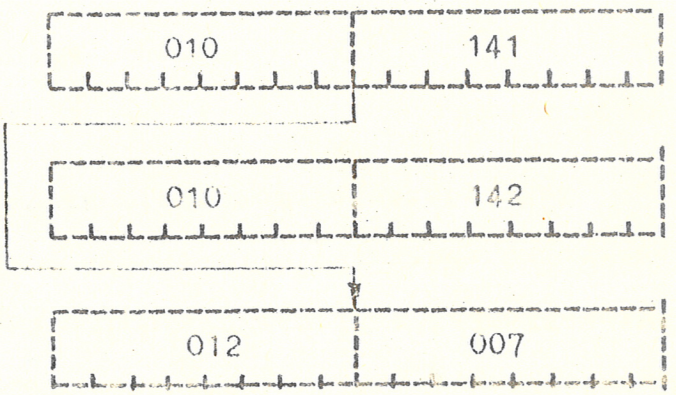
Po wykonaniu rozkazu JMP; 010.003:



2. W ciągu instrukcji:

adres pamięci	Etykieta Kod op. Operand
010. 141	E1: JMP; E3
.....	.....
012. 006	E2: LRC
012. 007	E3: TIM
.....	.....

rozkaz JMP spowoduje, że PC zamiast zwiększać swą zawartość 01, zostanie załadowany adresem, zawartym w drugim i trzecim bajcie tego rozkazu



CAL/229. Skok do podprogramu

Format:

Kod op.	Operand	
229	adres	
1 1 1 0 0 1 0 1	adres strony	adres komórki

Realizacja: Zawartość rozkazowego licznika adresowego zwiększona o 1 /tj. adres drugiego bajta rozkazu CAL/ zostaje zapamiętana w stosie. Równocześnie adresy, wcześniej zapisane w stosie, zostają przesunięte o 1 poziom wgłąb stosu. Rozkazowy licznik adresowy zostaje ustawiony na adres zawarty w drugim i trzecim bajcie rozkazu CAL. Następuje realizacja podprogramu od pierwszego jego rozkazu, znajdującego się pod adresem, wskazanym w rozkazie CAL. Powrót z programu realizuje rozkaz EXI/230.

Stos pozwala na rekursję podprogramów do piątego poziomu.

Przykład: Stan stosu:

Przed operacją:

STOS:

I	010.014
II	"
III	"
IV	"
V	"

Po wykonaniu rozkazu CAL:

011.008 CAL; AU

STOS:

I	011.009
II	010.014
III	"
IV	"
V	"

EXI/230. Powrót z podprogramu

Format:

Kod op.	Operand
230	———

1	1	1	0	0	0	1	0

Realizacja: Rozkaz powoduje załadowanie rozkazowego licznika adresowego PC adresem, zawartym w górnym, I poziomie stosu. Dzięki temu następuje powrót do programu w miejsce, skąd nastąpiło wywołanie podprogramu. Rozkaz EXI jest wiec rozkazem, kończącym pracę podprogramu.

Przykład: Przed wykonaniem rozkazu EXI

PC:

014	096

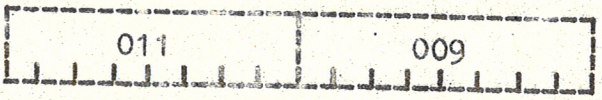
/adres rozkazu EXI/

STOS:

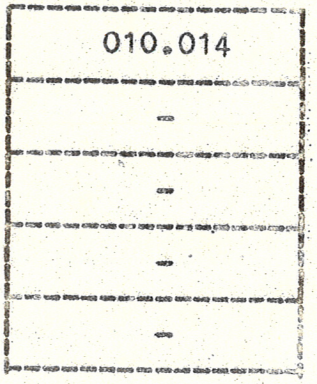
011.009
010.014
-
-
-

Po wykonaniu rozkazu EXI:

PC:



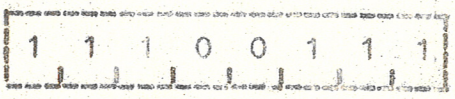
STOS:



INA/231. Modyfikacja adresu powrotu z podprogramu

Format:

Kod op.	Operand
231	———



Realizacja: Zawartość górnego, I poziomu stosu zostaje zwiększona o 1.

Uwaga:

1. Rozkazem CAL jest zapisywany w stosie adres drugiego bajta tego rozkazu. W kolejnej komórce pamięci znajduje się trzeci bajt rozkazu CAL. Obie te komórki należy minąć. W tym celu należy dwukrotnie użyć rozkazu INA, aby rozkaz EXI wystawiał na rozkazowy licznik adresowy PC właściwy adres powrotu z podprogramu.
2. Kilkakrotne użycie rozkazu INA pozwala organizować wyjście z podprogramu nie tylko na rozkaz kolejny za rozkazem CAL, ale również na dalsze rozkazy programu. Dzięki temu podprogramy mogą sprawdzać warunki, pełnić rolę złożonych instrukcji decyzyjnych itp.

Przykład:

Przed operacją:

STOS:

011.009
010.014
-
-
-

Po wykonaniu rozkazu INA:

STOS:

011.010
010.014
-
-
-

PAD/212. Wpisanie adresu do rejestru PADR

Format:

Kod op.	Operand	
212	adres	
1 1 0 1 0 1 0 0	adres strony	adres komórki

Realizacja: Rejestr PADR zostaje załatwiany adresem strony i adresem komórki, zawartymi w drugim i trzecim bajcie instrukcji.

Przykład: przed operacją:

PADR:

019	138

Po wykonaniu rozkazu PAD; 008.056:

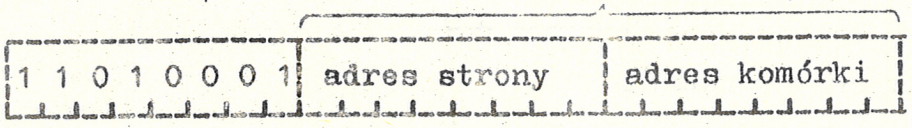
PADR:

008	056

BUD/209. Wpisanie adresu do rejestru BADR

Format:

Kod op.	Operand	
209	adres	



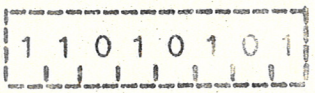
Realizacja: Rejestr BADR zostaje załadowany adresem, zawartym w drugim i trzecim bajcie instrukcji.

Przykład: analogiczny, jak dla instrukcji PAD.

INP/213. Zwiększenie zawartości rejestru PADR o 1.

Format:

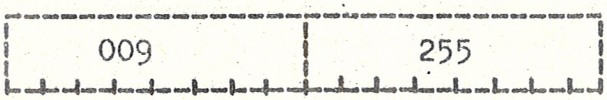
Kod op.	Operand
213	———



Realizacja: Zawartość rejestru PADR zostaje zwiększona o 1.

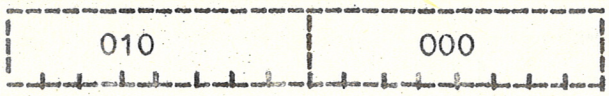
Przykład: przed operacją:

PADR:



Po wykonaniu rozkazu INP:

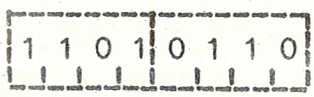
PADR:



DEP/214. Zmniejszenie zawartości rejestru PADR o 1

Format:

Kod op.	Operand
214	———



Realizacja: Zawartość rejestru PADR zostaje zmniejszona o 1.

Przykład: przed operacją:

PADR: | 010 | 000 |

po wykonaniu rozkazu DEP:

PADR: | 009 | 255 |

INB/210. Zwiększenie zawartości rejestru BADR o 1.

Format:

Kod op.	Operand
210	———

1	1	0	1	0	0	1	0

Realizacja: Zawartość rejestru BADR zostaje zwiększona o 1.

Przykład - analogicznie, jak dla rozkazu INP

DEB/211. Zmniejszenie zawartości rejestru BADR o 1.

Format:

Kod op.	Operand
211	———

1	1	0	1	0	0	1	1

Realizacja: Zawartość rejestru BADR zostaje zmniejszona o 1.

Przykład - analogicznie, jak dla rozkazu DEP.

BUF/208/. Adresowanie pamięci według zawartości rejestru BADR

Format:

Kod op.	Operand
1 1 0 1 0 0 0 0	———

Realizacja: Instrukcja BUF ustala reżim adresowania pamięci według zawartości rejestru BADR. Reżim adresowania pamięci poprzez rejestr BADR trwa do czasu pojawienia się rozkazu PAO.

PAO/215. Adresowanie pamięci według zawartości rejestru PADR

Format:

Kod op.	Operand
215	
1 1 0 1 0 1 1 1	———

Realizacja: Instrukcja PAO ustala reżim adresowania pamięci według zawartości rejestru PADR. Reżim adresowania pamięci poprzez rejestr PADR trwa do czasu pojawienia się rozkazu BUF.

MRP/222. Zapis w pamięci zawartości rejestru R1

Format:

Kod op.	Operand
222	
1 1 0 1 1 1 1 0	———

Realizacja: Zawartość rejestru R1 zostaje zapisana w pamięci operacyjnej pod adresem, ustawionym w rejestrze adresowym PADR - przy reżimie adresacji pamięci poprzez ten rejestr, lub BADR - przy reżimie adresacji poprzez BADR.

MPR/221. Czytanie z pamięci do rejestru R1

Format:

Kod op.	Operand
221	

1	1	0	1	1	1	0	1

Realizacja: Zawartość komórki, której adres jest ustawiony w rejestrze PADR /przy reżimie adresacji poprzez ten rejestr/ lub BADR /przy reżimie adresacji poprzez BADR/ jest odczytywana i zapisana w rejestrze R1.

Przykład: Odczytać zawartość komórki 009.252 i zapisać w komórce 012.007.

a/	.....	
	PAD; 009.252	212; 009.252
	PAO	215
	MPR	221
	BUD; 012.007	209; 012.007
	BUF	208
	MRP	222
	.....	.....
b/	.....	.....
	PAD; 009.252	212; 009.252
	BUD; 012.007	209; 012.007
	PAO	215
	MPR	221
	BUF	208
	MRP	222
	.....	.....

PRT/223. Wysłanie zawartości rejestru R1 do drukarki

Format:

Kod op.	Operand
223	

1	1	0	1	1	1	1	1

Realizacja: Kod, znajdujący się w rejestrze R1, zostaje wysłany do drukarki. Jeżeli jest to kod znaku drukarki, to znak ten zostaje wydrukowany w kolejnej pozycji w wierszu. Jeśli jest to kod funkcji drukarki, to zostaje ona wykonana. Wyjątek stanowi funkcja tabulacji poziomej, dla której należy wysłać również kod, określający adres tabulacji /jeśli tabulacja do x-ej pozycji, to należy wysłać kod 255-x/.

Jeśli po adresie tabulacji do drukarki zostanie wysłany kod znaku, to znak zostanie drukowany w pozycji x.

Jeśli po adresie tabulacji do drukarki zostanie wysłany kod funkcji drukarki, to wykonuje się tylko ta ostatnia funkcja /tabulacja pozioma nie jest wykonywana/.

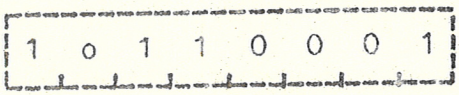
Kody funkcji i znaków drukarki podane są w tabeli T2. Długość drukowanego wiersza - 132 pozycje.

Uwaga: Po rozkazie PRT/223 nie należy umieszczać rozkazu przesyłania informacji /np. MZR itp./. Jeżeli jednak wymagają tego okoliczności, należy rozkazy PRT/223 i kolejny rozdzielić pustą instrukcją /kod 000/.

PAC/177. Uaktywnienie klawiatury

Format:

Kod op.	Operand
177	———

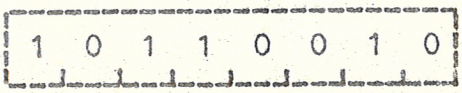


Relizacja: Procesor wysyła żądanie przesłania informacji z klawiatury. Rozpoczyna pracę układ sekwencyjny sterujący pracą klawiatury i następuje wyszukiwanie naciśniętego klawisza. Jeżeli układ stwierdzi naciśnięcie któregoś klawisza, następuje wystawienie kodu, odpowiadającego temu klawiszowi na szyny informacyjne. W przypadku, gdy żaden klawisz nie został naciśnięty, na szyny informacyjne zostaje wystawiony kod 255 /1111 1111/.

KAC/178. Koniec stanu aktywności klawiatury

Format:

Kod op.	Operand
178	———

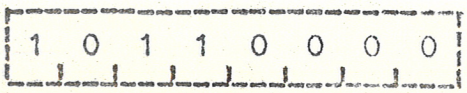


Realizacja: Rozkaz powoduje zablokowanie klawiatury. Zostaje wyłączony układ sekwencyjny, sterujący pracą klawiatury. Informacja z klawiatury nie jest przekazywana na szyny informacyjne.

RKL/176. Przeniesienie kodu naciśniętego klawisza do rejestru R1

Format:

Kod op.	Operand
176	———



Realizacja: Bajt informacji, przygotowany przez klawiaturę, zostaje załadowany do rejestru R1. Instrukcja RKL, użyta po instrukcji PAC nie blokuje pracy urządzenia. Użyta samodzielnie /bez PAC/ blokuje pracę urządzenia do momentu naciśnięcia jakiegokolwiek klawisza. Kod naciśniętego klawisza zostanie załadowany do rejestru R1 przy podwójnym użyciu instrukcji /RKL, RKL/.

Przykłady:

1.

```
.........  |  .........
    PAC    |     177
    RKL    |     176
    KAC    |     178
.........  |  .........
```

Jeśli został naciśnięty jakiś klawisz, to jego kod zostanie zapisany w rejestrze R1. Jeżeli żaden klawisz nie był naciśnięty, w R1 będzie zapisany kod 255. Poprzednia zawartość rejestru R1 zostanie zatarta. Podany ciąg rozkazów realizuje tzw. reżim podglądania klawiatury.

2.

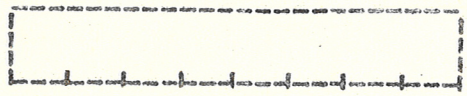
```
.......  |  .........
   RKL   |     176
   RKL   |     176
.......  |  .......
```

Urządzenie czeka na naciśnięcie jakiegokolwiek klawisza. Kod naciśniętego klawisza jest zapamiętywany w R1. Poprzednia zawartość R1 zostaje zatarta.

LON N. Włączenie lampki indukcyjnej o numerze N /N=1,2, ..., 12/.

Format:

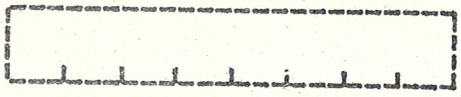
Kod op.	Operand	Lampka nr
180	——	1
182	——	2
184	——	3
186	——	4
188	——	5
190	——	6
160	——	7
162	——	8
164	——	9
166	——	10
168	——	11
170	——	12



LOF N. Wyłączenie lampki indukcyjnej o numerze N /N=1,2,...12/.

Format:

Kod op.	Operand	Lampka nr
181	——	1
183	——	2
185	——	3
187	——	4
189	——	5
191	——	6
161	——	7
163	——	8
165	——	9
167	——	10
169	——	11
171	——	12



CLR/207. Liczenie bajtu LRC z zawartości rejestru R1 i licznika LRC

Format:

Kod op.	Operand
207	———

1	1	0	0	1	1	1	1

Realizacja: Liczona jest suma modulo 2 /znak kontrolny LRC/ zawartości rejestru R1 i licznika LRC. Wynik operacji jest zapamiętywany w liczniku LRC.

Uwaga: Rozkaz REK/134 powoduje ustawienie w liczniku LRC kodu 000 /0000 0000/. Rozkaz CLR spowoduje, że w liczniku LRC utworzona zostanie inwersja kodu, zawartego w rejestrze R1,

Przykład: przed operacją:

LRC: | X | X | X | X | X | X | X | X |

R1: | 0 | 0 | 1 | 1 | 1 | 0 | 1 | 0 |

po wykonaniu rozkazu REK:

LRC: | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

R1: | 0 | 0 | 1 | 1 | 1 | 0 | 1 | 0 |

po wykonaniu rozkazu CLR:

LRC: | 1 | 1 | 0 | 0 | 0 | 1 | 0 | 1 |

R1: | 0 | 0 | 1 | 1 | 1 | 0 | 1 | 0 |

Rozkaz LRC przeniesie zawartość licznika LRC do rejestru R1:

LRC: | 1 | 1 | 0 | 0 | 0 | 1 | 0 | 1 |

R1: | 1 | 1 | 0 | 0 | 0 | 1 | 0 | 1 |

LRC/128. Przeniesienie zawartości licznika LRC do rejestru R1

Format:

Kod op.	Operand
128	———

1	0	0	0	0	0	0	0

Realizacja: Zawartość licznika LRC zostaje przeniesiona do rejestru R1. Rozkaz zaciera poprzednią zawartość rejestru R1, natomiast nie zmienia zawartości licznika LRC.

Instrukcje, dotyczące pamięci kasetowej PK-1

Zapis informacji w pamięci kasetowej odbywa się zgodnie ze standartem ECMA-34.

REK/134. Zerowanie pamięci kasetowej

Format:

Kod op.	Operand
134	———

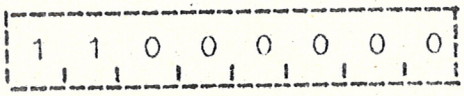
1	0	0	0	0	1	1	0

Realizacja: Następuje wyzerowanie licznika LRC oraz rejestrów pamięci kasetowej. Pamięć kasetowa przechodzi w stan niegotowości /do czasu pojawienia się rozkazu TMB/194/. Stan rezerwacji pamięci kasetowej nie ulega zmianie.

BTR/192. Rezerwacja pamięci kasetowej

Format:

Kod op.	Operand
192	———

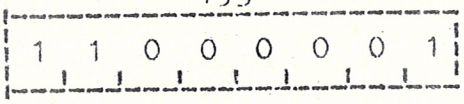


Realizacja: Jeżeli do pamięci kasetowej została załadowana kaseta, to karetka z kasetą zostaje zablokowana w położeniu roboczym. Z staje uruchomiony silnik napędowy pamięci kasetowej oraz zapala się lampka indykacyjna "LOCK", sygnalizująca stan rezerwacji pamięci kasetowej.

STR/193. Koniec rezerwacji pamięci kasetowej

Format:

Kod op.	Operand
193	———

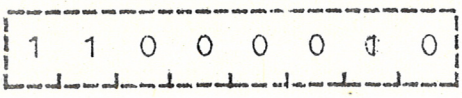


Realizacja: Następuje wyłączenie silnika napędowego pamięci kasetowej. Gaśnie lampka "LOCK", sygnalizująca stan rezerwacji PK, oraz zostaje zdjęta blokada karetki. Operator może wyjąć kasetę.

TMB/194. Podłączenie logiczne pamięci kasetowej

Format:

Kod op.	Operand
194	———

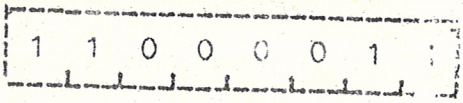


Realizacja: Następuje logiczne podłączenie pamięci kasetowej, tj. odblokowanie układów interfejsu pamięci kasetowej. Po tym rozkazie możliwe są operacje odczytu i zapisu informacji, o ile pamięć kasetowa osiągnie stan gotowości /270 ms/

TME/195. Odłączenie logiczne pamięci kasetowej

Format:

Kod op.	Operand
195	———

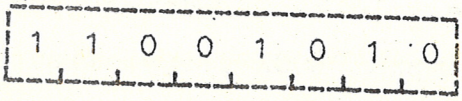


Realizacja: Następuje zablokowanie układów interfejsu pamięci kasetowej, czyli jej logiczne odłączenie.

DRN/202. Pozycjonowanie taśmy kasetowej

Format:

Kod op.	Operand
202	———

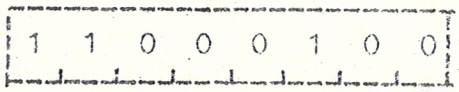


Realizacja: Następuje pozycjonowanie taśmy w kasecie do fizycznego początku, tzn. cała taśma zostaje nawinięta na lewą szpulę.
W czasie pozycjonowania taśmy pamięć kasetowa znajduje się w stanie niegotowości /nie przyjmowane są żadne rozkazy, nawet rozkaz zerowania systemu/.

BEP/196. Początek ruchu taśmy do przodu

Format:

Kod op.	Operand
196	———

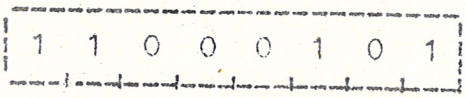


Realizacja: Rozpoczyna się ruch taśmy do przodu /na prawą szpulę/ z szybkością roboczą. Rozkaz jest realizowany tylko z wypełku, gdy pamięć kasetowa znajduje się w stanie gotowości. Sam rozkaz BEP nie zmienia stanu gotowości pamięci kasetowej.

ENP/197. Koniec ruchu taśmy do przodu

Format:

Kod op.	Operand
197	

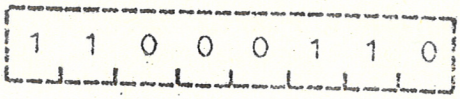


Realizacja: Rozkaz kończy ruch taśmy do przodu, rozpoczęty rozkazem BEP. Stan gotowości pamięci kasetowej nie ulega zmianie.

BEB/198. Początek ruchu taśmy do tyłu

Format

Kod op.	Operand
198	

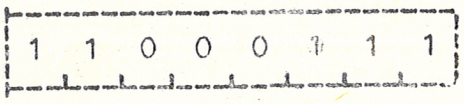


Realizacja: Rozpoczyna się ruch taśmy do tyłu z szybkością roboczą. Rozkaz jest realizowany tylko w wypadku, gdy pamięć kasetowa znajduje się w stanie gotowości. Sam rozkaz BEB nie zmienia stanu gotowości pamięci kasetowej.

ENB/199. Koniec ruchu taśmy do tyłu

Format:

Kod op.	Operand
199	———

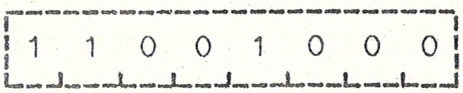


Realizacja: Rozkaz kończy ruch taśmy do tyłu, rozpoczęty rozkazek BEB. Stan gotowości pamięci kasetowej nie ulega zmianie.

BES/200. Początek ruchu przyśpieszonego

Format:

Kod op.	Operand
200	———

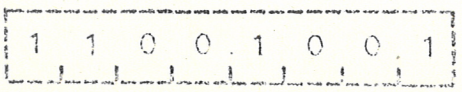


Realizacja: Wykonanie rozkazu BES równoznaczne jest z przyśpieszeniem ruchu /wprzód lub wtył/, aktualnie wykonywanego przez taśmę w pamięci kasetowej. Rozkaz nie zmienia stanu gotowości pamięci kasetowej.

ENS/201. Koniec ruchu przyśpieszonego

Format:

Kod op.	Operand
201	———

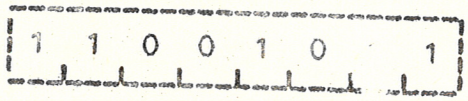


Realizacja: Następuje powrót z szybkości podwyższonej rozkazem BES do szybkości roboczej ruchu taśmy w pamięci kasetowej. Rozkaz nie zmienia stanu gotowości pamięci PK.

BEW/203. Zezwolenie na zapis

Format:

Kod op. Operand



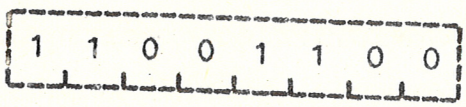
Realizacja: Zostają uruchomione układy kasowania i zapisu informacji w pamięci kasetowej. Następuje sprawdzenie obecności wkładek, zezwalających na zapis informacji na danej kasetce. Jeżeli kasetka posiada wkładki, następuje zapis informacji na taśmie. Jeśli kasetka nie ma wkładek /została w ten sposób zabezpieczona przed skasowaniem/, zapis nie jest wykonywany, a pamięć kasetowa przechodzi w stan niegotowości.

ENW/204. Koniec zezwolenia na zapis

Format:

Kod op. Operand

204

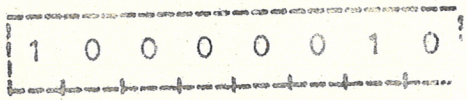


Realizacja: Układy kasowania i zapisu informacji w pamięci kasetowej zostają wyłączone. Aby nie dopuścić do powstawania na taśmie miejsc nieskasowanych, należy rozkaz ENW podawać, gdy ustanie ruch roboczy taśmy, tzn. po rozkazie ENP /lub ENB/.

MKR/130. Przesłanie zawartości rejestru R1 do pamięci kasetowej

Format:

Kod op.	Operand
130	

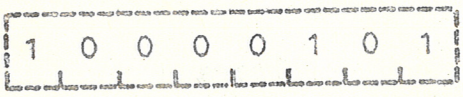


Realizacja: Bajt informacji, znajdujący się w rejestrze R1, zostaje przesłany do układu zapisu w pamięci kasetowej. Jeśli taśma jest w ruchu z szybkością roboczą, pamięć kasetowa otrzymała zezwolenie na zapis /rozkazem BEW/ i nie sygnalizuje stanu niegotowości, to przesłany bajt zapisywany jest na taśmie magnetycznej. Sam rozkaz MRK nie zmienia zawartości rejestru R1 i stanu gotowości pamięci kasetowej.

END/133. Koniec informacji do zapisu w pamięci kasetowej

Format:

Kod op.	Operand
133	—



Realizacja: Rozkaz ten informuje układ zapisu pamięci kasetowej o tym, że poprzedni zapisany na taśmie bajt informacji był ostatnim bajtem bloku.
Rozkaz nie zmienia stanu gotowości pamięci kasetowej.

BER/205 Odczyt bloku danych z pamięci kasetowej

Format:

Kod op.	Operand
205	

1	1	0	0	1	1	0	1

Realizacja: Następuje włączenie układów odczytu pamięci kasetowej oraz automatyczne rozpoczęcie ruchu roboczego taśmy do przodu. Ruch roboczy taśmy oraz odczyt informacji trwa do czasu wykrycia przerwy międzyblokowej lub odcinka taśmy długości ok. 1,5 mm, na którym nie ma śladów zapisu informacji. Rozkaz nie zmienia stanu gotowości pamięci kasetowej.

MKR/129. Ładowanie rejestru R1 bajtem, odczytanym z pamięci kasetowej

Format:

Kod op.	Operand
129	

1	0	0	0	0	0	0	1

Realizacja: Kolejny bajt informacji, odczytany z pamięci kasetowej, zostaje załadowany do rejestru R1. Moment wykonania tego rozkazu musi nastąpić przed upływem 127 ms od chwili, gdy piąty bit statusu szczegółowego przyjmie wartość 1.

STA/131. Status pamięci kasetowej

Format:

Kod op.	Operand
131	

1	0	0	0	0	0	1	1

Realizacja: Słowo statusu postaci:

7							0
1	1	1	1	1	1	X	X

- Nie było rozkazu BEW/203
  - 1 - pamięć niegotowa
  - 0 - pamięć gotowa
- Realizowany jest rozkaz BEW/203
  - 1 - brak wkładki zezwalającej na zapis
  - 0 - zapis dozwolony, wkładka jest
- 0 - czujnik początku taśmy /STT/ustawiony jest na przezroczystej rozbiegówce lub na otworze BOT albo EOT
- 1 - czujnik ustawiony jest na ścieżce magnetycznej

jest ładowane do rejestru R1.

Przykład: Należy sprawdzić, czy kasetka, załadowana do pamięci kasetowej, posiada wkładki zezwalające na zapis informacji na kasetce:

---	---	} - do tego momentu taśma została przygotowana i była w stanie gotowości
BEW	203	
STA	131	
LDV;255	233;255	
REV;DR	226;DR	
---	---	
DR:	DR:	drukowanie informacji, że kasetka nie posiada wkładek

STS/132 Status szczegółowy pamięci kasetowej

Format:

Kod op.	Operand
132	

1	0	0	0	0	1	0	0

Realizacja: Do rejestru R1 jest ładowane słowo statusu szczegółowego pamięci kasetowej postaci:

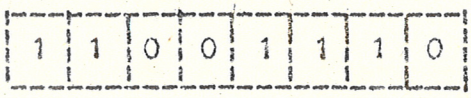
1	1	X	X	1	X	1	1

- 0 - jeśli nie jest realizowany rozkaz BER/205 lub podczas jego realizacji przez okres ok.8 ms nie znaleziono na taśmie żadnej informacji
- 1 - jeśli jest realizowany rozkaz BER/205 i odczyt informacji przechodzi prawidłowo
- 1 - pamięć realizuje zapis informacji, i żąda przesłania w czasie nie większym niż 123 ms, następnego bajta danych do zapisu lub zakończenia operacji zapisu rozkazem END/133
- 0 - pamięć nie realizuje zapisu lub jest w trakcie zapisu kolejnego bajtu danych
- 0 - jeśli pamięć nie realizuje odczytu lub czyta, ale nie odczytała pełnego bajtu
- 1 - odczytany bajt jest gotowy do odebrania; należy go załadować do R1 rozkazem MKR przed upływem 12 s.

SWT/206. Przełączenia pamięci kasetowych

Format:

Kod op.	Operand
206	———



Realizacja: W wersji systemu MERA-100 z dwiema pamięciami kasetowymi - następuje przełączenie pamięci kasetowych, tzn. pamięć, która ostatnio nie współpracowała z procesorem, zostaje wybrana /dołączona/. Równocześnie zostaje odłączona pamięć która ostatnio była wybrana. Kolejne użycie rozkazu spowoduje ponowne przełączenie pamięci.

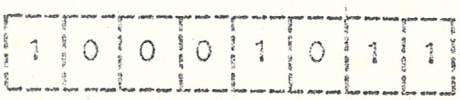
Uwaga: Wyzerowanie systemu /klawiszem SEL/ powoduje automatyczne wybranie pamięci kasetowej nr 1.

Rozkazy, dotyczące pracy monitora ekranowego VDU/12x40/ lub CRT/24x80/

INT/139. Rozpoczęcie pracy monitora ekranowego

Format:

Kod op.	Operand
139	———



Realizacja: Następuje całkowite wyzerowanie monitora ekranowego i przygotowanie go do pracy. Zalecane jest użycie tego rozkazu na początku pracy monitora ekranowego.

RDP/140. Przeniesienie zawartości rejestru R1 na monitor ekranowy

Format:

Kod op.	Operand
140	

1	0	0	0	1	1	0	0

Realizacja: Bajt informacji, znajdujący się w rejestrze R1, zostaje przeniesiony do bufora monitora ekranowego. Jeśli jest to kod znaku, to znak ten zostanie wyświetlony na ekranie, jeśli zaś kod funkcji, to dana funkcja zostanie przez monitor wykonana. Jeśli przeniesiony kod nie odpowiada żadnemu znakowi aniżadnej funkcji monitora, to jest on ignorowany /nie zostaje zapisany w buforze monitora/. Zawartość rejestru R1 nie ulega zmianie. Kody znaków monitora podane są w tab.T2.

Uwaga: Podanie kodu znaku, zmniejszonego o 128 spowoduje, że znak będzie wyświetlany na przemian z kursorem /z częstotliwością ok. 4Hz/.

Funkcje dotyczące kursora:

BS/247 - kursor w lewo o 1 pozycję,
HT/246 - kursor w prawo o 1 pozycję,
LF/245 - kursor w dół o 1 pozycję,
VT/244 - kursor w górę o 1 pozycję,
FF/243 - zerowanie ekranu/kursor wraca do początku ekranu/,
CR/242 - kursor do początku wiersza,
SO/241 - kursor do początku ekranu,
CI/240 - wstawienie/dodanie/ znaku,
CD/239 - anulowanie/wycięcie/ znaku

DC1/238 - tabulacja pozioma[1/],
DC2/237 - włączenie /zapalenie/ kursora,
DC3/236 - tabulacja pionowa[2/],
DC4/235 - wyłączenie /zgaszenie kursowa/,
LI/233 - wstawienie /dodanie/ linii,
LD/232 - anulowanie /wyoięcie/ linii.

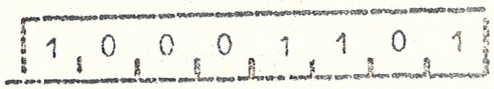
Przy zapełnieniu jednego wiersza przejście na początek następnego dokonuje się automatycznie. Podobnie, po zapełnieniu ostatniego wiersza ekranu, następuje automatyczne przejście na początek ekranu.

PDP/141. Rozpoczęcie procesu czytania zawartości ekranu monitora

Format:

Kod op. Operand

141



Realizacja: Rozkaz gasi ekran i przygotowuje bufor monitora do odczytu jego zawartości. Po wykonaniu tego rozkazu, a przed podaniem pierwszych rozkazów odczytu zawartości bufora należy wprowadzić opróżnienie około 100 ms, lub podać rozkaz INT/139.

1/ Po podaniu kodu DC1 kursor znika. Należy następnie podać negację kodu liczby 0÷39, aby uzyskać zapalenie się kursora w odpowiedniej pozycji w wierszu. Kod 255 odpowiada więc pierwszej pozycji w wierszu, zaś kod 216 - ostatniej, czterdziestej pozycji.

2/ Po podaniu kodu DC3 kursor znika. Należy następnie podać negację kodu liczby 0÷11, aby uzyskać pojawienie się kursora w odpowiedniej linii. Kod 255 odpowiada więc pierwszej linii, a kod 244 - ostatniej, dwunastej linii.

Uwaga: powyższe odnoszą się również do monitora CRT z tą różnicą, że po kodzie DC1 można podać negację kodu liczby 0÷79, a po kodzie DC3 negację kodu 0÷23.

DPX/142. Wybranie bajta informacji z bufora monitora ekranowego

Format:

Kod op.	Operand
142	———

1	0	0	0	1	1	1	0

Realizacja: Po uprzednim podaniu rozkazu PDP/141, rozkaz DPX powoduje wybranie z bufora monitora bajtu informacji. Kolejne rozkazy DPX powodują wybranie kolejnych bajtów /z kolejnych pozycji ekranu/. Miejsca puste na ekranie czytane są jako spacje /kod 233/.

Uwaga: W procesie czytania ekranu musi być wyczytane z pamięci monitora 512 bajtów, z czego 480 odpowiada znakom, wyświetlanym przedtem na ekranie. Niepełne wyczytanie pamięci monitora powoduje zakłócenie pracy monitora, które można usunąć tylko za pomocą rozkazu INT. Czytanie rozpoczyna się odpoczątku ekranu. Po wyczytaniu zawartości pierwszego wiersza czytana jest zawartość drugiego itd. Rozkaz nie zmienia zawartości rejestru R1.

MXR/138. Załadowanie do rejestru R1 bajta informacji, wybranego z pamięci monitora

Format:

Kod op.	Operand
138	———

1	0	0	0	1	0	1	0

Realizacja: Bajt informacji, wybrany z pamięci monitora, zostaje zapamiętany w rejestrze R1. Rozkaz ten powinien występować zawsze po rozkazie DPX/142. Po zapisaniu zawartości rejestru R1 w pamięci operacyjnej zwykle następuje kolejna para rozkazów DPX i MXR. Taki cykl powinien być powtórzony 512 razy.

Wniosek: Zapis bajtu informacji, pobieranego z bufora monitora, w rejestrze R1, następuje w dwóch etapach:

DPX
MXR
...

KDP/143. Zakończenie procesu czytania zawartości ekranu monitora

Format:

Kod op.	Operand
143	

1	0	0	0	1	1	1	1

Realizacja: Rozkaz kończy stan gotowości monitora do wydawania informacji i przygotowuje go do wpisywania nowych danych.

Uwaga: Zaleca się podawać ten rozkaz na zakończenie każdego procesu czytania monitora. Na ekranie pozostaje zawartość, będąca tam przed rozpoczęciem procesu czytania pamięci monitora. Znika tylko kursor. W rzeczywistości kursor znajduje się na początku ekranu. Przy pisaniu na ekran znak w pierwszej pozycji ekranu nie ulega zmianie, zaś kursor pojawia się w następnej pozycji i dalej proces pisania odbywa się normalnie. Dostęp do pierwszej pozycji ekranu można uzyskać przez podanie odpowiedniej funkcji, przemieszczającej kursor na początek ekranu /patrz tabela ..../ Jeżeli nie zależy nam na poprzedniej zawartości ekranu, należy monitor wyzerować rozkazem INT.

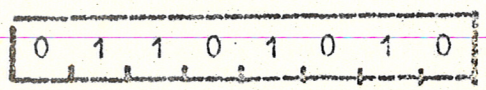
Rozkazy, dotyczące pamięci na dyskach elastycznych /FD/

Organizacja danych na dysku i sposób zapisu są zgodne ze standartem IBM 3740.
Na dysku można zapisać 77 ścieżek /numery od 00 do 76/. Na każdej ścieżce znajduje się 26 sektorów o numerach od 01 do 26. W sektorze zapisywane jest 128 bajtów informacji /nie licząc znaków kontrolnych/. Na jednej stronie dysku elastycznego można więc zapisać 77x26x128 = 256 256 bajtów.

ZFD/106. Zerowanie pamięci na dyskach elastycznych

Format:

Kod op.	Operand
106	———

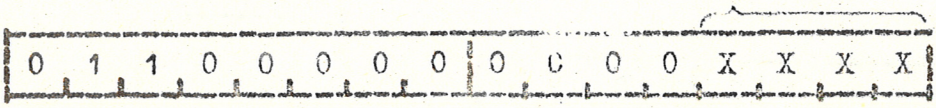


Realizacja: Następuje wyzerowanie rejestrów sterujących pracą pamięci na dyskach elastycznych.

PRUN/096. Rezerwacja drive'u

Format:

Kod op.	Operand
096	nr drive'u



Realizacja: Zostaje wybrana pamięć na dyskach elastycznych, oznaczona numerem, podanym w drugim bajcie rozkazu /numer pamięci, od 000 do 015/. Równocześnie następuje odłączenie tej pamięci, która została wybrana wcześniej.

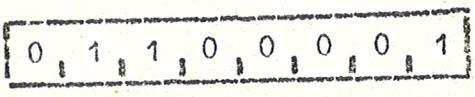
Przykład:

```
......
096; 000
......
......
......
096; 012      w tym momencie zostaje odłączona
                pamięć FD nr 0, a wybrana zostaje
                pamięć nr 12.
......
......
......
```

KRUN/097. Koniec rezerwacji drive'u

Format:

Kod op.	Operand
097	———

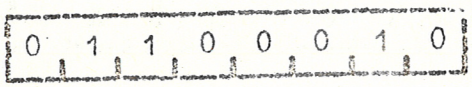


Realizacja: Rozkaz powoduje odłączenie aktualnie wybranej jednostki pamięci na dyskach elastycznych /drive'u/.

DLW/098. Wybranie dysku lewego

Format:

Kod op.	Operand
098	———

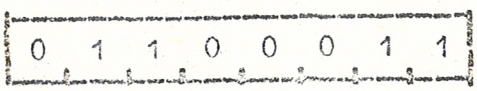


Realizacja: Następuje wybranie lewego dysku w wybranej jednostce pamięci na dyskach elastycznych. Urządzenie zostaje zabezpieczone przed ingerencją z zewnątrz - - włączona zostaje blokada drzwiczek drive'u.

DPR/099. Wybranie dysku prawego

Format:

Kod op.	Operand
099	———

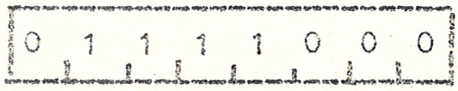


Realizacja: Następuje wybranie prawego dysku w wybranej jednostce pamięci na dyskach elastycznych. Urządzenie zostaje zabezpieczone przed otwarciem drzwiczek.

SFD/120. Status pamięci na dyskach elastycznych.

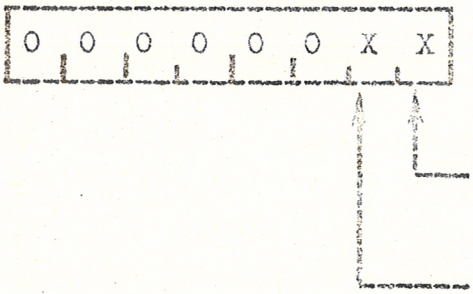
Format:

Kod op.	Operand
120	-



Realizacja: Bajt statusu pamięci na dyskach elastycznych zostaje umieszczony w rejestrze R1, zacierając przy tym poprzednią zawartość rejestru R1.

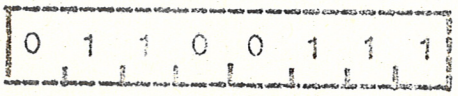
Słowo statusu:



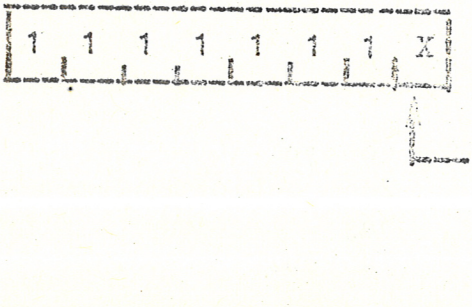
STFD1/103. Pierwszy status pamięci FD.

Format:

Kod op.	Operand
103	-



Realizacja: Do rejestru R1 jest ładowany bajt statusu postaci:

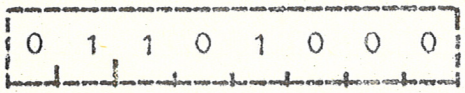


Poprzednia zawartość rejestru R1 zostaje zatarta.

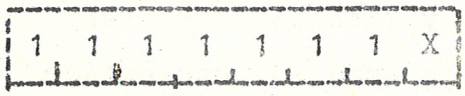
STFD2/104. Drugi status pamięci F.D.

Format:

Kod op.	Operand
104	-



Realizacja: Do rejestru R1 jest ładowany bajt statusu postaci:



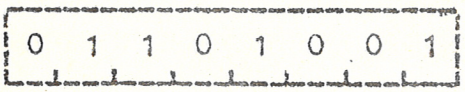
X — 0 - jeśli pamięć zakończyła wykonywanie ostatniego rozkazu

Poprzednia zawartość rejestru R1 zostaje zatarta.

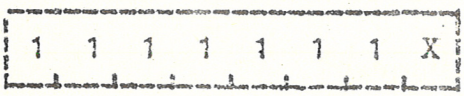
STED3/105. Trzeci status pamięci F.D.

Format:

Kod op.	Operand
105	-



Realizacja: Do rejestru R1 zostaje załadowany bajt statusu postaci:



X — Przy zapisie: 0 - jeśli pamięć żąda podania bajtu informacji do rejestru danych.

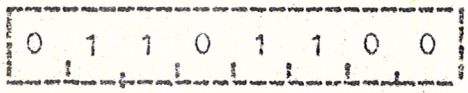
Przy odczycie: 0 - jeśli pamięć żąda odbioru odczytanego bajtu informacji z rejestru danych.

Poprzednia zawartość rejestru R1 zostaje zatarta.

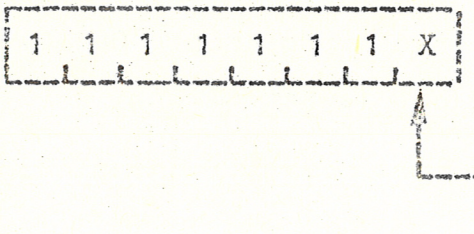
STFD4/108. Czwarty status pamięci F.D.

Format:

Kod op.	Operand
108	-



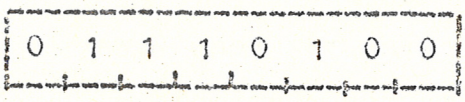
Realizacja: Do rejestru R1 zostaje załadowany bajt statusu postaci:



WROZ/116. Wpisanie rozkazu.

Format:

Kod op.	Operand
116	-

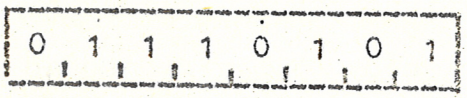


Realizacja: Zawartość rejestru R1 zostaje przeniesiona do układu sterującego pracą pamięci na dyskach elastycznych.

WNSC/117. Wpisanie numeru ścieżki.

Format:

Kod op.	Operand
117	-

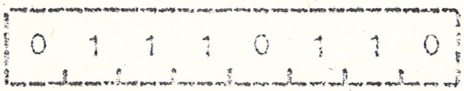


Realizacja: Zawartość rejestru R1 zostaje przepisana do rejestru numeru ścieżki w układzie sterującym pracą pamięci F.D. Numer ścieżki powinien leżeć w przedziale 000 - 076. Zawartość rejestru R1 pozostaje bez zmian.

WNSK/118. Wpisanie numeru sektora.

Format:

Kod op.	Operand
118	-

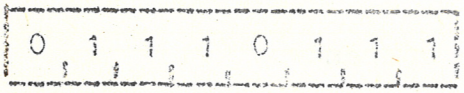


Realizacja: Zawartość rejestru R1 zostaje przepisana do rejestru numeru sektora pamięci na dyskach elastycznych. Numer sektora powinien leżeć w przedziale 001-026. Zawartość rejestru R1 pozostaje bez zmian.

WDAN/119. Wpisanie danych.

Format:

Kod op.	Operand

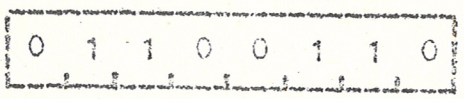


Realizacja: Zawartość rejestru R1 zostaje przepisana do rejestru danych pamięci na dyskach elastycznych. Zawartość rejestru R1 nie jest tym rozkazem zmieniana.

PZZ/102. Początek zezwolenia na zapis.

Format:

Kod op.	Operand
102	-

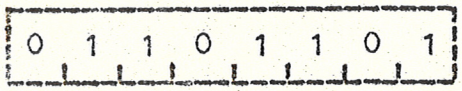


Realizacja: Rozkaz zezwala na uruchomienie głowicy zapisującej w wybranej jednostce pamięci F.D.

KZZ/109. Koniec zezwolenia na zapis.

Format:

Kod op.	Operand
109	-

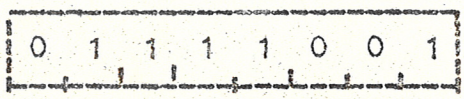


Realizacja: Rozkaz cofa zezwolenie na uruchomienie głowicy zapisującej w wybranej pamięci na dyskach elastycznych.

CNSC/121. Czytanie numeru ścieżki.

Format:

Kod op.	Operand
121	-

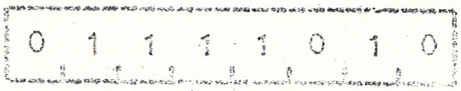


Realizacja: Zawartość rejestru numeru ścieżki pamięci na dyskach elastycznych zostaje przepisana do rejestru R1. Stan rejestru numeru ścieżki nie ulega zmianie, a poprzednia zawartość rejestru R1 jest zacierana.

CNSK/122. Czytanie numeru sektora.

Format:

Kod op.	Operand
122	-



Realizacja: Zawartość rejestru numeru sektora zostaje przepisana do rejestru R1, niszcząc poprzednią zawartość rejestru R1.

CDAN/123. Czytanie danych.

Format:

Kod op.	Operand
123	

0	1	1	1	1	0	1	1

Realizacja: Zawartość rejestru danych pamięci na dyskach elastycznych zostaje zapisana w rejestrze R1.

Rozkazy, dotyczące jednostki sterującej transmisją danych.

Minisystem może pracować w systemach teleprzetwarzania danych zarówno poprzez modem, jak i bez niego. Realizowana może być transmisja synchroniczna lub asynchroniczna, zgodnie ze standartem V 24 lub V 28, w systemach dwupunktowych, jak i wielopunktowych. W jednostce sterowania transmisją wykorzystano układ INTEL 8251.

ROZ/144. Wpisanie słowa sterującego transmisją.

Format:

Kod op.	Operand
144	-

1	0	0	1	0	0	0	0

Realizacja: Zawartość rejestru R1 zostaje przesłana do układu USART jednostki sterującej transmisją.
Rozkaz służy do programowania układu USART.

Przykład:

```
- - -          | - - -
L D R; X       | 232; X   - słowo sterujące /MODE/[1]
R O Z          | 144
L D R; X       | 232; X   - pierwszy znak SYN/[2]
R O Z          | 144
L D R; X       | 232; X   - drugi znak SYN/2/
R O Z          | 144
L D R; X       | 232; X   - słowo COMMAND[1]
R O Z          | 144
L D R; X       | 232; X   - słowo COMMAND[1]
R O Z          | 144
- - -          | - - -
```

1/ Opis znaczeń słów MODE, SYN i COMMAND znaleźć można w opisie układu INTEL 8251.

2/ Znaki SYN są używane tylko przy pracy synchronicznej /z jednym lub dwoma znakami synchronizującymi/.

Uwaga: Bajty MODE i COMMAND należy ustawiać w rejestrze R1 w wersji zanegowanej.

NAD/145. Wysłanie bajta informacji z rejestru R1 do USART-a.

Format:

Kod op.	Operand
145	-

1	0	0	1	0	0	0	1

Realizacja: Zanegowany bajt informacji z rejestru R1 zostaje przesłany do rejestru nadajnika układu USART. Zawartość rejestru R1 nie ulega zmianie.

Przykład:

R1: | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | Symbol "?" w kodzie ASCII

w wyniku wykonania rozkazu NAD

w rejestrze nadajnika będzie umieszczony kod:

NADAJNIK: | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | Symbol "?" w kodzie ISO

ODB/146. Odbiór danych /z układu USART do rejestru R1/

Format:

Kod op.	Operand
146	-

1	0	0	1	0	0	1	0

Realizacja: Negacja zawartości rejestru odbiornika /informacji przyjętej z linii/ zostaje przesłana do rejestru R1. Kod ISO zostaje więc przekształcony w kod ASCII.

Przykład: Z linii odebrano kod 0100 0001 /"A" w kodzie ISO/. Po wykonaniu rozkazu ODB w R1 będzie kod: 1011 1110 /"A" w kodzie ASCII/.

STU/147. Status układu USART

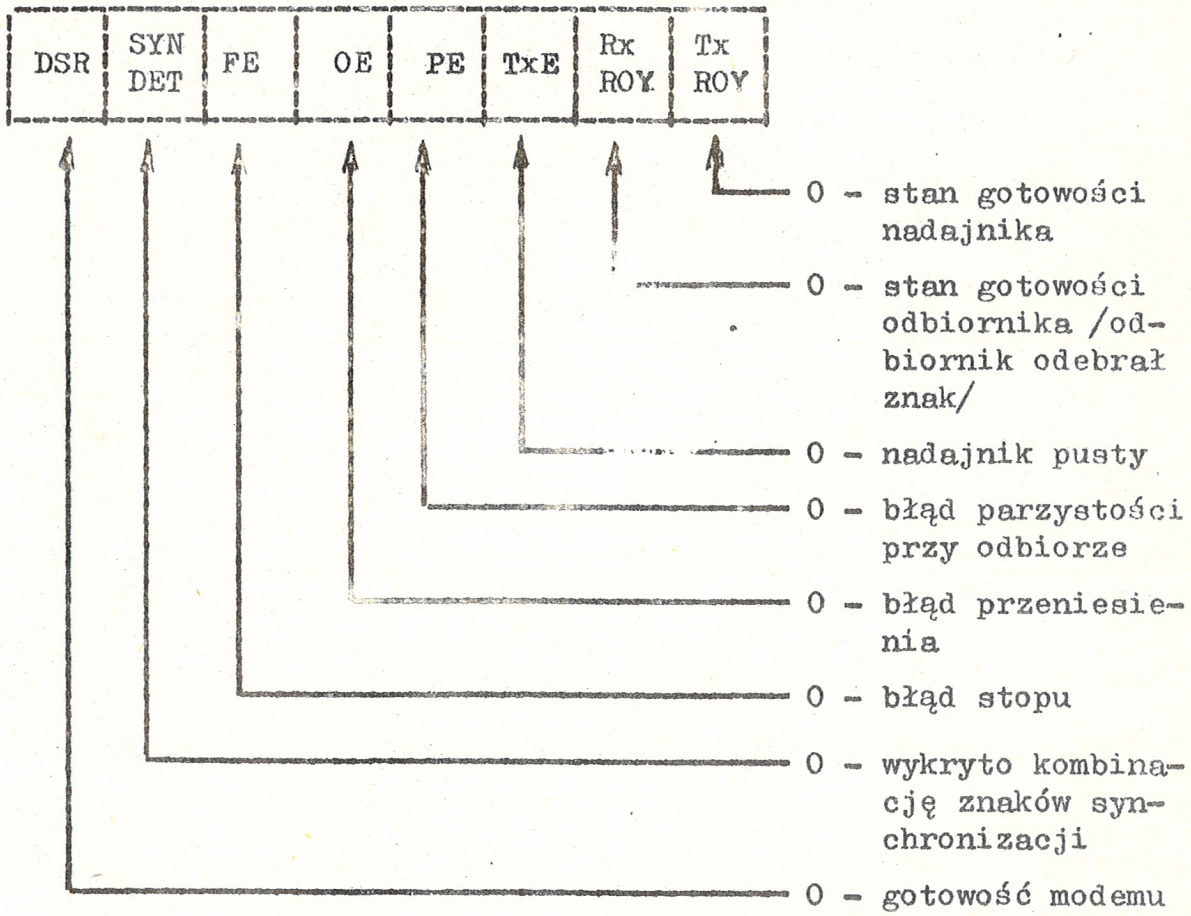
Format:

Kod op.	Operand
147	-

1	0	0	1	0	0	1	1

Realizacja: Słowo statusu układu USART zostaje załadowane do rejestru R1. Poprzednia zawartość rejestru R1 jest zacierana.

Słowo statusu USART-a po załadowaniu go do rejestru R1:



AR/148. Adres terminala.

Format:

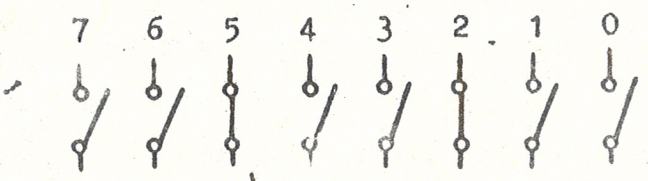
Kod op.	Operand
148	-

1	0	0	1	0	1	0	0

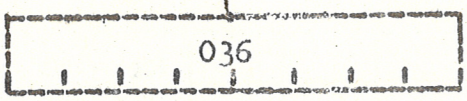
Realizacja: Bajt informacji o stanie kluczy adresowych naszego terminala zostaje załadowany do rejestru R1.
Rozkaz zaciera poprzednią zawartość rejestru R1.

Przykład:

Stan kluczy:



Kod, załadowany do rejestru R1:

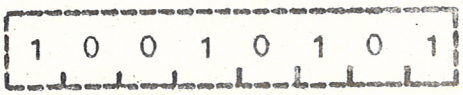


/0010 0100/

RS/149. Status linii.

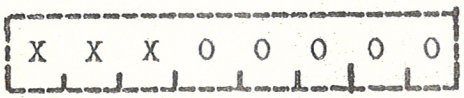
Format:

Kod op.	Operand
149	-



Realizacja: Informacja o stanie linii "109", "125" oraz o zmianie /zaniku/ stanu linii "110" interfejsu V-24 zapisywana jest w rejestrze R1.
Rozkaz zaciera poprzednią zawartość rejestru R1.

Format słowa statusu linii:



stan linii "109"

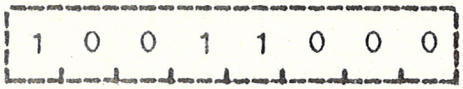
zmiana stanu linii "110":
1 - jeśli ze stanu "1" do "0"

stan linii "125".

CRS/152. Gaszenie szóstego bitu statusu linii.

Format:

Kod op.	Operand
152	-



Realizacja: Następuje wyzerowanie bitu, pamiętającego zmianę stanu z ON na OFF linii "110" interfejsu V-24. Bit ten przy wykonywaniu rozkazu RS jest szóstym bitem słowa statusu linii.

Rozkazy, dotyczące konwertera

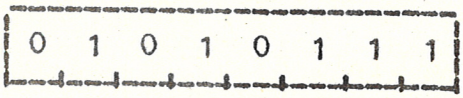
Minisystem, pracujący w reżimie konwertera, pozwala organizować zapis danych na taśmie magnetycznej szerokości 1/2", powszechnie stosowanych w pamięciach zewnętrznych dużych komputerów. Poza tym minisystem może realizować zadania minisystemu w wersji podstawowej.

Zapis informacji na taśmie magnetycznej możliwy jest w systemie PE /Phase Encoding/ lub NRZ /Non-Return-to-Zero/

NRZF/087. Wybranie pamięci taśmowej.

Format:

Kod op.	Operand
087	-

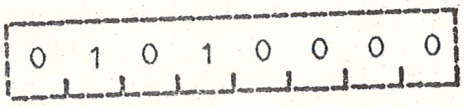


Realizacja: Rozkaz powoduje zmianę stanu podłączenia pamięci taśmowej do systemu. Przy pierwszym użyciu rozkazu następuje wybranie PT, przy kolejnym odłączeniu itd.

WST/080. Ruch taśmy w tył lub w przód.

Format:

Kod op.	Operand
080	-

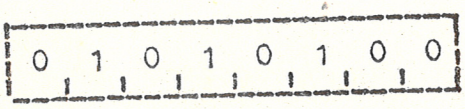


Realizacja: Zerowanie systemu lub brak wybrania pamięci taśmowej jest równoznaczne z ustawieniem pamięci taśmowej na ruch w przód.
Rozkaz WST może być zrealizowany tylko po wybraniu PT i przy pierwszym użyciu ustawia PT na ruch wstecz, przy drugim na ruch w przód itd.

PST/084. Odczyt lub zapis.

Format:

Kod op.	Operand
084	-



Realizacja: Zerowanie systemu lub brak wybrania pamięci taśmowej ustawia ją w stan odczytu.
Po wybraniu PT, pierwszy rozkaz PSN ustawia PT w stan zapisu bloku danych, kolejny rozkaz PSN spowoduje ustawienie PT w stan odczytu itd.

PPSN/082. Pozycjonowanie głowicy zapisującej przed ponownym zapisem.

Format:

Kod op.	Operand
082	-

0	1	0	1	0	0	1	0

Realizacja: Rozkaz modyfikuje czas opóźnienia zatrzymania się taśmy po operacji odczytu przy ruchu wstecz, dla ustalenia właściwej pozycji głowicy magnetycznej przed operacją ponownego zapisu bloku. Zerowanie systemu lub brak wybrania pamięci taśmowej ustawia ją w stan "brak pozycjonowania głowicy". Po wybraniu PT pierwszy rozkaz PPSN wprowadza pozycjonowanie, kolejny - zdejmuje itd.

KAS/086. Kasowanie.

Format:

Kod op.	Operand
086	-

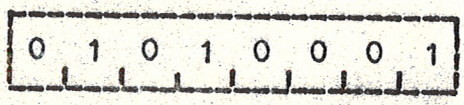
0	1	0	1	0	1	1	0

Realizacja: Zerowanie systemu lub brak wybrania pamięci taśmowej ustawia PT w stan "brak kasowania". Po wybraniu PT pierwszy rozkaz KAS ustala stan kasowania, kolejny zdejmuje itd. Kasowanie taśmy trwa do czasu podania rozkazu OSTB/090.

PZT/081. Zapisywanie znacznika taśmy.

Format:

Kod op.	Operand
081	-

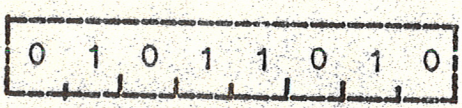


Realizacja: Zerowanie systemu lub brak wybrania pamięci taśmowej ustawia ją w stan "bez zapisu znacznika". Po wybraniu PT pierwszy rozkaz PZT ustala reżim zapisu znacznika, kolejny zdejmuje go itd.

OSTB/090. Ostatni bajt informacji.

Format:

Kod op.	Operand
090	-

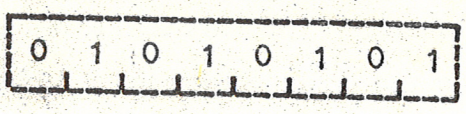


Realizacja: Rozkaz ten należy wysłać po przesłaniu ostatniego bajtu informacji do zapisu na taśmie lub na zakończenie procesu kasowania.

REW/085. Przewijanie.

Format:

Kod op.	Operand
085	-

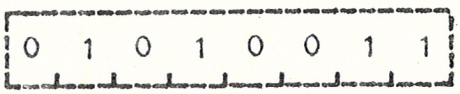


Realizacja: Po wybraniu pamięci taśmowej i podaniu rozkazu REW nastąpi przewinięcie taśmy do fizycznego początku /do znacznika BOT/. W czasie wykonywania tego rozkazu pamięć taśmowa nie przyjmuje żadnych rozkazów.

ST/083. Strob do wykonania rozkazu.

Format:

Kod op.	Operand
083	-



Realizacja: Po wybraniu pamięci taśmowej i ustawieniu reżimu pracy przy pomocy w/w rozkazów, należy podać rozkaz ST, inicjujący pracę MT w zadanym reżimie.

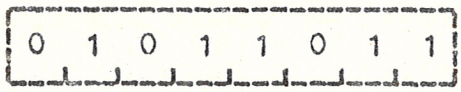
Przykład:

NRZF	087	Dany ciąg rozkazów spowoduje
WST	080	odczyt jednego bloku danych
ST	083	przy ruchu taśmy wstecz.

UNG/091. Ustawienie pamięci w reżim pracy w systemie NRZ.

Format:

Kod op.	Operand
091	-



Realizacja: Pamięć taśmowa po wyzerowaniu ustawia się zawsze w reżim pracy w systemie PE /phase encode/. Wybranie pamięci i rozkaz UNG spowoduje przestawienie pamięci na pracę w systemie NRZ /Non-Return-to Zero/. Jeżeli po ustawieniu reżimu NRZ spowodujemy zapis bloku, a taśma znajduje się na BOT, to zostanie wymazany znacznik taśmy. Wtedy taśma startując od początku, czyli od BOT, ustawia się sama w rodzaj pracy NRZ.

SPT/088. Wystawienie statusu pamięci taśmowej.

Format:

Kod op.	Operand
088	-

0	1	0	1	1	0	0	0

Realizacja: Na szyny informacyjne systemu zostaje wystawione słowo statusu pamięci taśmowej w postaci:

7	6	5	4	3	2	1	0
X	X	X	X	X	X	X	X

- 1 - stan przewijania
- 0 - wykrycie znacznika taśmy,
- 1 - wykrycie błędu zapisu lub odczytu,
- 0 - jeśli taśma stoi na EOT,
- 0 - jeśli taśma stoi na znaczniku BOT,
- 1 - brak pierścienia, zezwalającego, na zapis
  0 - zapis dozwolony /jest pierścień',
- 1 - pamięć jest w stanie gotowości,
- 1 - zajętość PT. PT wykonuje jakiś rozkaz i nowych rozkazów nie przyjmuje.

Uwaga. Możliwe jest maskowanie słowa statusu zawartością rejestru R1. Pozwala to na badanie wybranych bitów słowa statusu. Mechanizm zilustrujemy na przykładzie.

Przykład:

Badany stan gotowości PT, czyli szósty bit statusu PT:

LDV; 064	233; 064	- ładujemy oczekiwany kod do V,
LDR; 064	232; 064	- ładujemy maskę /0000 0010/ do R1,
SPT	088	- wystawiany status PT
MRP	222	- realizacja maskowania statusu z zapisem w uprzednio zaadresowanej komórce pamięci,
REV, GT	226; GT	- skok eśli pamięć była gotowa.

GT: Pamięć gotowa

Maska w R1:

0	1	0	0	0	0	0	0

Słowo statusu PT:

X	X	X	X	X	X	X	X

Zawartość R1 po wykonaniu operacji maskowania:

0	X	0	0	0	0	0	0

Oczekiwany kod ustawiony w rejestrze V:

0	1	0	0	0	0	0	0

Rutyny

Rutyny /makroinstrukcje/ rozszerzają listę instrukcji minikomputera. Operują całymi blokami danych. Blok danych jest określany przez adres pierwszego jego bajtu oraz znak końca bloku /kod 255, umieszczony za ostatnim bajtem informacyjnym/.

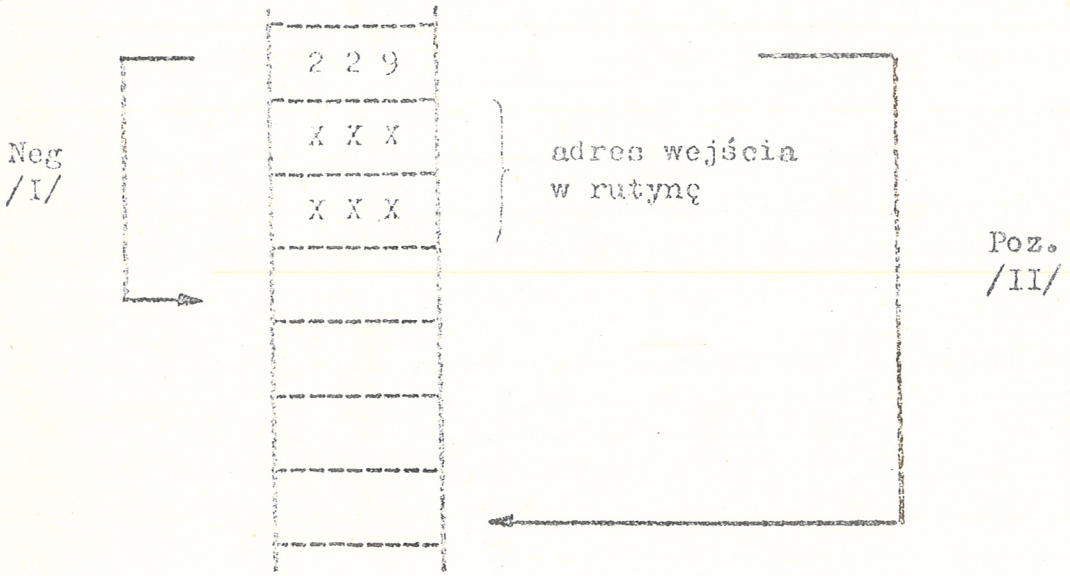
Wywołanie rutyny odbywa się tak samo, jak wywołanie podprogramu, tj. za rozkazem CAL/229 podaje się adres wejścia w rutynę. Rutyny są umieszczone w pamięci typu PROM, czyli ich adresy początkowe są w granicach 000.000 do 007.255.

Każda rutyna i każdy podprogram kończy się sekwencją rozkazów:

INA/231
INA/231
EXI/230

zapewniającą powrót z rutyny na kolejny rozkaz za rozkazem jej wywołania.

Jeżeli rutyna sprawdza jakiś warunek, to w zależności od spełnienia tego warunku posiada dwa wyjścia. Zazwyczaj przy niespełnionym warunku następuje powrót z rutyny na kolejną instrukcję. Przy spełnionym warunku następuje tzw. pozytywne wyjście z rutyny na instrukcję, adres której jest o 3 większy od adresu instrukcji, na którą nastąpiłby powrót przez negatywne wyjście:



Każda rutyna jest identyfikowana przez jej adres wejścia /adres pierwszej instrukcji w rutynie/.

Adresy wejścia w rutyny, stanowiące podstawowy komplet, oraz opisy tych rutyn zamieszczany poniżej. Zawartość pamięci PROM /w tym treść rutyn/ zamieszczamy w załączniku 2.

Rutyny, działanie których dotyczy procesora, drukarki i klawiatury.

004.037. Sygnalizacja błędu w danych.

Rutyna zapala lampkę nr 12 /czerwona lampka przy klawiaturze, oznaczona literami "FD"/ i zatrzymuje realizację programu. Kontynuacja programu jest możliwa po zgaszeniu lampki FD klawiszem Ck .

Rutyna wykorzystuje rejestry R1 i V. Pozostawia w nich ślady - kod 230.

004.041. Sygnalizacja błędu w obsłudze.

Rutyna zapala lampkę nr 11 /czerwona lampka przy klawiaturze, oznaczona literami "FB"/ i zatrzymuje realizację programu. Kontynuacja programu jest możliwa po zgaszeniu lampki FB klawiszem Ck.

004.021. Test maksymalnej długosci bloku.

Rutyna sprawdza, czy w bloku, którego adres jest ustawiony w rejestrze PADR, liczba bajtów nie przekracza x. Posiada dwa wyjścia:

- wyjscie I, negatywne - gdy blok posiada conajmniej x bajtów,
- wyjście II, pozytywne - gdy blok jest krótszy, niż x bajtów.

Warunki początkowe:

- adres bloku jest ustawiony w rejestrze PADR,
- w liczniku Z znajduje się kod 255-x, gdzie x - testowana długośc bloku.

Rutyna wykorzystuje rejestry R1, V, Z, PADR.
Pozostawia następujące ślady:

rejestr R1 - kod 255
rejestr V - kod 255
licznik Z - kod 255
rejestr PADR - przy wyjsciu pozytywnym - adres konca bloku,
- przy wyjsciu negatywnym - znaczenie losowe,

oraz reżim adresacji pamięci poprzez rejestr PADR.

004.018. Test maksymalnej długości bloku z RO.

Rutyna działa jak 004.021 z tym, że automatycznie ustawia w rejestrze PADR adres Rejestru Operacyjnego /009.000/. Testuje blok z RO. Wymaga tylko ustawienia w liczniku Z testowanej długości bloku.
Wykorzystuje te same rejestry i pozostawia takie same ślady, jak rutyna 004.021.

004.002. Test długości minimalnej bloku, adresowanego PADR-em.

Rutyna sprawdza, czy w bloku, którego adres jest ustawiony w rejestrze PADR, liczba bajtów jest nie mniejsza, niż x.
Rutyna posiada dwa wyjścia:
- wyjście I, negatywne - gdy blok jest krótszy, niż x bajtów,
- wyjście II, pozytywne - gdy testowany blok ma conajmniej x bajtów.

Warunki początkowe:
- adres bloku jest ustawiony w rejestrze PADR,
- w liczniku Z znajduje się kod 255-x, gdzie x - testowana ilość bajtów w bloku.

Rutyna wykorzystuje rejestry: R1, V, Z, PADR.
Pozostawia następujące ślady:
rejestr R1 - kod 255
rejestr V - kod 255
licznik Z - kod 255
rejestr PADR - przy pozytywnym wyjściu - adres końca bloku
- przy negatywnym wyjściu - znaczenie losowe.

003.255. Test długości minimalnej bloku z RO.

Rutyna działa jak 004.002 z tym, że automatycznie ustawia w rejestrze PADR adres RO /009.000/ i testuje blok z RO.
Wymaga tylko ustawienia w liczniku Z testowanej liczby bajtów w bloku.
Wykorzystuje te same rejestry i pozostawia takie same ślady, jak rutyna 004.002.

003.174. Zapis cyfr z klawiatury do bloku, zaadresowanego BADR-em.

Rutyna pozwala na zapis cyfr z klawiatury do bloku, tworzonego począwszy od adresu, ustawionego w rejestrze BADR. Kod klawisza "00" jest zastępowany dwoma kolejnymi kodami klawisza "0". Zakończenie zapisu cyfr następuje przez naciśnięcie dowolnego klawisza nienumerycznego. Kod tego klawisza zostaje zapamiętany w komórce 008.255 oraz w rejestrze R1. Za ostatnią wprowadzoną cyfrą jest wpisywany znak końca bloku /kod 255/.

Warunek początkowy: w rejestrze BADR jest ustawiony adres początku tworzonego bloku.

Rutyna wykorzystuje rejestry: R1, V, PADR, BADR. Pozostawia następujące ślady:

rejestr V - kod 231

rejestr R1 } - kod naciśniętego klawisza nienumerycznego
komórka 008.253 }

rejestr PADR - adres 008.253

rejestr BADR - adres końca bloku cyfr,

oraz reżim adresacji pamięci poprzez rejestr PADR.

003.171. Zapis cyfr z klawiatury do bloku w R0.

Rutyna działa jak rutyna 003.174 z tym, że blok z wprowadzanych z klawiatury cyfr tworzy w R0 /adres R0, tj.009.000 jest automatycznie ustawiany w rejestrze BADR/.
Rutyna wykorzystuje te same rejestry i pozostawia takie same ślady, jak rutyna 003,174.

000.044. Dodawanie lub odejmowanie bloków, zaadresowanych PADR-em i BADR-em. Wynik pod adresem z PADR.

Rutyna liczy sumę lub różnicę liczb w blokach, których adresy są ustawione w rejestrach PADR i BADR. Wynik jest zapisywany w bloku, zaadresowanym przez PADR. W komórkach, poprzedzających pierwsze bajty sumowanych bloków, wpisywany jest kod 255. Niedopuszczalne jest odejmowanie liczby większej od mniejszej. Rutyna operuje kodami cyfr, więc pozwala na dodawanie lub odejmowanie liczb naturalnych. Typ wykonywanej operacji - dodawanie lub odejmowanie jest określony przez zawartość licznika Z.

Warunki początkowe:

- adres bloku - akumulatora został załadowany do rejestru PADR,
- adres bloku z odejmnikiem /lub drugim składnikiem/ jest ustawiony w rejestrze BADR,
- licznik Z zawiera kod 207 - jeśli liczymy różnicę bloków lub inny kod /zaleca się używanie kodu 206/ - jeśli sumujemy bloki.

Rutyna wykorzystuje rejestry R1, V, Z, A, B, PADR i BADR.
Pozostawia następujące ślady:

rejestr V
- kod 255 - jeśli było przeniesienie "1",
- kod 206 - jeśli przeniesienia nie było,

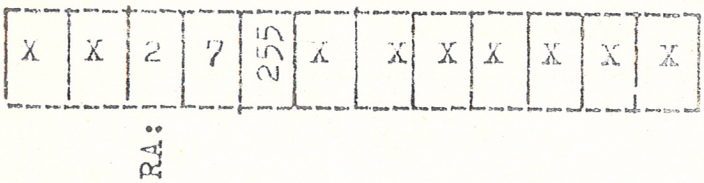
rejestr R1
- kod 206 - jeśli było przeniesienie "1" przy dodawaniu,
- kod 207 - jeśli przeniesienia nie było,

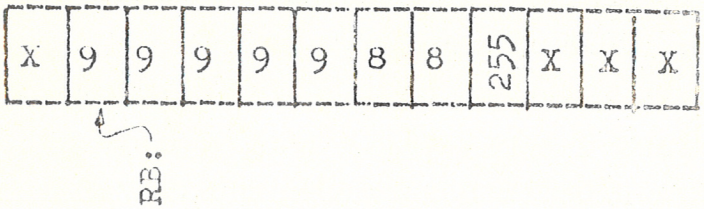
rejestry A i B
- kody 0000 /binarnie/, co odpowiada kodowi 207 w rejestrze R1,

rejestry PADR i BADR
- jeśli nie było przeniesienia - adresy początkowe bloków, zmniejszone o 1,
- jeśli było przeniesienie - adresy początkowe bloków,

oraz reżim adresacji pamięci poprzez rejestr PADR.

Przykłady:
Dodawane są liczby, zawarte w rejestrach RA i RB.





a/ Po wykonaniu ciągu instrukcji:

```
- - -          | - - -
LDZ; 206       | 234, 206
PAD; RB        | 212; RB
BUD; RA        | 209; RA
CAL; 000.044   | 229; 000.044
- - -          | - - -
```

/RB: = RB + RA/



RA:

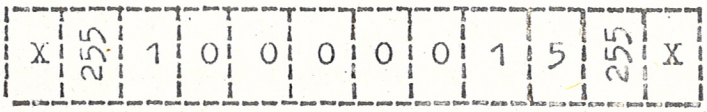


RB:

b/ Po wykonaniu ciągu instrukcji:

```
- - -          | - - -
LDZ; 206       | 234; 206
PAD; RA        | 212; RA
BUD; RB        | 209; RB
CAL; 000.044   | 229; 000.044
- - -          | - - -
```

/RA: = RA + RB/



RA:

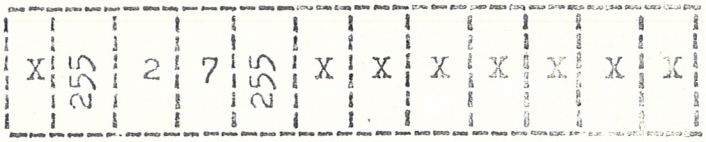


RB:

c/ Po wykonaniu ciągu instrukcji:

- - -	- - -
LDZ; 207	234; 207
PAD; RB	212; RB
BUD; RA	209; RA
CAL; 000.044	229; 000.044
- - -	- - -

/RB: = RB - RA/



RA:



RB:

d/ Ciąg instrukcji:
dla danej zawartości rejestrów RA i RB jest błędny.

002.230. Dodawanie lub odejmowanie bloku z RO i bloku zaadresowanego BADR-em. Wynik w RO.

Rutyna działa jak 000.044. z tym, że w rejestrze PADR jest automatycznie ustawiany adres Rejestru Operacyjnego /009.000/. Wynik operacji zapisywany jest w RO.
Warunki początkowe - jak w rutynie 000.044. Nie obowiązuje warunek ustawienia adresu w rejestrze PADR.
Rutyna wykorzystuje te same rejestry i pozostawia takie same ślady, jak rutyna 000.044.

002.224. Dodawanie lub odejmowanie bloku, zaadresowanego PADR-em i bloku z RO. Wynik pod adresem z PADR.

Rutyna działa jak 000.044. Operandy pobiera z RO i z bloku, zaadresowanego PADR-em. Wynik zapisuje na miejscu operanda, zaadresowanego PADR-em.

Warunki początkowe - jak w rutynie 000.044. Nie obowiązuje warunek ustawienia adresu w rejestrze BADR. Rutyna wykorzystuje te same rejestry i pozostawia takie same ślady, jak rutyna 000.044.

004.057. Badanie relacji liczb w blokach, zaadresowanych PADR-em i BADR-em.

Rutyna bada relację dwóch liczb dziesiętnych z bloków, których adresy są ustawione w rejestrach PADR i BADR. Posiada dwa wyjścia:

- wyjście I, negatywne - gdy liczba w bloku, zaadresowanym PADR-em, jest większa lub równa liczbie w bloku, adresowanym przez BADR.
- wyjście II, pozytywne - gdy liczba w bloku, zaadresowanym przez BADR jest większa od liczby w bloku, adresowanym przez PADR.

Rutyna wykorzystuje rejestry R1, V, Z, PADR, BADR, A i B.
Po zakończeniu pracy rutyny stany rejestrów nie są jednoznacznie określone, natomiast pamięć pozostaje w reżimie adresacji poprzez rejestr BADR.

002.237. Badanie relacji liczby w bloku, adresowanym PADR-em i liczby z RO.

Rutyna działa jak 004.057. Operandy pobiera z RO i z bloku, zaadresowanego PADR-em. Posiada dwa wyjścia:

- wyjście I - gdy liczba w bloku, zaadresowanym PADR-em, jest większa lub równa liczbie z RO,
- wyjście II - gdy liczba w RO jest większa od liczby w bloku, adresowanym przez PADR.

Jedyny warunek: rejestr PADR zawiera adres bloku z liczbą, porównywaną z liczbą z RO.
Rutyna wykorzystuje te same rejestry i pozostawia takie ślady, jak rutyna 004.057.

004.042. Badanie relacji liczby z RO i liczby z bloku, zaadresowanego BADR-em.

Rutyna działa jak 004.057. Operandy pobiera z RO i z bloku, zaadresowanego BADR-em.
Posiada 2 wyjścia:

- wyjscie I - gdy liczba z RO jest większa lub równa liczbie z bloku, adresowanego przez rejestr BADR,
- wyjscie II - gdy liczba w bloku jest większa od liczby z RO.

Warunek początkowy: Rejestr BADR zawiera adres bloku z liczbą, porównywaną z liczbą z RO.
Rutyna wykorzystuje te same rejestry i pozostawia takie same, jak rutyna 004.057.

003.072 Przeniesienie bloku, zaadresowanego PADR-em pod adres, ustawiony w rejestrze BADR

Rutyna pozwala dublować informację, przedstawioną w postaci bloków i rozmieszczoną w dowolnych obszarach pamięci operacyjnej. Razem z zawartością bloku dublowany jest również znak końca bloku /kod 255/.

Warunki początkowe:

- adres dublowanego bloku jest ustawiony w rejestrze PADR,
- adres, dokąd będzie przenoszony blok danych, jest ustawiony w rejestrze BADR.

Rutyna wykorzystuje rejestry R1, V, PADR i BADR. Pozostawia następujące ślady:

rejestry R1 i V - kod 255

rejestry PADR i BADR } adresy końców bloków: dublowanego i tworzonego,

oraz reżim adresacji pamięci poprzez rejestr BADR.

Blok dublowany nie ulega zniekształceniu. Przykład: Należy wyzerować pięciobajtowy licznik L1. Wykorzystamy w tym celu rutynę oraz rejestr pięciobajtowy, zawierający same zera:

X	X	0	0	0	0	0	255	X	X

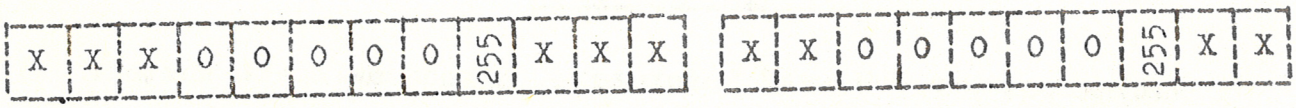
ZE:

X	X	X	X	X	X	X	255

L1

Po wykonaniu ciągu instrukcji:

PAD;	ZE	212;	ZE
BUD;	L1	209;	L1
CAL;	003.072	229;	003.072



ZE: L1:

PADR: BADR:

002.211. Przeniesienie bloku, zaadresowanego PADR-em do RO

Rutyna działa podobnie, jak 003.072 z tym, że automatycznie ustawia adres rejestru operacyjnego /009.000/ w rejestrze BADR. Obowiązuje więc tylko jeden warunek początkowy: - adres przenoszonego bloku jest ustawiony w rejestrze PADR.
Rutyna wykorzystuje te same rejestry i pozostawia takie same ślady, jak rutyna 003.072.

002.217. Przeniesienie bloku z RO pod adres, ustawiony w rejestrze BADR

Rutyna działa jak 003.072 z tym, że automatycznie ustawia adres rejestru operacyjnego /009.000/ w rejestrze PADR. Warunek początkowy: - adres, dokąd będzie przenoszony blok z RO, jest ustawiony w rejestrze BADR.
Rutyna wykorzystuje te same rejestry i pozostawia takie same ślady, jak rutyna 003.072.

003.061. Opóźnienie X milisekund

Rutyna wprowadza opóźnienie w granicach od 1 do 256 ms. Wielkość opóźnienia deklarowana jest w liczniku Z.
Warunek początkowy: w liczniku Z znajduje się kod liczby /255-X/, gdzie x - deklarowane opóźnienie w milisekundach. Kod 255 odpowiada maksymalnemu opóźnieniu 256 ms.
Rutyna używa rejestrów R1, V, Z. Pozostawia następujące ślady:

rejestr R1 - kod 255

rejestr V - kod 255

licznik Z - kod 255

Przykład: Instrukcje:

LDZ; 135	234; 135
CAL; 003.061	229; 003.061
...	...

wprowadzają opóźnienie 120 ms.

000.232. Formatowanie liczby w bloku, adresowanym PADR-em, do X pozycji

Rutyna uzupełnia ilość cyfr liczby do X, dopisując z lewej strony zera. W komórce, poprzedzającej pierwszy byjt bloku, zostaje zapisany kod 255.

Warunki początkowe:

- adres bloku jest ustawiony w rejestrze PADR,
- w liczniku Z znajduje się kod liczby /255-X/, gdzie X - żądana ilość cyfr w formacie liczby.

Rutyna wykorzystuje rejestry R1, V, Z i PADR.

Pozostawia następujące ślady:

rejestry R1, V, Z - kod 255

rejestr PADR - adres końca bloku, tj. adres większy o x+1 od początkowego,

oraz reżim adresacji pamięci poprzez rejestr PADR.

Przykład: Uzupełnimy liczbę w rejestrze RS do 7 cyfr.

Sytuacja wyjściowa:

X	X	1	2	5	0	X	X	X	X	X

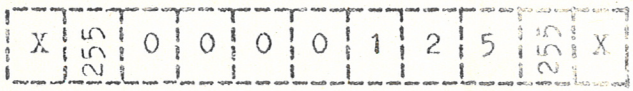
/RS=125/

RS: ↑

Po wykonaniu ciągu instrukcji:

PAD;RS	212; RS
IDZ;248	234; 248 /x=7/
CAL;000.232	229; 000.232

otrzymamy następującą formę danych w rejestrze RS.



RS:0000125

RS: PADR:

002.205. Formatowanie liczby w RO

Rutyna działa jak 000.232 z tym, że operuje danymi z RO. Adres RO /009.000/ jest automatycznie ustawiany w rejestrze PADR. Obowiązuje tylko warunek zadeklarowania w liczniku Z ilości cyfr liczby.

Rutyna działa na tych samych rejestrach i pozostawia takie same ślady, jak 000.232.

000.196. Przeniesienie ostatnich x bajtów z bloku, zaadresowanego PADR-em do bloku, adresowanego BADR-em

Ostatnie x bajtów bloku, którego adres jest ustawiony w rejestrze PADR, jest przenoszone do bloku, tworzonego począwszy od adresu, ustawionego w rejestrze BADR. Na końcu tworzonego bloku zostaje wpisany kod 255 /znak końca bloku/. W miejsce pobranych bajtów informacji wpisywany jest kod 255 /blok zostaje skrócony/.

Warunki początkowe:

- adres bloku, z którego pobieramy informację, ustawiony jest w rejestrze PADR,
- adres tworzonego bloku ustawiony jest w rejestrze BADR,

- w liczniku Z znajduje się kod liczby /255-x/, gdzie x - liczba przenoszonych bajtów.

Rutyna wykorzystuje rejestry R1, Z, V, PADR i BADR.

Pozostawia następujące ślady:

rejestry R1, V, Z - kod 255

rejestr PADR - adres końca bloku, z którego pobierana była informacja,

rejestr BADR - adres końca stworzonego bloku /adres o x+1 większy od począ kowego/,

oraz reżim adresacji pamięci poprzez rejestr BADR.

Przykład: W 9-bajtowym rejestrze od adresu 019.000 zapisany jest tekst: "SEKTOR 13". Rejestr RS będzie tworzony z ostatnich trzech bajtów rejestru 019.000.

X	X	S	E	K	T	O	R	spacja	1	3	255	X	X

019.000:

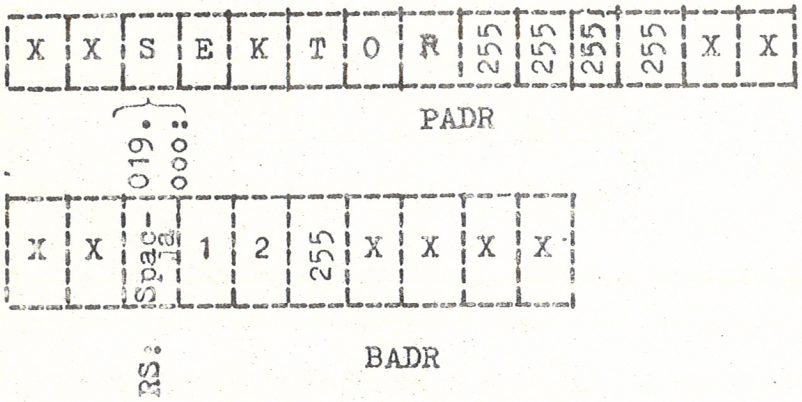
X	X	X	X	X	X	X	X	X

RS:

Po wykonaniu ciągu instrukcji:

LDZ; 252	234; 252	/x=3/
PAD; 019.000	212; 019.000	
BUD; RS	209; RS	
CAL; 000.196	229; 000.196	

os adresu 019.000 rozpoczyna się rejestr 6-bajtowy, zawierający tekst: "SEKTOR", a od adresu RS rozpoczyna się jrejestr 3-bajtowy, zawierający tekst: " 13".



000.193. Przeniesienie x ostatnich bajtów bloku z RO do bloku zaadresowanego BADR-em

Rutyna działa podobnie, jak 000.196 z tym, że w rejestrze PADR zostaje automatycznie ustawiony adres RO /009.000/. Po przeniesieniu x bajtów z RO do bloku, którego adres jest ustawiony w rejestrze BADR, blok w RO zostaje skrócony o x znaków.

Warunki początkowe:

- adres tworzonego bloku jest załadowany do rejesytru BADR,
- w liczniku Z znajduje się kod liczby /255-x/, gdzie x-ilość przenoszonych bajtów,

Rutyna wykorzystuje te same rejestry i pozostawia takie same ślady, jak 000.196.

003.086. Sprawdzenie identyczności bloków, zaadresowanych PADR-em i BADR-em

Rutyna porównuje bajt po bajcie dwa bloku, których adresy zostały ustawione w rejestrach PADR i BADR. Posiada dwa wyjścia:

- wyjście I, negatywne - jeśli bloki różnią się długością lub zawartością,
- wyjście II, pozytywne - gdy bloki są identyczne.

Warunki początkowe: adresy porównywanych bloków są załadowane do rejestrów PADR i BADR.

Rutyna wykorzystuje rejestry R1, V, LRC, PADR i BADR. Pozostawia następujące ślady:

rejestr V - kod 000

rejestry R1, LRC - kod 000 } tylko przy wyjściu pozytywnym

rejestry PADR i BADR adresy końców bloków } tylko przy wyjściu pozytywnym

Przy wyjściu negatywnym tylko w rejestrze V jest kod 000, w pozostałych rejestrach zawartość nie jest jednoznacznie określona. Pamięć pozostaje adresowana przez rejestr BADR.

Przykład: Porównywany jest stan licznika L1 z wartością graniczną 1000, zapisaną w rejestrze WZ /np. przy liczeniu obrotów pętli/. Zakładamy, że obarejestry L1 i WZ zostały zadeklarowane jako czterobajtowe:

```
PT: /Zwiększenie
    liczby w L1 o 1,
    wykonanie peł-
    nego cyklu/
SP: PAD; WZ
    BUD;L1
    CAL;003.086
    JMP;PT
KP: /wyjście z
    pętli/
```

```
PT: ---
    ---
    ---  } cykl operacji
    ---
    ---
SP: 212; WZ
    209; L1
    229; 003.086
    224; PT
KP: ---
    ---
    ---
```

002.249. Sprawdzenie identyczności bloku w RO i bloku, zaadresowanego PADR-em

Rutyna działa podobnie, jak 003.086 z tym, że w rejestrze BADR zostaje automatycznie ustawiony adres rejestru operacyjnego /009.000/.

Obowiązuje jeden warunek początkowy: adres bloku, porównywanego z blokiem z RO, musi być ustawiony w rejestrze PADR.

Rutyna używa tych samych rejestrów i pozostawia takie same ślady, jak 003.086.

003.159. Drukowanie bloku, zaadresowanego BADR-em

Rutyna powoduje wydrukowanie bloku, adres którego jest ustawiony w rejestrze BADR. Pierwszy znak bloku jest drukowany w kolejnej wolnej pozycji wiersza.

Warunek początkowy: adres bloku jest ustawiony w rejestrze BADR.

Rutyna wykorzystuje rejestry R1, V, BADR. Pozostawia następujące ślady:

rejestry R1 i V - kod 255

rejestr BADR - adres końca bloku,

oraz reżim adresacji pamięci poprzez rejestr BADR.

Przykład: Przesunięcie papieru o 1 wiersz i drukowanie tekstu, zawartego w rejestrze T1:

---	---	
LDR;LF	232; 245	przesunięcie papieru o 1 wiersz
PRT	223	
BUD;T1	209; T1	
CAL; 003.159	229; 003.159	
---	---	
T1:001;PRZYKŁAD	T1:001; PRZYKŁAD	

003.156. Drukowanie bloku z RO

Rutyna działa jak 003.159 z tym, że automatycznie ustawia w rejestrze BADR adres RO /009.000/, w wyniku czego informacja do wydruku pobierana jest z RO.

Warunków początkowych rutyna nie wymaga. Wykorzystuje te same rejestry i pozostawia takie same ślady, jak rutyna 003.159.

003.124. Drukowanie z tabulacją bloku z R0

Rutyna pozwala drukować blok z R0 z tabulacją zwykłą lub modyfikowaną:

Tabulacja zwykła:	Tabulacja modyfikowana:
1	1
12	12
123	123
12345	1234
1234567	12345678
	123456789

Tabulacji zwykłej odpowiada kod 207, ustawiony w rejestrze V, a tabulacji modyfikowanej - dowolny kod, różny od 207 /zaleca się używanie kodu 206/.

Warunki początkowe:

- zawartość rejestru V określa rodzaj tabulacji,
- w liczniku Z ustawiony jest kod /255-x/, gdzie x - adres tabulacji.

Rutyna wykorzystuje rejestry R1, V, Z i BADR. Pozostawia następujące ślady:

rejestry R1 i V - kod 255,
licznik Z - stan losowy,
rejestr BADR - adres końca bloku,

oraz reżim adresacji pamięci poprzez rejestr BADR.

Przykład: Wypełniany jest formularz. Rubryka w której trzeba wydrukować liczbę, znajduje się między piętnastą a dwudziestą piątą pozycją w wierszu:

```
10        15        20        25        30        35
          :         :         :         :         :
          :         :       1 :         :         :
          :         :     125 :         :         :
          :         :  1256712 :         :         :
          :         :123456017 :         :         :
          :         :    1422 :         :         :
          :         :  123456 :         :         :
```

Realizują to instrukcje:

LDV; 1	233;206
LDZ; 231	234;231 /255-24/
CAL; 003.124	229;003.124

004.160. Drukowanie tekstu z klawiatury z zapamiętywaniem od adresu z BADR

Rutyna pozwala drukować wprowadzane z klawiatury znaki z równoczesnym tworzeniem z nich bloku, począwszy od adresu, ustawionego w rejestrze BADR. Wydruk pierwszego wprowadzonego znaku następuje w pierwszej wolnej pozycji w wierszu. Zakończenie wprowadzania znaków następuje po naciśnięciu klawisza funkcyjnego A, C_k lub BS. Pozostałe klawisze funkcyjne są traktowane jak klawisze alfanumeryczne. Kod klawisza "00" jest zastępowany dwoma kodami klawisza "0", zaś kod klawisza, którym zakończono wprowadzanie znaków, zostaje zapamiętany w rejestrze R1 oraz w komórce 008.253.

Warunki początkowe:

- adres tworzonego bloku jest ustawiony w rejestrze BADR.

Rutyna wykorzystuje rejestry R1, V, Z, PADR i BADR.

Pozostawia następujące ślady:

licznik Z - bez zmiany

rejestry R1 i V - kod klawisza funkcyjnego, którym zakończono wprowadzanie danych,

rejestr PADR - adres 008.253,

rejestr BADR - adres końca utworzonego bloku,

oraz reżim adresacji pamięci poprzez rejestr PADR.

004.153. Drukowanie z tabulacją tekstu z klawiatury z zapamiętywaniem od adresu z BADR

Rutyna działa podobnie, jak rutyna 004.160. Pozwala jednak drukować wprowadzane znaki z tabulacją /zwykłą/.

Warunki początkowe:

- adres tworzonego bloku jest ustawiony w rejestrze BADR,
- licznik Z zawiera kod /255-x/, gdzie x - żądany adres tabulacji.

Rutyna wykorzystuje te same rejestry i pozostawia takie same ślady, jak 004.160.

Przykład: Wypełniany jest formularz. Rubryka, w której należy wydrukować nazwisko i imię pracownika, znajduje się między ósmą a dwudziestą dziewiątą pozycją w wierszu:

```
5    8.   10    15    20    25    .30
     |                            |
     |   KOWALSKI  LUCJAN         |
     |   MALINOWSKI ZBIGNIEW      |
```

Realizują to instrukcje:

```
---                  |  ---
LDZ; 246             |  234;246          /255-9/
BUD; 018.000         |  209;018.000
CAL; 004.153         |  229;004.153
---                  |  ---
```

Blok jest tu tworzony od adresu 018.000.

004.150. Drukowanie z tabulacją tekstu z klawiatury z zapamiętywaniem w RO

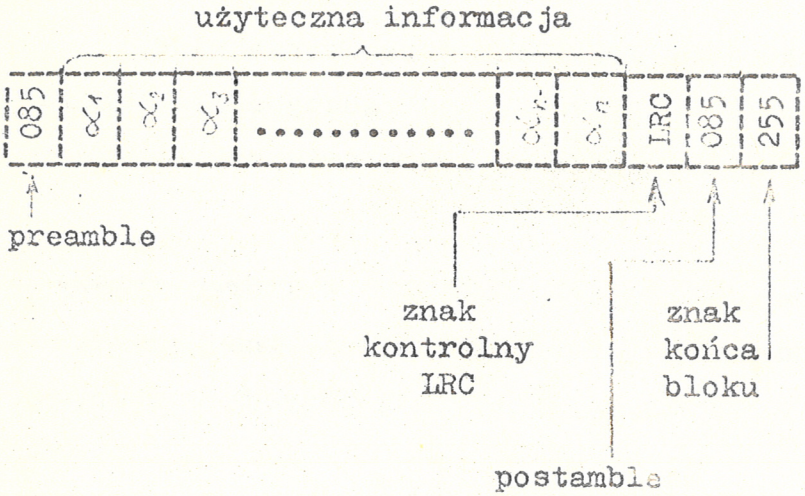
Rutyna działa jak 004.153 z tym, że wprowadzane dane są zapamiętywane w RO. Adres RO jest automatycznie ustawiany w rejestrze BADR.

Obowiązuje tylko warunek ustawienia w rejestrze Z adresu tabulacji.

Rutyna używa tych samych rejestrów i pozostawia takie same ślady, jak rutyny 004.153 i 004.160.

Rutyny, wykorzystywane przy pracy z pamięcią kasetową

Blok danych, zapisywany w pamięci kasetowej, ma postać:



Ponieważ blok do zapisu w PK jest przygotowywany na 8 stronie, to istnieje ograniczenie jego długości do 256 bajtów, tj. 252 bajtów informacyjnych + 4 bajtów służbowych.

003.022. Przygotowanie bloku, zaadresowanego PADR-em, do zapisu na PK

Blok danych, którego adres został ustawiony w rejestrze PADR, zostaje przeniesiony na ósmą stronę pamięci. Przed pierwszym bajtem bloku zostaje wpisany "preamble" /kod 085/. Dla całego bloku zostaje obliczony znak kontrolny LRC i wpisany po ostatnim bajcie danych. Za znakiem LRC zostaje wpisany "postamble" /kod 085/ i znak końca bloku /kod 255/.

Warunek początkowy: adres bloku danych jest ustawiony w rejestrze PADR.

Rutyna wykorzystuje rejestry R1, V, PADR, BADR i licznik LRC.

Pozostawia następujące ślady:

rejestry R1 i V - kod 255,

rejestry PADR i BADR } odpowiednio końce bloków: z danymi i przygotowanego do zapisu,

licznik ORC - stan losowy,

reżim adresacji pamięci poprzez rejestr BADR.

Rutyna nie zniekształca bloku źródłowego, natomiast zaciera poprzednią zawartość ósmej strony.

003.019. Przygotowanie bloku z RO do zapisu na PK

Rutyna działa jak 003.022 z tym, że dane pobiera z rejestru operacyjnego. Adres RO jest automatycznie ustawiany w rejestrze PADR. Rutyna wykorzystuje te same rejestry i pozostawia takie same ślady, jak 003.022.

003.003. Dopisywanie dalszego ciągu bloku, przygotowanego do zapisu na PK

Rutyna usuwa z przygotowanego na ósmej stronie bloku znaki: LRC, "postamble" i końca bloku. Umieszcza dane z bloku, zaadresowanego PADR-em, bezpośrednio za ostatnim bajtem informacji uprzednio utworzonego na ósmej stronie bloku. Dla całego w ten sposób utworzonego bloku liczy nowy znak kontrolny LRC i zapisuje go po ostatnim bajcie danych. Za znakiem LRC wpisuje "postamble" i znak końca bloku.
Warunek początkowy: adres bloku, którym jest uzupełniany blok do zapisu na PK, jest ustawiony w rejestrze PADR.
Rutyna wykorzystuje te same rejestry i pozostawia takie same ślady, jak 003.022.

003.000. Dopisywanie z RO dalszego ciągu bloku, przygotowanego do zapisu na PK

Rutyna działa jak 003.003 z tym, że dane, którymi jest uzupełniany uprzednio przygotowany do zapisu blok, pobierane są z RO. Adres RO jest automatycznie ustawiany w rejestrze PADR. Dzięki temu rutyna nie wymaga żadnych warunków początkowych. Wykorzystuje te same rejestry i pozostawia takie same ślady, jak 003.022.

000.127. Pozycjonowanie taśmy do zapisu

Rutyna powoduje zarezerwowanie i podłączenie logiczne pamięci kasetowej, przewinięcie taśmy do fizycznego początku, odszukanie znacznika BOT oraz wykasowanie odcinka początkowego taśmy do ok. 40 mm za otworem BOT. W czasie działania rutyny świeci się lampka indykacyjna przy klawiaturze, oznaczona literą "L". Po zakończeniu pracy rutyny pamięć kasetowa pozostaje w stanie rezerwacji i podłączenia.
Rutyna nie wymaga żadnych warunków początkowych. Wykorzystuje rejestry R1, V i Z. Pozostawia następujące ślady:

rejestry R1, V, Z - kod 255

oraz licznik LRC w stanie wyzerowanym.

002.000. Zapis bloku na PK1

Rutyna powoduje zapis bloku, przygotowanego na ósmej stronie, na taśmę kasetową. Za koniec bloku, przeznaczonego do zapisu uważa się kolejność bajtów z kodami 085 i 255. Zawartość licznika LRC niesie informację o prawidłowości zapisu: kod 085 świadczy o tym, że zapis został wykonany prawidłowo, sa kod 054 w liczniku LRC i kod 255 w komórce 010.000 świadczy o tym, że taśma została przesunięta o ok. 400 mm, ale zapis nie nastąpił.
Obecność w komórce 010.001 kodu 001 świadczy o tym, że w trakcie zapisywania bloku został przekroczony znacznik końca taśmy/EOT/.
Rutyna wykorzystuje rejestry R1, V, Z, LRC, A, B, PADR i BADR. Stan rejestrów po wyjściu z rutyny nie jest jednoznacznie określony.
Jedyny warunek prawidłowego działania rutyny: przygotowanie bloku do zapisu na PK przy pomocy jednej z rutyn: 003.000, 003.003, 003.019 lub 003.022. W czasie działania rutyny świeci się lampka indykacyjna przy klawiaturze, oznaczona literą "L". Po zakończe-

niu pracy rutyny pamięć kasetowa pozostaje w stanie zarezerwowania i podłączenia.

001.000. Pozycjonowanie taśmy do odczytu

Rutyna powoduje zarezerwowanie i podłączenie PK, przewinięcie taśmy do fizycznego początku, odszukanie znacznika BOT i zatrzymanie taśmy. W czasie działania rutyny świeci się lampka indykacyjna przy klawiaturze, oznaczona literą "L". Rutyna wykorzystuje rejestry R1, V, Z, A i B. Pozostawia następujące ślady:

licznik Z - kod 255,
rejestry R1 i V - kod 255,
licznik LRC - w stanie wyzerowania.

Po zakończeniu pracy rutyny PK pozostaje w stanie rezerwacji i podłączenia.

001.128. Odczyt bloku z PK

Rutyna powoduje odczytanie z PK jednego bloku danych i umieszczenie go w rejestrze operacyjnym. Odczytany blok przed zapisem w rejestrze operacyjnym jest pozbawiany znaków "preamble", "postamble" i kodu kontrolnego LRC. Za ostatnim bajtem informacyjnym ma tylko znak końca bloku-kod 255. Zawartość licznika LRC niesie informację o prawidłowości odczytu: kod 085 świadczy o tym, że odczyt został wykonany prawidłowo, zaś kod 054 w liczniku LRC i kod 255 w komórce 010.000 świadczą o tym, że taśma była niezapisana na odcinku conajmniej 400 mm.
Rutyna wykorzystuje rejestry R1, V, Z, A, B, LRC, PADR i BADR.
Po zakończeniu pracy rutyny zawartość wszystkich używanych rejestrów nie jest jednoznacznie określona /wyjątkiem jest omówiony wyżej licznik LRC/, a pamięć pozostaje w reżimie adresacji poprzez rejestr BADR.

002.421. Cofnięcie taśmy o 1 blok

Rutyna cofa taśmę kasetową o 1 blok lub do otworu BOT. Cofnięcie do znacznika BOT nastąpi wtedy, gdy taśma jest ustawiona przed pierwszym blokiem lub jest całkowicie niezapisana. Kod 003 w liczniku LRC i kod 255 w komórce 010.000 świadczy o tym, że taśma została cofnięta do znacznika BOT.

Rutyna wykorzystuje rejestry R1, Z, V, LRC i PADR.

Pozostawia następujące ślady:

rejestry R1 i Z - kod 255

rejestr V
- przy cofnięciu o 1 blok - kod 255
- przy cofnięciu do BOT - kod 252

licznik LRC
- przy cofnięciu o 1 blok - kod 000
- przy cofnięciu do BOT - kod 003.

Rutyny, wykorzystywane przy pracy z pamięcią na dyskach elastycznych

006.010. Sprawdzenie stanu gotowości dysku

Rutyna sprawdza stan, w jakim znajduje się uprzednio wybrany dysk. Posiada dwa wyjścia:

- Wyjście I - gdy wybrany dysk osiągnął stan gotowości i jego głowica została sprowadzona na ścieżkę nr 00
- wyjście II - gdy dysk nie osiągnął stanu gotowości.

Stan gotowości jest osiągany przy spełnieniu następujących warunków:

a/ dysk został wybrany rozkazem "PRUN;X", gdzie X-numer jednostki pamięci FD;

b/ upłynął czas rzędu 500÷600 ms od chwili wybrania dysku,

c/ pamięć jest załadowana dyskiem elastycznym.

Rutyna wykorzystuje rejestry R1, V, Z. Poz-ostawia następujące ślady:

licznik Z - kod 255

rejestry V i R1 - kod 255

rejestr ścieżki - kod 000

rejestr sektora - kod 001

Przykład:

```
     ---                  ----
PP: PRUN;000         PP: 096;000      - Wybrano lewy dysk pamięci
    DLW                  098            FD nr 00
    ---                  ---          } Opóźnienie
    ---                  ---          } ok. 500÷600 ms
    ---                  ---
    CAL;006.010          229;006.010
    JMP,A1               224;A1       - Skok jest wykonywany jeżeli
                                        stwierdzono niegotowość dysku
    BUD; TC              209;TC       - kolejna instrukcja,wykonywana,
    ---                  ---            gdy dysk był niegotowy
A1: ---              A1: ---
```

006.001. Sprawdzenie stanu gotowości dysku po upływie opóźnienia 750 ms

Rutyna wprowadza sztuczne opóźnienie 750 ms. Po upływie tego czasu dalej pracuje jak rutyna 006.010.

Warunek początkowy: licznik Z powinien zawierać kod 255.

Rutyna używa tych samych rejestrów i pozostawia takie same ślady, jak 006.010.

006.000. Sprawdzenie stanu gotowości lewego dysku po upływie opóźnienia 750 ms

Rutyna różni się od poprzedniej tylko automatycznym wybraniem lewego dysku w uprzednio wybranej jednostce pamięci FD. Poza tym jej działanie, warunki początkowe i pozostawiane ślady są takie, jak dla rutyny 006.001.

006.014. Pozycjonowanie głowicy dysku

Rutyna realizuje tzw. funkcję odzyskania pamięci, tzn. ustawia głowicę w wybranej jednostce FD naprzeciw ścieżki nr 00 wybranego dysku.

Rutyna posiada tylko jedno wyjście. Nie należy jej wywoływać, jeżeli wybrany dysk nie osiągnął jeszcze stanu gotowości.

Wykorzystywane są rejestry R1, V i Z. Pozostają w nich następujące ślady:

rejestry R1, V i Z - kod 255
rejestr ścieżki - kod 000
rejestr sektora - kod 001

Przykład:

```
   ---                  ---
   LDV; 255             233; 255
ZN:STFD4             ZN:108              Badanie stanu
   REV;ZN               226;ZN             gotowości dysku
   CAL;006.014          229;006.014
   ---                  ---
```

006.051. Czytanie zawartości sektora

Rutyna odczytuje 128 bajtów informacji z pola informacyjnego sektora, którego adres określają zawartości rejestru danych /nr ścieżki/ i rejestru sektora /nr sektora/. Odczytana informacja zostaje wpisana do pamięci operacyjnej począwszy od adresu zadanego wcześniej w rejestrze adresowym BADR. Na końcu informacji zostaje wpisany znak końca bloku - kod 255. Rutyna posiada dwa wyjścia:

- wyjście I, negatywne, gdy:
  a/ kod kontroli CRC, zapisany na dysku, nie zgadza się z kodem kontroli CRC, liczonym przy odczycie;
  b/ rutyna została wywołana bez zabezpieczenia warunków początkowych.

- wyjście II, pozytywne - gdy odczyt odbył się bez zakłóceń i kody kontroli CRC nie wykazały błędu.

Warunki początkowe:

a/ dysk jest wybrany i w stanie gotowości,

b/ numer sektora został wpisany do rejestru sektora /rozkazem WNSK/118/,

c/ numer ścieżki, na której znajduje się dany sektor, został wpisany do rejestru danych /rozkazem w DAN/119/,

d/ adres początku obszaru pamięci operacyjnej, wydzielonego pod odczytywaną informację, został wpisany do rejestru BADR.

Rutyna wykorzystuje rejestry R1, V, Z i BADR. Pozostawia następujące ślady:

rejestr V - kod 000

rejestr R1 - kod 000
rejestr Z - kod 255
rejestr danych - ostatni bajt informacji
} jeżeli odczyt odbył się prawidłowo

- lub nr ścieżki - przy nieudanym odczycie

rejestr BADR - adres, zwiększony o 129,

oraz reżim adresacji pamięci przez rejestr BADR.

Przykład:

```
---                    |  ---
LDR;075                |  232;075      } 75 ścieżka
WDAN                   |  119
LDR;012                |  232;012      } 12 sektor
WNSK                   |  118
BUD;015.126            |  209;015.126 - adres P0
CAL;006.051            |  229;006.051
JMP;AA                 |  224;AA    - przy błędnym odczycie
JMP;AB                 |  224;AB    - przy prawidłowym odczycie
---                    |  ---
```

Odczytane będzie 128 bajtów informacji 12 sektora 75 ścieżki. Informacja będzie zapisana w postaci bloku, począwszy od adresu

015.126. Znak końca bloku będzie zapisany w komórce 015.254. Taka będzie zawartość rejestru BADR.

006.048. Czytanie zawartości sektora do RO

Rutyna działa jak 006.051 z tym, że w rejestrze BADR jest automatycznie ustawiany adres RO. Warunki początkowe /oprócz ustawienia adresu w rejestrze BADR/, wykorzystanie rejestrów i pozostawiane ślady są takie same, jak dla rutyny 006.051.

006.196. Zapis informacji do sektora

Rutyna zapisuje w sektorze 128 bajtów informacji z pamięci operacyjnej, począwszy od adresu, ustawionego w rejestrze BADR. Numer sektora powinien być uprzednio załadowany do rejestru sektora, a numer ścieżki, na której znajduje się żądany sektor, powinien być wpisany do rejestru danych.

Rutyna posiada dwa wyjścia:

- wyjście I, negatywne, gdy:
  a/ wynik kontroli poprawności zapisu jest negatywny,
  b/ rutyna została wywołana bez zabezpieczenia warunków początkowych.
- wyjście II, pozytywne, gdy zapis został wykonany poprawnie.

Warunki początkowe:

a/ dysk jest w stanie gotowości,
b/ numer sektora i numer ścieżki zostały załadowane do odpowiednich rejestrów,
c/ adres pierwszego bajta informacji do zapisu został ustawiony w rejestr BADR.

Rutyna wykorzystuje rejestry R1, V, Z i BADR. Pozostawia takie same ślady, jak rutyny odczytu: 006.051 i 006.048.

006.193. Zapis informacji z ósmej strony do sektora

Rutyna działa jak 006.196 z tym, że informację do zapisu pobiera z ósmej strony /adres 008.001 jest automatycznie ładowany do rejestru BADR/.

Warunki początkowe /oprócz ustawienia adresu w rejestrze BADR/, wykorzystanie rejestrów i pozostawiane ślady są takie same, jak dla rutyny 006.196.

006.101. Zaadresowanie kolejnego sektora dysku

Rutyna zwiększa adres sektora według algorytmu:

a/ poprzedni numer sektora jest mniejszy od 26 - następuje zwiększenie numeru sektora o 1, numer ścieżki pozostaje bez zmiany, a rutyna zostaje zakończona przez II wyjście /pozytywne/;

b/ poprzedni numer sektora jest równy 26, a numer ścieżki jest mniejszy od 76 - następuje zwiększenie numeru ścieżki o 1, ustawienie w rejestrze sektora numeru 01 i rutyna zostaje zakończona przez II wyjście /pozytywne/;

c/ poprzedni numer sektora jest równy 26 i numer ścieżki jest równy 76 - na dysku nie ma już sektora z większym adresem. Następuje zakończenie pracy rutyny przez I wyjście /negatywne/ bez zmiany numeru ścieżki i numeru sektora.

Warunki początkowe: dysk jest w stanie gotowości, numery sektora i ścieżki były ustawione w odpowiednich rejestrach pamięci FD. Rutyna pozostawia w rejestrach R1, V, Z i LRC stany losowe.

Przykład: Dysk jest przygotowany. Odczytujemy sektor po sektorze i drukujemy odczytaną informację.

```
AA: CAL;006.048
    CAL;003.156
    CAL;006.101
    JMP;AB
    JMP;AA
AB: BUD; CC
    CAL;003.159
CC: 001; KONIEC DANYCH
```

```
AA: 229;006.048
    229;003.156
    229;006.101
    224;AB
    224;AA
    209;CC
    229;003.159
```

Uwaga.

W czasie wykonywania rutyn 006.048, 006.051, 006.193 i 006.196 głowica dysku zostaje przesunięta do ścieżki, numer której jest zapisany w rejestrze danych. Jeżeli w wyniku działania w/w rutyn głowica dysku będzie przesunięta więcej, niż o dwie ścieżki, zaleca się bezpośrednio przed wywołaniem jednej z w/w rutyn stosować podprogram PZ, przygotowujący głowicę do manewru. Podprogram PZ powinien być ładowany do pamięci operacyjnej jako część programu użytkownika.

Przykład użycia podprogramu:

```
...
LDR;012
WDAN
LDR;001
WNSK
CAL;PZ
CAL;006.048
```

```
...
232;012
119
232;001
118
229;PZ
229;006.048
```

Przewidujemy przemieszczenie głowicy ze ścieżki 00 na ścieżkę 12

Podprogram PZ:

```
PZ: 233;000
    234;00
a1: 121
    134
    207
    236
    207
    128
    226;a7
    134
    236
    207
    123
    207
    128
    226;a2
    235
    224;a1
a2: 121
    254
    119
    236
    254
    117
    232;096
    116
a3: 233;255
a4: 104
    226;a4
a5: 235
    134
    123
    207
    236
    207
    128
    233;000
    226;a9
    232;032
    116
    233;203
    134
    130
    254
    232;208
    116
a6: 132
    226;a6
    130
    133
    224;a5
a9: 121
    254
    119
    231
    231
    230
a7: 134
    123
    207
    236
    207
    128
    226;003.121
    123
    254
    117
    254
    232;064
    116
    224;a3
```

TABELA 1. Rozkazy MERY-100

Lp.	Mnemonik	Mnemonik polski	Kod dziesiętny	Kod heksadecym.	Operacja	Opis na str.
1	2	3	4	5	6	7
1	PRB	R1 → B	255	FF	Przenieś R1 do B	15
2	TIM	TIME	254	FE	Opóźnienie 1 ms.	11
3	PKR	K → R1	253	FD	Przenieś wynik do R1	16
4	ADS	"+"	251	FB	Dodawanie	14
5	PRA	R1 → A	247	F7	Przenieś R1 do A	15
6	SBS	"-"	243	F3	Odejmowanie	14
7	MZR	Z → R1	236	EC	Przenieś Z do R1	13
8	INZ	Z+1	235	EB	Zwiększ zawartość Z+1	13
9	LDZ	PO → Z	234	EA	Ładuj Z	12
10	LDV	PO → V	233	E9	Ładuj V	12
11	LDR	PO → R1	232	E8	Ładuj R1	11
12	INA	ADR+1	231	E7	Zwiększ adres powrotu o 1	23
13	EXI	POWRÓT	230	E6	Powrót z podprogramu	22
14	CAL	PODPR.	229	E5	Skok do podprogramu	21
15	AGB	SK.A > B	228	E4	Skocz, jeśli A B	16
16	ALB	SK.A < B	227	E3	Skocz, jeśli A B	17
17	REV	SK.R=V	226	E2	Skocz, jeśli R1=V	18
18	AEB	SK.A=B	225	E1	Skocz, jeśli A=B	19
19	JMP	SKOK	224	EØ	Skocz.	19
20	PRT	R1 → DR	223	DF	Drukuj R1	29
21	MRP	R1 → P	222	DE	Zapisz R1 w pamięci	27
22	MPR	P → R1	221	DD	Odczytaj z pamięci do R1	28
23	PAO	P	215	D7	Reżim adresacji z PADR-u	27
24	DEP	P-1	214	D6	Zmniejsz adres w PADR o 1	25
25	INP	P+1	213	D5	Zwiększ adres w PADR o 1	25
26	PAD	PADR	212	D4	Ładuj PADR adresem	24
27	DEB	B-1	211	D3	Zmniejsz adres w BADR o 1	26
28	INB	B+1	210	D2	Zwiększ adres w BADR o 1	26
29	BUD	BADR	209	D1	Ładuj BADR adresem	25
30	BUF	B	208	DØ	Reżim adresacji z BADR-u	27
31	CLR	LLRC	207	CF	Licz LRC	33
32	SWT	PRZEŁĄCZ	206	CE	Przełącz PK	44

1	2	3	4	5	6	7
33	BER	P.CZT	205	CD	Początek odczytu z PK	41
34	ENW	K.PSZ	204	CC	Koniec zezwolenia na zapis	39
35	BEW	P.PSZ	203	CB	Początek zezwolenia na zapis	39
36	DRN	PRZ	202	CA	Przewiń taśmę w PK	36
37	ENS	K RS	201	C9	Koniec ruchu szybkiego	38
38	BES	PRS	200	C8	Początek ruchu szybkiego	38
39	ENB	KRW	199	C7	Koniec ruchu taśmy w tył	38
40	BEB	PRW	198	C6	Początek ruchu taśmy w tył	37
41	ENP	KRP	197	C5	Koniec ruchu taśmy w przód	37
42	BEP	PRP	196	C4	Początek ruchu taśmy wprzód	37
43	TME	KPO	195	C3	Odłącz PK	36
44	TMB	PPO	194	C2	Podłącz PK	35
45	STR	K.REZ	193	C1	Koniec rezerwacji PK	35
46	BTR	P.REZ	192	CØ	Początek rezerwacji PK	35
47	LOF6	KZ6	191	BF	Zgaś lampkę 6	32
48	LONG	PZ6	190	BE	Zapal lampkę 6	32
49	LOF5	KZ5	189	BD	Zgaś lampkę 5	32
50	LON5	PZ5	188	BC	Zapal lampkę 5	32
51	LOF4	KZ4	187	BB	Zgaś lampkę 4	32
52	LON4	PZ4	186	BA	Zapal lampkę 4	32
53	LOF3	KZ3	185	B9	Zgaś lampkę 3	32
54	LON3	PZ3	184	B8	Zapal lampkę 3	32
55	LOF2	KZ2	183	B7	Zgaś lampkę 2	32
56	LON2	PZ2	182	B6	Zapal lampkę 2	32
57	LOF1	KZ1	181	B5	Zgaś lampkę 1	32
58	LON1	PZ1	180	B4	Zapal lampkę 1	32
59	KAC	KAC	178	B2	Koniec aktywności klawiatury	30
60	PAC	PAC	177	B1	Uaktywnienie klawiatury	30
61	RKL	KL→R1	176	BØ	Ładuj kod z klawiatury	30
62	LOF12	KZ12	171	AB	Zgaś lampkę 12/FD/	32
63	LON12	PZ12	170	AA	Zapal lampkę 12/FD/	32

1	2	3	4	5	6	7
64	LOF11	KZ11	169	A9	Zgaś lampkę 11/FB/	32
65	LON11	PZ11	168	A8	Zapal lampkę 11/FB/	32
66	LOF10	KZ10	167	A7	Zgaś lampkę 10/L/	32
67	LON10	PZ10	166	A6	Zapal lampkę 10/L/	32
68	LOF9	KZ9	165	A5	Zgaś lampkę 9	32
69	LON9	PZ9	164	A4	Zapal lampkę 9	32
70	LOF8	KZ8	163	A3	Zgaś lampkę 8	32
71	LON8	PZ8	168	A2	Zapal lampkę 8	32
72	LOF7	KZ7	161	A1	Zgaś lampkę 7	32
73	LON7	PZ7	160	AØ	Zapal lampkę 7	32
74	CRS	CRS	152	98	Gaszenie 6-go bitu statusu linii	61
75	RS	RS	149	95	Status linii	60
76	AR	AR	148	94	Adres terminala	59
77	STU	STU	148	93	Status USART-a	58
78	ODB	ODB	146	92	Przyjęcie/odbiór/ bajta do R1	58
79	NAD	NAD	145	91	Wysłanie bajta danych	57
80	ROZ	ROZ	144	9Ø	Wpisz słowo sterowania transmisja	56
81	KDP	KDP	143	8F	Koniec wyczytywania ekranu	48
82	DPX	DP →X	142	8E	Wybierz bajt danych z monitora	47
83	PDP	PDP	141	8D	Początek wyczytywania ekranu	46
84	RDP	R1→DP	140	8C	Zanieś R1 na ekran monitora	45
85	INT	INT	139	8B	Początek pracy monitora	44
86	MXR	X →R1	138	8A	Załaduj bajt z monitora do R1	47
87	REK	ZER.PK	134	86	Zeruj PK	34
88	END	K.DAN	133	85	Koniec danych do zapisu	40
89	STS	ST.szcz.	132	84	Status szczegółowy PK	43
90	STA	STAT	131	83	Status PK	42
91	MRK	R1 →PK	130	82	Zapisz bajt z R1 w PK	40
92	MKR	PK →R1	129	81	Czytaj bajt z PK do R1	41
93	LRC	LRC	128	BØ	Przenieś LRC do R1	34

1	2	3	4	5	6	7
94	CDAN	CDAN	123	7B	Czytaj dane	56
95	CNSK	CNSK	122	7A	Czytaj numer sektora	55
96	CNSC	CNSC	121	79	Czytaj numer ścieżki	55
97	SFD	CSTAT	120	78	Status FD	51
98	WDAN	WDAN	119	77	Wpisz dane	54
99	WNSK	WNSK	118	76	Wpisz numer sektora	54
100	WNSC	WNSC	117	75	Wpisz numer ścieżki	53
101	WROZ	WROZ	116	74	Wpisz rozkaz	53
102	KZZ	KZZ	109	6C	Odwołanie zezwolenia na zapis	55
103	STFD4	STFD-4	108	6B	Czwarty status FD	53
104	ZFD	ZER.FD	106	6A	Zeruj FD	49
105	STFD3	STFD-3	105	69	Trzeci status FD	52
106	STFD2	STFD-2	104	68	Drugi status FD	52
107	STFD1	STFD-1	103	67	Pierwszy status FD	51
108	PZZ	PZZ	102	66	Zezwolenie na zapis w FD	54
109	DPR	DPR	099	63	Wybierz dysk prawy	50
110	DLW	DLW	098	62	Wybierz dysk lewy	50
111	KRUN	KRUN	097	61	Odłącz wybrany FD	50
112	PRUN	PRUN	096	6Ø	Wybierz FD numer...	49
113	UNG	UNG	091	5B	Ustaw reżim NRZ	65
114	OSTB	OSTB	090	5A	Ostatni bajt do zapisu	64
115	SPT	SPT	088	58	Status pamięci taśmowej/PT/	66
116	NRZF	NRZF	087	57	Podłączenie PT	61
117	KAS	KAS	086	56	Kasowanie	63
118	REW	PRZEW.	085	55	Przewijanie taśmy do początku	64
119	PST	PST	084	54	Odczyt lub zapis	62
120	ST	STB	083	53	Strob do wykonania rozkazu	65
121	PPSN	PPSN	082	52	Pozycjonowanie głowicy PT	63
122	PZT	PZT	081	51	Pisz znacznik taśmy	64
123	WST	WST	080	5Ø	Kierunek ruchu taśmy	62

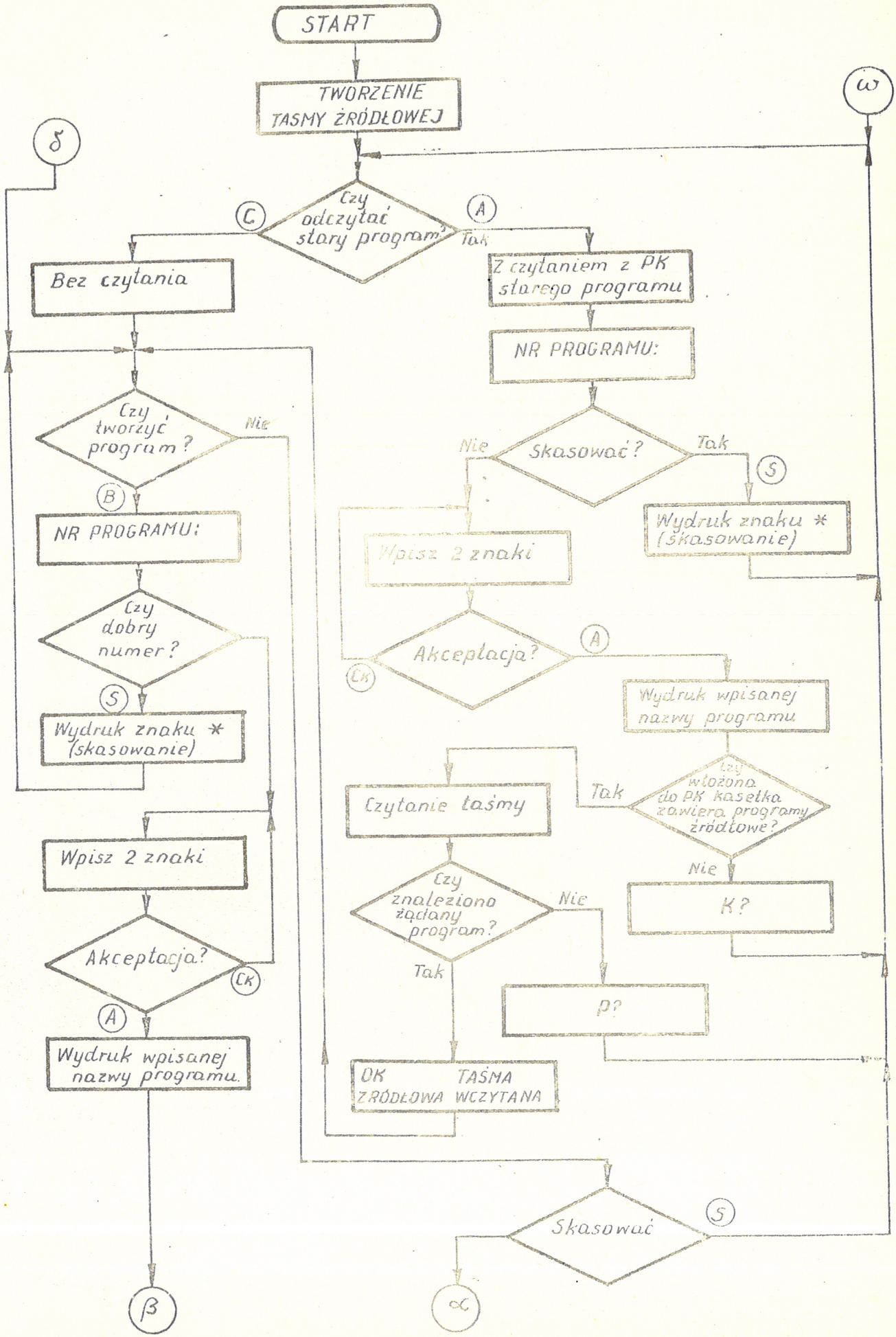
Tablica 2. KODY ASCII /klawiatury, drukarki, monitora i teletransmisji/.

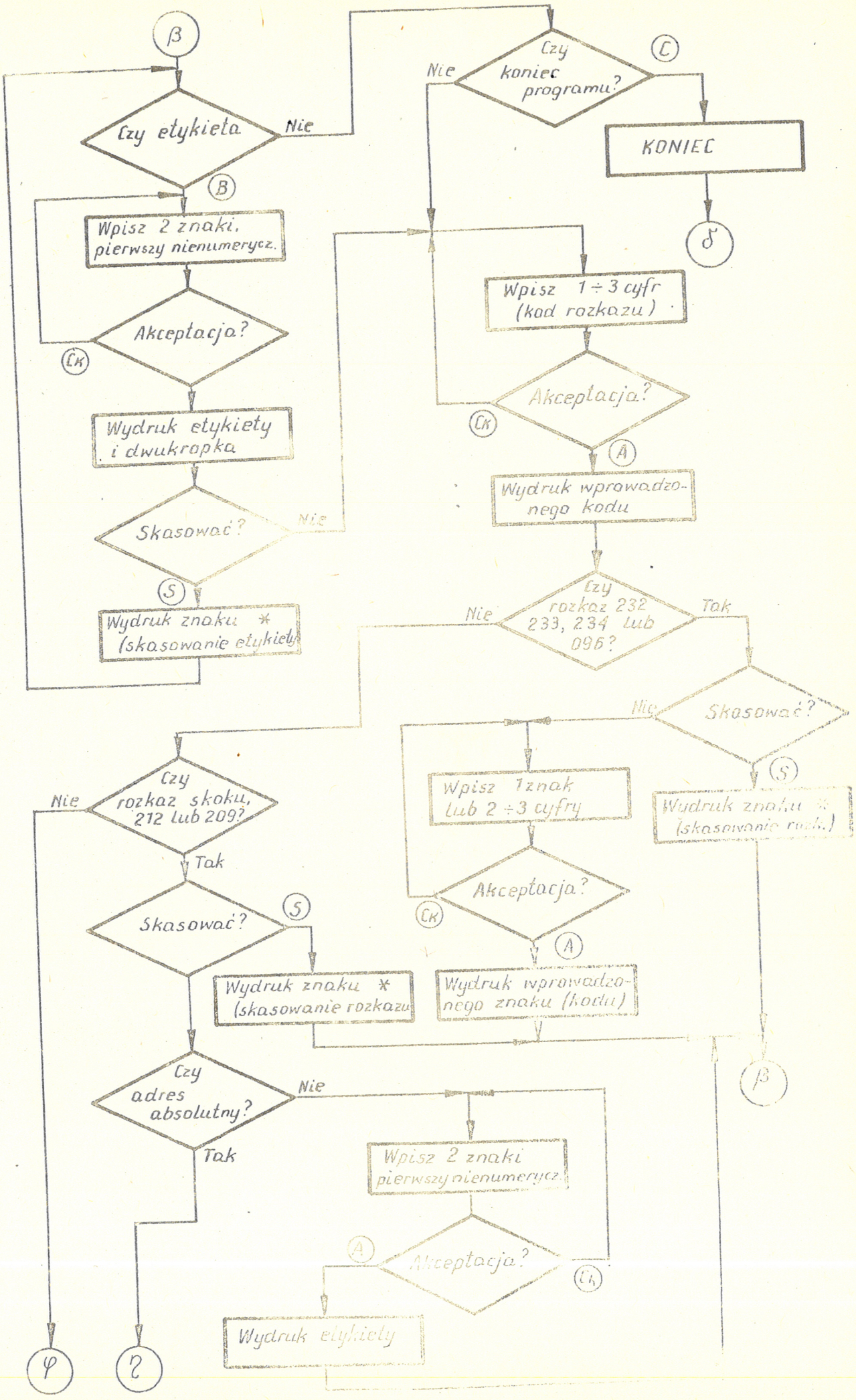
Kod znaku lub funkcji	Klawiatura	Drukarka	Monitor /VDU lub CRT/	Tele-transmisja	Kod znaku lub funkcji	Klawiatura i drukarka	Teletransm. i monitor	Kod znaku lub funkcji	Klaw.,druk., CRT i teletr.	Monitor VDU	Kod znaku lub funkcji	Klaw.,druk., CRT i teletr.	Monitor VDU
255				NUL	223	Sp	Sp	191	@	@	159	`	@
254	Ⓐ			SOH	222	!	!	190	A	A	158	a	A
253	Ⓑ			STX	221	"	"	189	B	B	157	b	B
252	Ⓒ			ETX	220	#	#	188	C	C	156	c	C
251	◊			EOT	219	¤	$	187	D	D	155	d	D
250	*			ENQ	218	%	%	186	E	E	154	e	E
249	Ⓛ			ACK	217	&	&	185	F	F	153	f	F
248				BEL	216	'	'	184	G	G	152	g	G
247	LK		BS	BS	215	(	(	183	H	H	151	h	H
246	HT	HT	HT	HT	214	)	)	182	I	I	150	i	I
245	LF	LF	LF	LF	213	*	*	181	J	J	149	j	J
244	Ⓢ	VT	VT	VT	212	+	+	180	K	K	148	k	K
243	ST	FF	FF	FF	211	,	,	179	L	L	147	l	L
242		CR	CR	CR	210	-	-	178	M	M	146	m	M
241			SO	SO	209	.	.	177	N	N	145	n	N
240			CI	CI	208	/	/	176	O	O	144	o	O
239			CD	DLE	207	0	Ø	175	P	P	143	p	P
238			DC1	DC1	206	1	1	174	Q	Q	142	q	Q
237			DC2	DC2	205	2	2	173	R	R	141	r	R
236			DC3	DC3	204	3	3	172	S	S	140	s	S
235			DC4	DC4	203	4	4	171	T	T	139	t	T
234				NAK	202	5	5	170	U	U	138	u	U
233			LI	SYN	201	6	6	169	V	V	137	v	V
232			LD	ETB	200	7	7	168	W	W	136	w	W
231	00			CAN	199	8	8	167	X	X	135	x	X
230	Ck			EM	198	9	9	166	Y	Y	134	y	Y
229	BS			SUB	197	:	:	165	Z	Z	133	z	Z
228				ESC	196	;	;	164	[	[	132	{	[
227				FS	195	<	<	163	\	~	131	!	~
226				GS	194	=	=	162	]	]	130	}	]
225				RS	193	>	>	161	^	{	129	~	{
224				US	192	?	?	160	_	}	128	◊	}

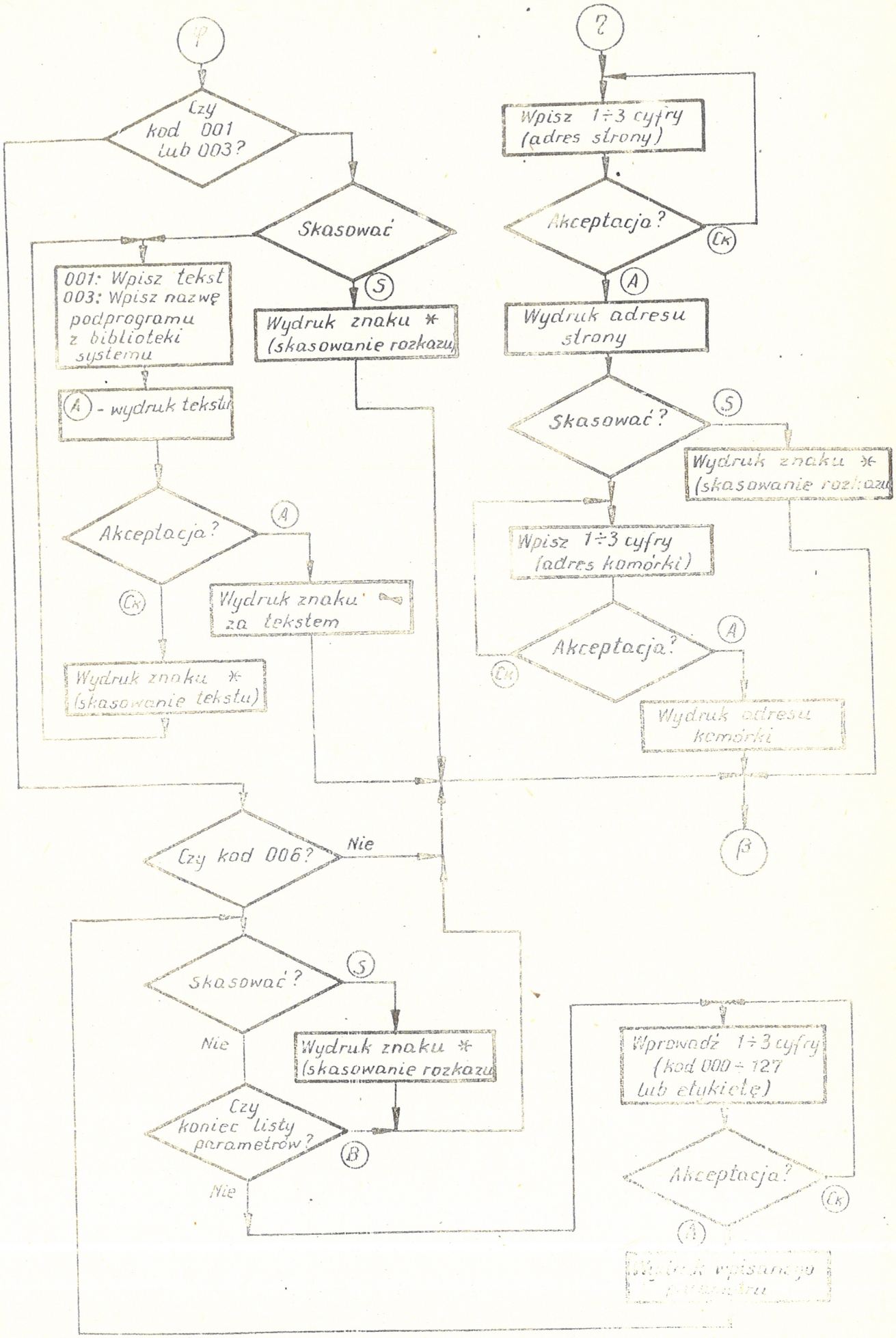
Funkcje i sygnały w kodzie ASCII

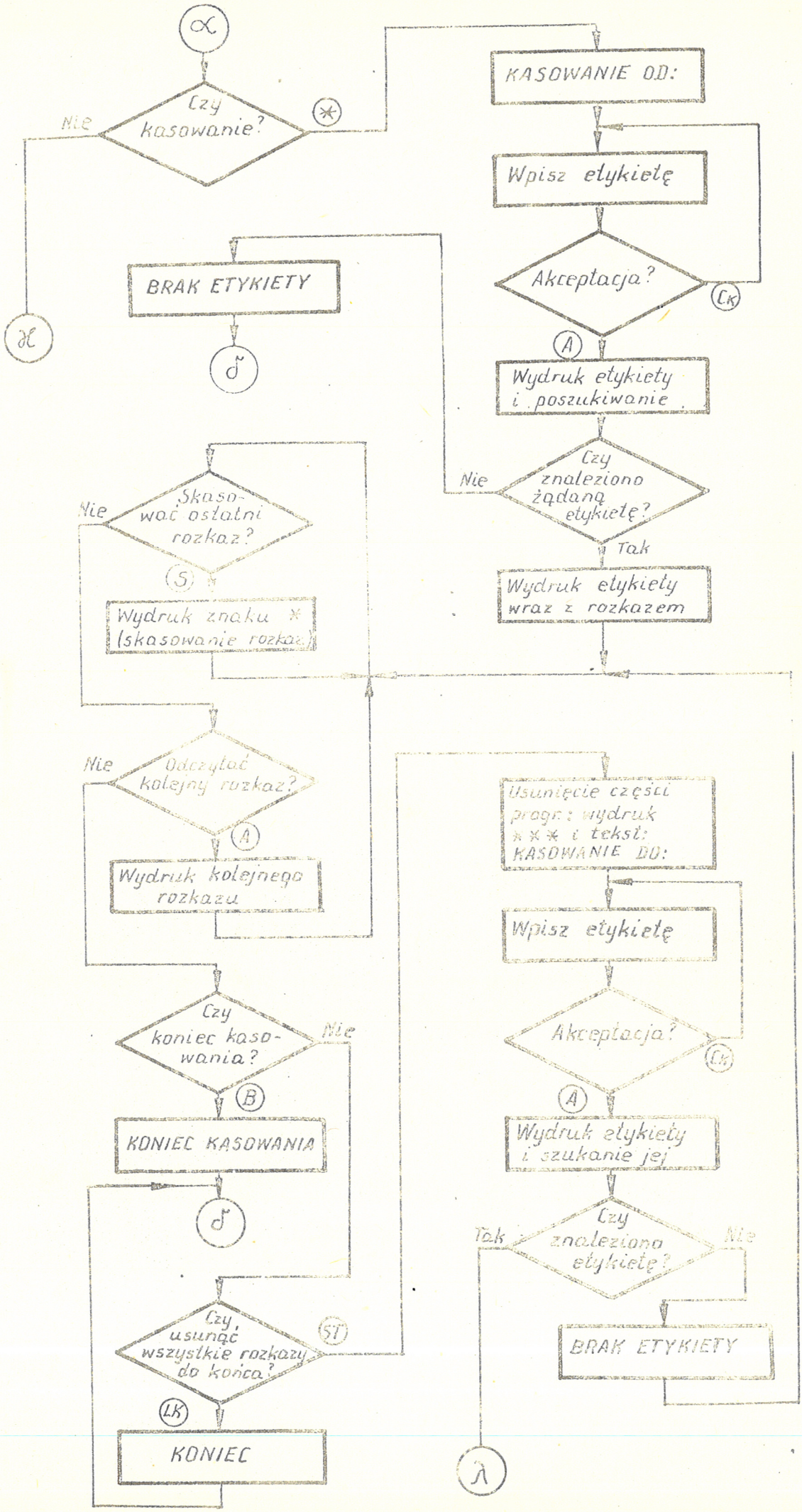
NUL - brak informacji
SOH - początek nagłówka
STX - początek tekstu
ETX - koniec tekstu
EOT - koniec transmisji
ENQ - zapytanie
ACK - potwierdzenie odbioru
BEL - dzwonek
BS - cofanie
HT - tabulacja pozioma
LF - zmiana wiersza
VT - tabulacja pionowa
FF - zmiana strony
CR - powrót karetki
SO - poza kodem
SI - w kodzie
DLE - zmiana nadzoru ciągu
DC1 - Sterowanie urządzeniem nr: 1
DC2 - Sterowanie urządzeniem nr: 2
DC3 - Sterowanie urządzeniem nr: 3
DC4 - Sterowanie urządzeniem nr: 4
NAK - brak potwierdzenia odbioru
SYN - znak synchronizacji
ETB - koniec transmisji bloku dnych
CAN - anulowanie
EM - koniec danych na nośniku
SUB - zastąpić
ESC - zmiana zestawu znaków
FS - separator fajla
GS - separator grupy
RS - separator zapisów
US - separator jednostek
DEL - kasowanie

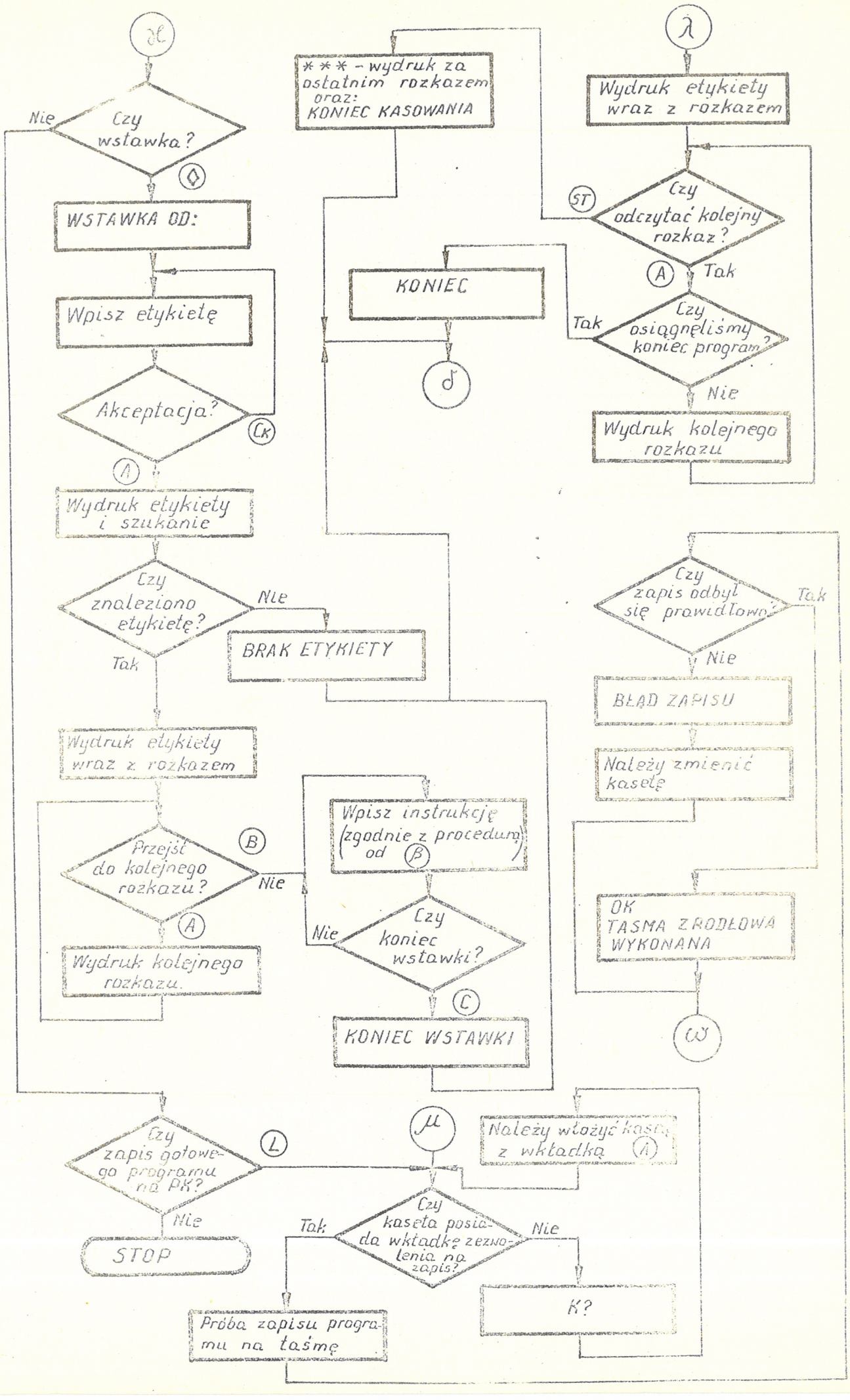
Załącznik 1.


START
TWORZENIE TASMY ŹRÓDŁOWEJ
ω
δ
Czy odczytać stary program?
C
A
Tak
Bez czytania
Z czytaniem z PK starego programu
NR PROGRAMU:
Czy tworzyć program?
Nie
Nie
Skasować?
Tak
S
B
NR PROGRAMU:
Wydruk znaku * (skasowanie)
Wpisz 2 znaki
Czy dobry numer?
Akceptacja?
A
CK
S
Wydruk znaku * (skasowanie)
Wydruk wpisanej nazwy programu
Czytanie taśmy
Tak
Czy włożona do PK kasetka zawiera programy źródłowe?
Wpisz 2 znaki
Nie
K?
Akceptacja?
Czy znaleziono żądany program?
Nie
CK
Tak
A
P?
Wydruk wpisanej nazwy programu
OK TAŚMA ŹRÓDŁOWA WCZYTANA
Skasować
S
β
α


β
Czy etykieta
Nie
B
Wpisz 2 znaki, pierwszy nienumeryczn.
Akceptacja?
CK
Wydruk etykiety i dwukropka
Skasować?
Nie
S
Wydruk znaku * (skasowanie etykiety)
Czy koniec programu?
Nie
C
KONIEC
δ
Wpisz 1÷3 cyfr (kod rozkazu)
Akceptacja?
CK
A
Wydruk wprowadzonego kodu
Czy rozkaz 232, 233, 234 lub 096?
Nie
Tak
Skasować?
Nie
S
Wpisz 1 znak lub 2÷3 cyfry
Akceptacja?
CK
A
Wydruk wprowadzonego znaku (kodu)
Wydruk znaku * (skasowanie rozk.)
Czy rozkaz skoku, 212 lub 209?
Nie
Tak
Skasować?
S
Wydruk znaku * (skasowanie rozkazu)
β
Czy adres absolutny?
Nie
Tak
Wpisz 2 znaki pierwszy nienumeryczn.
A
Akceptacja?
CK
Wydruk etykiety
φ
γ


γ
Czy kod 001 lub 003?
Skasować
S
001: Wpisz tekst
003: Wpisz nazwę podprogramu z biblioteki systemu
A - wydruk tekstu
Akceptacja?
A
Ck
Wydruk znaku za tekstem
Wydruk znaku * (skasowanie tekstu)
Wydruk znaku * (skasowanie rozkazu)
?
Wpisz 1÷3 cyfry (adres strony)
Akceptacja?
Ck
A
Wydruk adresu strony
Skasować?
S
Wydruk znaku * (skasowanie rozkazu)
Wpisz 1÷3 cyfry (adres komórki)
Akceptacja?
A
Ck
Wydruk adresu komórki
β
Czy kod 006?
Nie
Skasować?
S
Nie
Wydruk znaku * (skasowanie rozkazu)
Czy koniec listy parametrów?
B
Nie
Wprowadź 1÷3 cyfry (kod 000÷127 lub etykietę)
Akceptacja?
Ck
A
Wydruk wpisanego parametru


α
Czy kasowanie?
Nie
*
KASOWANIE OD:
Wpisz etykietę
Akceptacja?
CK
A
Wydruk etykiety i poszukiwanie
Czy znaleziono żądaną etykietę?
Nie
Tak
Wydruk etykiety wraz z rozkazem
BRAK ETYKIETY
δ
Skasować ostatni rozkaz?
Nie
S
Wydruk znaku * (skasowanie rozkazu)
Odczytać kolejny rozkaz?
Nie
A
Wydruk kolejnego rozkazu
Czy koniec kasowania?
Nie
B
KONIEC KASOWANIA
δ
Czy usunąć wszystkie rozkazy do końca?
ST
LK
KONIEC
Usunięcie części progr.: wydruk *** i tekst: KASOWANIE DO:
Wpisz etykietę
Akceptacja?
CK
A
Wydruk etykiety i szukanie jej
Czy znaleziono etykietę?
Tak
Nie
BRAK ETYKIETY
λ


κ
Czy wstawka?
Nie
Q
WSTAWKA OD:
Wpisz etykietę
Akceptacja?
Ck
A
Wydruk etykiety i szukanie
Czy znaleziono etykietę?
Nie
BRAK ETYKIETY
Tak
Wydruk etykiety wraz z rozkazem
Przejść do kolejnego rozkazu?
B
Nie
A
Wydruk kolejnego rozkazu.
Wpisz instrukcję (zgodnie z procedurą od β)
Czy koniec wstawki?
Nie
C
KONIEC WSTAWKI
*** - wydruk za ostatnim rozkazem oraz: KONIEC KASOWANIA
KONIEC
δ
λ
Wydruk etykiety wraz z rozkazem
ST
Czy odczytać kolejny rozkaz?
A
Tak
Tak
Czy osiągnęliśmy koniec programu?
Nie
Wydruk kolejnego rozkazu
Czy zapis odbył się prawidłowo?
Tak
Nie
BŁĄD ZAPISU
Należy zmienić kasetę
OK TASMA ZRODŁOWA WYKONANA
ω
μ
Należy włożyć kasetę z wkładką A
Czy zapis gotowego programu na PK?
L
Nie
STOP
Czy kaseta posiada wkładkę zezwolenia na zapis?
Tak
Nie
Próba zapisu programu na taśmę
K?

Załącznik 2

Strona *000*

Nr komórki	Kodowanie dziesiętnie	Kodowanie mnemonika	Nr komórki	Kodowanie dziesiętnie	Kodowanie mnemonika	Nr komórki	Kodowanie dziesiętnie	Kodowanie mnemonika	Nr komórki	Kodowanie dziesiętnie	Kodowanie mnemonika
000	000	000	032	176	RKL	064	054	054	096	221	MZR
001	164	LON9	033	233	LDV	065	208	BUF	097	226	REV
002	209	BUD	034	254	A	066	221	MPR	098	001	001
003	008	008	035	226	REV	067	226	REV	099	038	038
004	000	000	036	010	010	068	000	000	100	247	PRA
005	208	BUF	037	002	002	069	074	074	101	253	PKR
006	232	LDR	038	229	CAL	070	210	INB	102	222	MRP
007	224	224	039	004	004	071	224	JMP	103	211	DEB
008	222	MRP	040	041	041	072	000	000	104	214	DEP
009	210	INB	041	224	JMP	073	065	065	105	224	JMP
010	222	MRP	042	000	000	074	236	MZR	106	000	000
011	210	INB	043	031	031	075	233	LDV	107	089	089
012	222	MRP	044	232	LDR	076	207	207	108	232	LDR
013	176	RKL	045	255	255	077	226	REV	109	207	207
014	176	RKL	046	214	DEP	078	000	000	110	255	PRB
015	233	LDV	047	215	PAO	079	084	084	111	215	PAO
016	249	L	048	222	MRP	080	251	ADS	112	221	MPR
017	226	REV	049	211	DEB	081	224	JMP	113	226	REV
018	005	005	050	208	BUF	082	000	000	114	001	001
019	000	000	051	222	MRP	083	085	085	115	069	069
020	233	LDV	052	213	INP	084	243	SBS	116	247	PRA
021	247	LK	053	210	INB	085	233	LDV	117	253	PKR
022	226	REV	054	215	PAO	086	255	255	118	222	MRP
023	006	006	055	221	MPR	087	211	DEB	119	214	DEP
024	090	090	056	233	LDV	088	214	DEP	120	224	JMP
025	229	CAL	057	255	255	089	208	BUF	121	000	000
026	004	004	058	226	REV	090	221	MPR	122	111	111
027	041	041	059	000	000	091	226	REV	123	224	JMP
028	224	JMP	060	065	065	092	000	000	124	001	001
029	000	000	061	213	INP	093	108	108	125	038	038
030	000	000	062	224	JMP	094	255	PRB	126	000	PRA
031	176	RKL	063	000	000	095	215	PAO	127	232	LDR

Nr komórki	Kodowanie dziesiętnie	Kodowanie mnemonika	Nr komórki	Kodowanie dziesiętnie	Kodowanie mnemonika	Nr komórki	Kodowanie dziesiętnie	Kodowanie mnemonika	Nr komórki	Kodowanie dziesiętnie	Kodowanie mnemonika
128	001	001	160	233	LDV	192	230	EXI	224	255	255
129	000	000	161	219	219	193	212	PAD	225	215	PAO
130	166	LON10	162	226	REV	194	009	009	226	222	MRP
131	194	TMB	163	003	003	195	000	000	227	213	INP
132	203	BEW	164	118	118	196	215	PAO	228	210	INB
133	196	BEP	165	235	INZ	197	221	MPR	229	224	JMP
134	234	LDZ	166	236	MZR	198	233	LDV	230	000	000
135	255	255	167	233	LDV	199	255	255	231	216	216
136	224	JMP	168	255	255	200	226	REV	232	232	LDR
137	001	001	169	205	BER	201	000	000	233	255	255
138	120	120	170	226	REV	202	207	207	234	214	DWP
139	232	LDR	171	000	000	203	213	INP	235	215	PAO
140	206	206	172	176	176	204	224	JMP	236	222	MRP
141	255	PRB	173	224	JMP	205	000	000	237	213	INP
142	251	ADS	174	000	000	206	196	196	238	221	MPR
143	232	LDR	175	148	148	207	214	DEP	239	233	LDV
144	207	207	176	253	MKR	208	235	INZ	240	255	255
145	247	PRA	177	233	LDV	209	236	MZR	241	226	REV
146	234	LDZ	178	201	201	210	226	REV	242	000	000
147	255	255	179	226	REV	211	000	000	243	249	249
148	254	TIM	180	001	001	212	216	216	244	235	INZ
149	132	STS	181	223	223	213	224	JMP	245	213	INP
150	233	LDV	182	224	JMP	214	000	000	246	224	JMP
151	255	255	183	000	000	215	207	207	247	000	000
152	226	REV	184	145	145	216	215	PAO	248	238	238
153	000	000	185	231	INA	217	221	MPR	249	236	MZR
154	185	185	186	231	INA	218	208	BUF	250	226	REV
155	254	TIM	187	231	INA	219	222	MRP	251	004	004
156	132	STS	188	224	JMP	220	226	REV	252	054	054
157	226	REV	189	003	003	221	004	004	253	224	JMP
158	000	000	190	118	118	222	054	054	254	001	001
159	185	185	191	231	INA	223	232	LDR	255	197	197

Nr komórki	Kodowanie dziesiętnie	Kodowanie mnemonika	Nr komórki	Kodowanie dziesiętnie	Kodowanie mnemonika	Nr komórki	Kodowanie dziesiętnie	Kodowanie mnemonika	Nr komórki	Kodowanie dziesiętnie	Kodowanie mnemonika
000	136	LON10	032	024	024	064	253	PKR	096	222	MRP
001	134	REK	033	197	ENP	065	222	MRP	097	214	DEP
002	192	BTR	034	167	LOF10	066	224	JMP	098	214	DEP
003	194	TMB	035	231	INA	067	000	000	099	221	MPR
004	202	DRN	036	231	INA	068	103	103	100	226	REV
005	131	STA	037	230	EXI	069	232	LDR	101	001	001
006	233	LDV	038	232	LDR	070	207	207	102	106	106
007	252	252	039	255	255	071	247	PRA	103	224	JMP
008	226	REV	040	247	PRA	072	253	PKR	104	001	001
009	001	001	041	215	PAO	073	233	LDV	105	095	095
010	014	014	042	213	INP	074	206	206	106	213	INP
011	224	JMP	043	221	MPR	075	226	REV	107	232	LDR
012	001	001	044	226	REV	076	001	001	108	206	206
013	005	005	045	001	001	077	081	081	109	222	MRP
014	196	BEB	046	050	050	078	231	INA	110	231	INA
015	131	STA	047	224	JMP	079	231	INA	111	231	INA
016	233	LDV	048	001	001	080	230	EXI	112	230	EXI
017	254	254	049	041	041	081	213	INP	113	129	MKR
018	226	REV	050	232	LDR	082	233	LDV	114	207	CLR
019	001	001	051	255	255	083	255	255	115	234	LDZ
020	024	024	052	213	INP	084	215	PAO	116	001	001
021	224	JMP	053	222	MRP	085	213	INP	117	224	JMP
022	001	001	054	214	DEP	086	221	MPR	118	002	002
023	015	015	055	214	DEP	087	226	DEV	119	246	246
024	131	STA	056	221	MPR	088	001	001	120	229	CAL
025	233	LDV	057	226	REV	089	093	093	121	003	003
026	252	252	058	001	001	090	224	JMP	122	061	061
027	226	REV	059	063	063	091	001	001	123	134	REK
029	001	001	060	224	JMP	092	085	085	124	167	LOF10
029	033	033	061	001	001	093	232	LDR	125	231	INA
030	224	JMP	062	052	052	094	255	255	126	231	INA
031	001	001	063	213	INP	095	213	INP	127	230	EXI

Nr komórki	Kodowanie dziesiętnie	Kodowanie mnemonika	Nr komórki	Kodowanie dziesiętnie	Kodowanie mnemonika	Nr komórki	Kodowanie dziesiętnie	Kodowanie mnemonika	Nr komórki	Kodowanie dziesiętnie	Kodowanie mnemonika
128	166	LON10	160	208	BUF	192	000	000	224	207	CLR
129	134	REK	161	129	MKR	193	224	JMP	225	209	BUD
130	192	BTR	162	233	LDV	194	002	002	226	010	010
131	194	TMB	163	223	223	195	103	103	227	000	000
132	208	BUF	163	132	STS	196	000	000	228	208	BUF
133	209	BUD	165	226	REV	197	232	LDR	229	232	LDR
134	009	009	166	001	001	198	255	255	230	255	225
135	000	000	167	164	164	199	213	INP	231	222	MRP
136	196	BEP	168	233	IDV	200	222	MRP	232	231	INA
137	234	LDZ	169	255	255	201	214	DEP	233	231	INA
138	215	215	170	226	REV	202	214	DEP	234	230	EXI
139	229	CAL	171	002	002	203	221	MPR	235	208	BUF
140	003	003	172	114	114	204	226	REV	236	231	INA
141	061	061	173	211	DEB	205	001	001	237	231	INA
142	130	MRK	174	211	DEB	206	210	210	238	230	EXI
143	233	LDV	175	232	LDR	207	224	JMP	239	209	BUD
144	203	203	176	255	255	208	001	001	240	008	008
145	132	STS	177	222	MRP	209	199	199	241	255	255
146	226	REV	178	234	LDZ	210	232	LDR	242	221	MPR
147	001	001	179	215	215	211	207	207	243	233	LDV
148	145	145	180	229	CAL	212	213	INP	244	000	000
149	133	END	181	002	002	213	222	MRP	245	226	REV
150	205	BER	182	187	187	214	235	INZ	246	004	004
152	229	CAL	183	197	ENP	215	221	MRP	247	037	037
152	000	000	184	234	LDZ	216	226	REV	248	231	INA
153	139	139	185	050	050	217	000	000	249	231	INA
154	224	JMP	186	229	CAL	218	249	249	250	230	EXI
155	002	002	187	002	002	219	213	INP	251		
156	148	148	188	187	187	220	224	JMP	252		
157	224	JMP	187	204	ENW	221	001	001	253		
158	001	001	190	212	PAD	222	215	215	254		
159	128	128	191	009	009	223	134	REK	255		

Nr komórki	Kodowanie dziesiętnie	Kodowanie mnemonika	Nr komórki	Kodowanie dziesiętnie	Kodowanie mnemonika	Nr komórki	Kodowanie dziesiętnie	Kodowanie mnemonika	Nr komórki	Kodowanie dziesiętnie	Kodowanie mnemonika
000	212	PAD	032	205	BER	064	002	002	096	224	JMP
001	008	008	033	213	INP	065	035	035	097	003	003
002	000	000	034	221	MPR	066	214	DEP	098	015	015
003	166	LON10	035	130	MRK	067	221	MPR	099	211	DEB
004	209	BUD	036	213	INP	068	233	LDV	100	224	JMP
005	010	010	037	233	LDV	069	085	085	101	003	003
006	000	000	038	254	254	070	226	REV	102	034	034
007	208	BUF	039	131	STA	071	002	002	103	215	PAO
008	232	LDR	040	226	REV	072	158	158	104	221	MPR
009	000	000	041	002	002	073	224	JMP	105	233	LDV
010	222	MRP	042	046	046	072	002	002	106	085	085
011	210	INB	043	229	CAL	075	033	033	107	167	LCF10
012	222	MRP	044	002	002	076	209	BUD	108	226	REV
013	134	REK	045	076	076	077	010	010	109	002	002
014	194	TMB	046	233	LDV	078	001	001	110	153	153
015	129	MKR	047	207	207	079	208	BUF	111	224	JMP
016	203	BEW	048	132	STS	080	232	LDR	112	001	001
017	196	BEP	049	226	REV	081	001	001	113	235	235
018	234	LDZ	050	002	002	082	222	MRP	114	129	MKR
019	155	155	051	048	048	083	215	PAO	115	207	CLR
020	229	CAL	052	233	LDV	084	231	INA	116	222	MRP
021	002	002	053	239	239	085	231	INA	117	210	INB
022	187	187	054	226	REV	086	230	EXI	118	224	JMP
023	215	PAO	055	001	001	087	210	INB	119	001	001
024	221	MPR	056	113	113	088	221	MPR	120	162	162
025	130	MRK	057	233	LDV	089	233	LDV	121	166	LON10
026	233	LDV	058	255	255	090	255	255	122	134	REK
027	203	203	059	221	MPR	091	226	REV	123	194	TMB
028	132	STS	060	226	REV	092	002	002	124	198	BFB
029	226	REV	061	002	002	093	099	099	125	234	LDZ
030	002	002	062	066	066	094	211	DEB	126	225	225
031	028	028	063	224	JMP	095	221	MPR	127	229	CAL

Nr komórki	Kodowanie dziesiętnie	Kodowanie mnemonika	Nr komórki	Kodowanie dziesiętnie	Kodowanie mnemonika	Nr komórki	Kodowanie dziesiętnie	Kodowanie mnemonika	Nr komórki	Kodowanie dziesiętnie	Kodowanie mnemonika
128	003	003	160	000	000	192	002	002	224	209	BUD
129	061	061	161	000	000	193	197	197	225	009	009
130	205	BER	162	233	LDV	194	229	CAL	226	000	000
131	234	LDZ	163	001	001	195	002	002	227	224	JMP
132	205	205	164	236	MZR	196	076	076	228	000	000
133	131	STA	165	226	REV	197	235	INZ	229	044	044
134	233	LDV	166	002	002	198	236	MZR	230	212	PAD
135	252	252	167	180	180	199	226	REV	231	009	009
136	226	REV	168	229	CAL	200	003	003	232	000	000
137	001	001	169	000	000	201	121	121	233	224	JMP
138	223	223	170	139	139	202	224	JMP	234	000	000
139	132	STS	171	000	000	203	002	002	235	044	044
140	233	LDV	172	000	000	204	187	187	236	000	000
141	235	235	173	000	000	205	212	PAD	237	209	BUD
142	226	REV	174	224	JMP	206	009	009	238	009	009
143	001	001	175	001	001	207	000	000	239	000	000
144	120	120	176	173	173	208	224	JMP	240	224	JMP
145	224	JMP	177	224	JMP	209	000	000	241	004	004
146	002	002	178	001	001	210	232	232	242	057	057
147	133	133	179	160	160	211	209	BUD	243	212	PAD
148	193	STR	180	232	LDR	212	009	009	244	009	009
149	192	BTR	181	085	085	213	000	000	245	000	000
150	224	JMP	182	207	CLR	214	224	JMP	246	224	JMP
151	002	002	183	208	BUF	215	003	003	247	004	004
152	107	107	184	224	JMP	216	072	072	248	057	057
153	207	CLR	185	001	001	217	212	PAD	249	207	CLR
154	213	INP	186	162	162	218	009	009	250	009	009
155	224	JMP	187	254	TIM	219	000	000	251	000	000
156	002	002	188	233	LDV	220	224	JMP	252	224	JMP
157	211	211	189	254	254	221	003	003	253	003	003
158	133	END	190	131	STA	222	072	072	254	072	072
159	209	BUD	191	226	REV	223	000	000	255		

Nr komórki	Kodowanie		Nr komórki	Kodowanie		Nr komórki	Kodowanie		Nr komórki	Kodowanie	
	dziesiętnie	mnemonika		dziesiętnie	mnemonika		dziesiętnie	mnemonika		dziesiętnie	mnemonika
000	212	PAD	032	003	003	064	233	LDV	096	003	003
001	009	009	033	037	037	065	255	255	097	109	109
002	000	000	034	211	DEB	066	226	REV	098	210	INB
003	209	BUD	035	221	MPR	067	003	003	099	213	INP
004	008	008	036	207	CLR	067	121	121	100	128	LRC
005	000	000	037	215	PAO	069	224	JMP	101	233	LDV
006	208	BUF	038	221	MPR	070	003	003	102	000	000
007	134	REK	039	233	LDV	071	061	061	103	226	REV
008	210	INB	040	255	255	072	215	PAD	104	003	003
009	221	MPR	041	208	BUF	073	221	MPR	105	086	086
010	233	LDV	042	226	REV	074	208	BUF	106	224	JMP
011	085	085	043	003	003	075	222	MRP	107	003	003
012	226	REV	044	052	052	ż76	233	LDV	108	121	121
013	003	003	045	222	MRP	077	255	255	109	128	LRC
014	034	034	046	207	CLR	078	226	REV	110	233	LDV
015	207	CLR	047	210	INB	079	003	003	111	000	000
016	224	JMP	048	213	INP	080	121	121	112	226	REV
017	003	003	049	224	JMP	081	213	INP	113	003	003
018	008	008	050	003	003	082	210	INB	114	118	118
019	212	PAD	051	037	037	083	224	JMP	115	224	JMP
020	009	009	052	128	LRC	084	003	003	116	003	003
021	000	000	053	222	MRP	085	072	072	117	121	121
022	209	BUD	054	232	LDR	086	134	REK	118	231	INA
023	008	008	055	085	085	087	215	PAO	119	231	INA
024	000	000	056	210	INB	088	221	MPR	120	231	INA
025	208	BUF	057	222	MRP	089	207	CLR	121	231	INA
026	232	LDR	058	224	JMP	090	208	BUF	122	231	INA
027	085	085	059	004	004	091	221	MPR	123	230	EXI
028	222	MRP	060	246	246	092	207	CLR	124	209	BUD
029	210	INB	061	254	TIM	093	233	LDV	125	009	009
030	134	REK	062	235	INZ	094	255	255	126	000	000
031	224	JMP	063	235	MZR	095	226	REV	127	208	BUF

Nr komórki	Kodowanie		Nr komórki	Kodowanie		Nr komórki	Kodowanie		Nr komórki	Kodowanie	
	dziesiętnie	mnemonika		dziesiętnie	mnemonika		dziesiętnie	mnemonika		dziesiętnie	mnemonika
128	232	LDR	160	221	MPR	192	233	LDV	224	226	REV
129	207	207	161	233	LDV	193	204	204	225	003	003
130	226	REV	162	255	255	194	226	REV	226	250	250
131	003	003	163	226	REV	195	003	003	227	233	LDV
132	144	144	164	003	003	196	250	250	228	231	231
133	221	MPR	165	121	121	197	233	LDV	229	226	REV
134	233	LDV	166	223	PRT	198	203	203	230	003	003
135	255	255	167	210	INB	199	226	REV	231	246	246
136	226	REV	168	224	JMP	200	003	003	232	212	PAD
137	003	003	169	003	003	201	250	250	233	008	008
138	144	144	170	159	159	202	233	LDV	234	253	253
139	235	LNZ	171	209	BUD	203	202	202	235	215	PAO
140	210	INB	172	009	009	204	226	REV	236	222	MRP
141	224	JMP	173	000	000	205	003	003	237	232	LDR
142	003	003	174	208	BUF	206	250	250	238	255	255
143	133	133	175	176	RKL	207	233	LDV	239	208	BUF
144	236	MZR	176	176	RKL	208	201	201	240	222	MRP
145	233	LDV	177	233	LDV	209	226	REV	241	215	PAO
146	000	000	178	207	207	210	003	003	242	221	MPR
147	226	REV	179	226	REV	211	250	250	243	231	INA
148	003	003	180	003	003	212	233	LDV	244	231	INA
149	156	156	181	250	250	213	200	200	245	230	EXI
150	232	LDR	182	233	LDV	214	226	REV	246	232	LDV
151	246	246	183	206	206	215	003	003	247	207	207
152	223	PRT	184	226	REV	216	250	250	248	222	MRP
153	000	000	185	003	003	217	233	LDV	249	210	INB
154	236	MZR	186	250	250	218	199	199	250	222	MRP
155	223	PRT	187	233	LDV	219	226	REV	251	210	INB
156	209	BUD	188	205	205	220	003	003	252	224	JMP
157	009	009	189	226	REV	221	250	250	253	003	003
158	000	000	190	003	003	222	233	LDV	254	174	174
159	208	BUF	191	250	250	223	198	198	255	212	PAD

Nr. komórki	Kodowanie		Nr. komórki	Kodowanie		Nr. komórki	Kodowanie		Nr. komórki	Kodowanie	
	dziesiętnie	mnemonika		dziesiętnie	mnemonika		dziesiętnie	mnemonika		dziesiętnie	mnemonika
000	009	009	032	235	INZ	064	224	JMP	096	226	REV
001	000	000	033	213	INP	065	004	004	097	003	003
002	215	PAO	034	224	JMP	066	071	071	098	121	121
003	233	LDV	035	004	004	067	213	INP	099	235	INZ
004	255	255	036	021	021	068	224	JMP	100	213	INP
005	236	MZR	037	170	LON12	069	004	004	101	210	INB
006	226	REV	038	224	JMP	070	057	057	102	224	JMP
007	003	003	039	004	004	071	208	BUF	103	004	004
008	118	118	040	042	042	072	221	MPR	104	087	087
009	221	MPR	041	168	LON11	073	233	LDV	105	208	BUF
010	226	REV	042	176	RKL	074	207	207	106	221	MPR
011	003	003	043	176	RKL	075	226	REV	107	226	REV
012	121	121	044	233	LDV	076	004	044	108	004	004
013	235	INZ	045	230	230	077	081	081	109	113	113
014	213	INP	046	226	REV	078	224	JMP	110	224	JMP
015	224	IMP	047	004	004	079	004	004	111	003	003
016	004	004	048	052	052	080	085	085	112	118	118
017	003	003	049	224	JMP	081	210	INB	113	236	MZR
018	212	PAD	050	004	004	082	224	IMP	114	247	PRA
019	009	009	051	042	042	083	004	004	115	234	LDZ
020	000	000	052	169	LOF11	084	071	071	116	000	000
021	215	PAO	053	171	LOF12	085	234	LDZ	117	236	MZR
022	233	LDV	054	231	INA	086	000	000	118	255	PRB
023	255	255	055	231	INA	087	215	PAO	119	225	AEB
024	236	MZR	056	230	EXI	088	221	MPR	120	004	004
025	226	REV	057	215	PAO	089	233	LDV	121	128	128
026	003	003	058	221	MPR	090	255	255	122	211	DEB
027	121	121	059	233	LDV	091	226	REV	123	214	DEP
028	221	MPR	060	207	207	092	004	004	124	235	INZ
029	226	REV	061	226	REV	093	105	105	125	224	JMP
030	003	003	062	004	004	094	208	BUF	126	004	004
031	118	118	063	067	067	095	221	MPR	127	117	117

Nr komórki	Kodowanie		Nr komórki	Kodowanie		Nr komórki	Kodowanie		Nr komórki	Kodowanie	
	dziesiętnie	mnemonika		dziesiętnie	mnemonika		dziesiętnie	mnemonika		dziesiętnie	mnemonika
128	215	PAO	160	000	000	192	222	MRP	224		
129	221	MPR	161	176	RKL	193	210	INB	225		
130	247	PRA	162	176	RKL	194	224	JMP	226		
131	208	BUF	163	233	LDV	195	004	004	227		
132	221	MPR	164	254	254	196	161	161	228		
133	255	PRB	165	226	REV	197			229		
134	233	LDV	166	005	005	198			230		
135	255	255	167	242	242	199			231		
136	226	REV	168	233	LDV	200			232		
137	003	003	169	230	230	201			233		
138	121	121	170	226	REV	202			234		
139	227	ALB	171	005	005	203			235		
140	003	003	172	242	242	204			236		
141	118	118	173	233	LDV	205			237		
142	228	AGB	174	229	229	206			238		
143	003	003	175	226	REV	207			239		
144	121	121	176	005	005	208			240		
145	210	INB	177	242	242	209			241		
146	213	INP	178	233	LDV	210			242		
147	224	IMP	179	231	231	211			243		
148	004	004	180	226	REV	212			244		
149	128	128	181	004	004	213			245		
150	209	BUD	182	186	186	214			246	210	INB
151	009	009	183	224	JMP	215			247	232	IDR
152	000	000	184	004	004	216			248	255	255
153	208	BUF	185	191	191	217			249	222	MRP
154	232	IDR	186	232	IDR	218			250	231	INA
155	246	246	187	207	207	219			251	231	INA
156	223	PRT	188	223	PRT	220			252	230	EXI
157	000	000	189	222	MRP	221			253		
158	236	MZR	190	210	INB	222			254		
159	223	PRT	191	223	PRT	223			255		

Nr komórki	Kodowanie		Nr komórki	Kodowanie		Nr komórki	Kodowanie		Nr komórki	Kodowanie	
	dziesiętnie	mnemonika		dziesiętnie	mnemonika		dziesiętnie	mnemonika		dziesiętnie	mnemonika
000	180	LON1	032	128	128	064	001	001	096	001	001
001	232	LDR	033	232	LDR	065	128	128	097	226	REV
002	245	245	034	248	248	066	232	LDR	098	005	005
003	223	PRT	035	222	MRP	067	248	248	099	108	108
004	176	RKL	036	232	LDR	068	222	MRP	100	233	LDV
005	176	RKL	037	175	175	069	128	LRC	101	000	000
006	223	PRT	038	229	CAL	070	233	LDV	102	226	REV
007	209	BUD	039	005	005	071	085	085	103	005	005
008	008	008	040	214	214	072	226	REV	104	157	157
009	000	000	041	224	JMP	073	005	005	195	224	JMP
010	208	BUF	042	005	005	074	083	083	106	005	005
011	222	MRP	043	056	056	075	233	LDV	107	063	063
012	176	RKL	044	232	LDR	076	171	171	108	229	CAL
013	176	RKL	045	180	K	077	226	REV	109	005	005
014	210	INB	046	223	PRT	078	005	005	110	214	214
015	222	MRP	047	232	LDR	079	083	083	111	224	JMP
016	223	PRT	048	192	?	080	229	CAL	112	005	005
017	181	LOF1	049	223	PRT	081	001	001	113	209	209
018	232	LDR	050	194	TMB	082	239	239	114	212	PAD
019	194	=	051	202	DRN	083	232	LDR	115	009	009
020	223	PRT	052	193	STR	084	180	180	116	000	000
021	232	LDR	053	224	JMP	085	229	CAL	117	221	MPR
022	248	248	054	000	000	086	005	005	118	134	REK
023	209	BUD	055	000	000	087	214	214	119	207	CLR
024	010	010	056	232	LDR	088	224	JMP	120	209	BUD
025	001	001	057	001	001	089	005	005	121	008	008
026	222	MRP	058	212	PAD	090	186	186	122	000	000
027	229	CAL	059	008	008	091	212	PAD	123	208	BUF
028	001	001	060	255	255	092	008	008	124	221	MPR
029	000	000	061	215	PAO	093	255	255	125	207	CLR
030	229	CAL	062	222	MRP	094	221	MPR	126	128	LRC
031	001	001	063	229	CAL	095	233	LDV	127	233	LDV

Nr komórki	Kodowanie dziesiętnie	Kodowanie mnemonika	Nr komórki	Kodowanie dziesiętnie	Kodowanie mnemonika	Nr komórki	Kodowanie dziesiętnie	Kodowanie mnemonika	Nr komórki	Kodowanie dziesiętnie	Kodowanie mnemonika
128	000	000	160	233	LDV	192	005	005	224	128	LRC
129	226	REV	161	248	248	193	197	197	225	226	REV
130	005	005	162	221	MPR	194	224	JMP	226	005	005
131	137	137	163	226	REV	195	005	005	227	231	231
132	232	LDR	164	005	005	196	056	056	228	224	JMP
133	002	002	165	170	170	197	232	LDR	229	003	003
134	224	JMP	166	213	INP	198	176	0	230	118	118
135	005	005	167	224	JMP	199	223	PRT	231	233	LDV
136	058	058	168	005	005	200	232	LDR	232	248	248
137	194	REK	169	162	162	201	180	K	233	213	INP
138	210	INB	170	209	BUD	202	223	PRT	234	221	MPR
139	221	MPR	171	009	009	203	194	TMB	235	226	REV
140	207	CLR	172	000	000	204	202	DRN	236	003	003
141	213	INP	173	214	DEP	205	193	STR	237	121	121
142	215	PAO	174	208	BUF	206	224	JMP	238	224	JMP
143	221	MPR	175	221	MPR	207	000	000	239	003	003
144	207	CLR	176	215	PAO	208	031	031	240	118	118
145	128	LRC	177	222	MRP	209	232	LDR	241	000	000
146	226	REV	178	226	REV	210	175	175	242	212	PAD
147	005	005	179	005	005	211	224	JMP	243	008	008
148	152	152	180	063	063	212	005	005	244	253	253
149	224	JMP	181	213	INP	213	046	046	245	215	PAO
150	005	005	182	210	INB	214	134	REK	246	222	MRP
151	132	132	183	224	JMP	215	207	CLR	247	232	LDR
152	232	LDR	184	005	005	216	212	PAD	248	255	255
153	000	000	185	174	174	217	009	009	249	208	BUF
154	214	DEP	186	214	DEP	218	000	000	250	222	MRP
155	214	DEP	187	214	DEP	219	215	PAO	251	215	PAO
156	222	MRP	188	221	MPR	220	221	MPR	252	221	MPR
157	212	PAD	189	233	LDV	221	207	CLR	253	231	INA
158	010	010	190	000	000	222	233	LDV	254	231	INA
159	001	001	191	226	REV	223	000	000	255	230	EXI

Nr komórki	Kodowanie		Nr komórki	Kodowanie		Nr komórki	Kodowanie		Nr komórki	Kodowanie	
	dziesiętnie	mnemonika		dziesiętnie	mnemonika		dziesiętnie	mnemonika		dziesiętnie	mnemonika
000	098	DLW	032	150	150	064	006	006	096	009	009
001	229	CAL	033	229	CAL	065	072	072	097	130	130
002	003	003	034	003	003	066	123	CDAN	098	224	JMP
003	061	061	035	061	061	067	222	MRP	099	006	006
004	229	CAL	036	231	INA	068	210	INB	100	241	241
005	003	003	037	231	INA	069	224	JMP	101	234	LDZ
006	061	061	038	230	EXI	070	006	006	102	000	000
007	229	CAL	039	233	LDV	071	062	062	103	122	CNSK
008	003	003	040	255	255	072	104	STFD2	104	134	REK
009	161	161	041	104	STFD2	073	226	REV	105	207	CLR
010	108	STFD4	042	226	REV	074	006	006	106	236	MZR
011	226	REV	043	006	006	075	062	062	107	207	CLR
012	006	006	044	041	041	076	232	LDR	108	233	LDV
013	085	085	045	231	INA	077	255	255	109	000	000
014	232	LDR	046	231	INA	078	222	MRP	110	128	IRC
015	001	001	047	230	EXI	079	120	CSTAT	111	226	REV
016	118	WNSK	048	209	BUD	080	224	JMP	112	006	006
017	232	LDR	049	009	009	081	006	006	113	118	118
018	000	000	050	000	000	082	230	230	114	235	INZ
019	229	CAL	051	232	LDR	083	000	000	115	224	JMP
020	006	006	052	016	016	084	000	000	016	006	006
021	027	027	053	229	CAL	085	106	ZERFD	117	103	103
022	232	LDR	054	006	006	086	097	KRUN	118	235	INZ
023	000	000	055	027	027	087	224	JMP	119	236	MZR
024	117	WNSC	056	208	BUF	088	003	003	120	233	LDV
025	232	LDR	057	232	LDR	089	118	118	121	027	027
026	064	064	058	140	140	090	096	PRUN	122	226	REV
027	116	WROZ	059	116	WROZ	091	000	000	123	006	006
028	229	CAL	060	233	LDV	092	232	LDR	124	131	131
029	006	006	061	255	255	093	245	245	125	118	WNSK
030	039	039	062	105	STFD3	094	223	PRT	126	121	CNSC
031	234	LDZ	063	226	REV	095	209	BUD	127	119	WDAN

Nr komórki	Kodowanie dziesiętnie	Kodowanie mnemonika	Nr komórki	Kodowanie dziesiętnie	Kodowanie mnemonika	Nr komórki	Kodowanie dziesiętnie	Kodowanie mnemonika	Nr komórki	Kodowanie dziesiętnie	Kodowanie mnemonika
128	224	JMP	160	187	187	192	000	000	224	233	LDV
129	003	003	161	226	REV	193	209	BUD	225	129	129
130	118	118	162	009	009	194	008	008	226	236	MZR
131	232	LDR	163	002	002	195	001	001	227	226	REV
132	001	001	164	097	KRUN	196	208	BUF	228	006	006
133	118	WNSK	165	106	ZERFD	197	232	LDR	229	079	079
134	121	CNSC	166	232	LDR	198	016	016	230	109	KZZ
135	233	LDV	167	245	245	199	224	JMP	231	233	LDV
136	076	076	168	223	PRT	200	006	006	232	000	000
137	226	REV	169	232	LDR	201	184	184	233	226	REV
138	006	006	170	185	F	202	233	LDV	234	003	003
139	036	036	171	223	PRT	203	000	000	235	118	118
140	232	LDR	172	232	LDR	204	116	WROZ	236	121	CNSC
141	080	080	173	187	D	205	120	CSTAT	237	119	WDAN
142	116	WROZ	174	223	PRT	206	226	REV	238	231	INA
143	229	CAL	175	232	LDR	207	006	006	239	231	INA
144	006	006	176	192	?	208	205	205	240	230	EXI
145	039	039	177	223	PRT	209	211	DEB	241	234	LDZ
146	224	JMP	178	224	JMP	210	233	LDV	242	255	255
147	006	006	179	000	000	211	001	001	243	229	CAL
148	126	126	180	000	000	212	120	CSTAT	244	004	004
149	229	CAL	181	000	000	213	226	REV	245	153	153
150	006	006	182	000	000	214	006	006	246	229	CAL
151	048	048	183	000	000	215	212	212	247	006	006
152	224	JMP	184	229	CAL	216	221	MPR	248	007	007
153	006	006	185	006	006	217	119	WDAN	249	232	LDR
154	164	164	186	027	027	218	210	INB	250	000	000
155	209	BUD	187	102	PZZ	219	235	INZ	251	119	WDAN
156	009	009	188	232	LDR	220	120	CSTAT	252	224	JMP
157	000	000	189	172	172	221	226	REV	253	006	006
158	221	MPR	190	224	JMP	222	006	006	254	149	149
159	233	LDV	191	006	006	223	212	212	253	--	--