

**ZAM-41**

---

---

**Kompendium  
oprogramowania**

---



*kat 4*

INSTYTUT MASZYN MATEMATYCZNYCH

Jan Wierzbowski


KOMPENDIUM OPROGRAMOWANIA



Warszawa 1973

7  
2  
5  
3  
4  
5  
1  
7  
8  
9  
4  
5  
6  
7  
3  
3

25, 370  
1202  
S. 19  
Projekt okładki: Bożena Bratkowska

  
Komitety Redakcyjne: J. Borowiec (red. nac.),  
W. Kossakowski, A. Mazurkiewicz,  
J. Wierzbowski, A. Wiśniewski,  
H. Drozdowska (sekr. red.)

Adres redakcji:

02-078 Warszawa, ul. Krzywickiego 34  
tel. 28-37-29



23310  
Materiały opracowane i przepisane na kredzie  
przez Zleceniodawcę

1984-D-38/419

## SPIS TREŚCI

str.

WSTĘP . . . . .	6
1. OGÓLNA ORGANIZACJA MASZYNY . . . . .	9
1.1. JEDNOSTKA CENTRALNA . . . . .	13
1.2. KANAŁY . . . . .	15
1.3. URZĄDZENIA WEJŚCIA I WYJŚCIA . . . . .	15
1.4. PAMIĘCI ZEWNĘTRZNE . . . . .	29
1.5. SYSTEM PRZERWAŃ . . . . .	31
1.6. ADRESOWANIE SŁÓW W PAMIĘCI . . . . .	34
2. OGÓLNE INFORMACJE O PROGRAMOWANIU . . . . .	37
2.1. SYSTEM OPERACYJNY I BIBLIOTEKA . . . . .	38
2.2. TRANSLATORY . . . . .	42
2.3. PROGRAMY UŻYTKOWE I TESTY. . . . .	44
2.4. JEDNOSTKA PRZETWARZANIA . . . . .	54
3. PROGRAMOWANIE . . . . .	57
3.1. POSTAĆ INFORMACJI W MASZYNIE . . . . .	57
3.2. REJESTRY, WSKAŹNIKI, ULR . . . . .	62
3.3. SPOSÓB REALIZACJI ROZKAZU. . . . .	65
3.4. LISTA ROZKAZÓW . . . . .	68
3.5. DZIAŁANIA Z WYNIKIEM O . . . . .	114
3.6. STANDARD ZAPISU INFORMACJI NA TAŚMIE MAGNETYCZNEJ . . . . .	115
3.7. WSPÓŁPRACA Z URZĄDZENIAMI ZEWNĘTRZNYMI	121
4. WŁĄCZANIE ZBIORU DO BIBLIOTEKI SYSTEMU . . . . .	127
4.1. TYP I ORGANIZACJA ZBIORU . . . . .	128
4.2. PRZYGOTOWANIE PLIKU ŹRÓDŁOWEGO ZBIORU	129
4.3. PRODUKOWANIE TAŚMY BINARNEJ . . . . .	134
4.4. WPROWADZENIE ZBIORU DO BIBLIOTEKI SYSTEMU. . . . .	135
4.5. WYWOŁANIE ZBIORU Z BIBLIOTEKI SYSTEMU	136
4.6. WPROWADZENIE NOWYCH FUNKCJI JĘZYKA SAS	137
5. OPEROWANIE MASZYNĄ . . . . .	143
5.1. STARTOWANIE SYSTEMU OPERACYJNEGO . . . . .	143

5.2.	UPROSZCZONE METODY STARTOWANIA SO . . .	151
5.3.	OBŚLUGA STOLIKA OPERATORA I URZĄDZEŃ POMOCNICZYCH . . . . .	153
5.4.	DODATKOWE UWAGI O OBSŁUDZE NIEKTÓRYCH URZĄDZEŃ . . . . .	161
5.5.	TEKSTY PRZY WSPÓLPRACY Z MONITOREM DALEKOPISOWYM . . . . .	166
6.	URZĄDZENIA ZEWNĘTRZNE I KODY . . . . .	169
6.1.	WYKAZ URZĄDZEŃ ZEWNĘTRZNYCH I ICH NUMERÓW . . . . .	169
6.2.	KOD KW-6 . . . . .	172
6.3.	KOD KW-8 . . . . .	173
6.4.	WARTOŚCI WYBRANYCH ZNAKÓW W KW-6 I KW-8 . . . . .	174
6.5.	SPIS PODPROGRAMÓW WEJŚCIA I WYJŚCIA .	176
6.6.	OPIS PODPROGRAMÓW WEJŚCIA I WYJŚCIA .	177
6.7.	SPIS DEKODERÓW . . . . .	188
6.8.	WYBRANE DEKODERY TYPU KOD ZEWNĘTRZNY - KW6 . . . . .	190
6.9.	WYBRANE DEKODERY TYPU KW6 - KOD ZEWNĘTRZNY . . . . .	194
6.10.	WYBRANE DEKODERY TYPU KOD ZEWNĘTRZNY - KW8 . . . . .	196
6.11.	WYBRANE DEKODERY TYPU KW8 - KOD ZEWNĘTRZNY . . . . .	201
6.12.	KOD M-2 . . . . .	206
6.13.	KOD FERRANTI . . . . .	207
6.14.	KOD OPTIMA 528 . . . . .	208
7.	T A B L I C E . . . . .	211
7.1.	WARTOŚCI DZIESIĘTNE $2^N$ , $2^{-N}$ ORAZ $2^x$ .	211
7.2.	WARTOŚCI DZIESIĘTNE $8^N$ i $8^{-N}$ . . . . .	212
7.3.	WARTOŚCI DZIESIĘTNE $16^N$ i $16^{-N}$ . . . . .	213
7.4.	WARTOŚCI OKTALNE $10^N$ i $10^{-N}$ . . . . .	214
7.5.	NIEKTÓRE STAŁE MATEMATYCZNE . . . . .	215
7.6.	TABLICZKA DODAWANIA I MNOŻENIA LICZB OKTALNYCH . . . . .	216

7.7. TABLICA ZMIANY LICZB OKTALNYCH NA DZIESIĘTNE I ODWROTNIE . . . . .	217
7.8. TABLICA ZMIANY LICZB SZESNASTKOWYCH NA DZIESIĘTNE I ODWROTNIE . . . . .	219
7.9. TABLICA ZMIANY LICZB OKTALNYCH NA SZESNASTKOWE . . . . .	222
7.10. WARTOŚCI DZIESIĘTNE, OKTALNE I NAZWY CZĘŚCI OPERACYJNYCH ROZKAZÓW . . . . .	224
7.11. SPIS PRZYCZYŃ ZAKOŃCZENIA PROGRAMÓW UŻYTKOWNIKA . . . . .	226
7.12. ZAWARTOŚĆ WYBRANYCH SŁÓW W PAO . . . . .	229
7.13. TABLICE PORÓWNAWCZE POJEMNOŚCI I SZYBKOŚCI NOŚNIKÓW I URZĄDZEŃ ZEWNĘTRZNYCH	232
7.14. ZALECANE SYMBOLE GRAFICZNE DLA SCHEMATÓW BLOKOWYCH I SIECI DZIAŁAŃ . . . . .	240
7.15. ŚCIAĞACZKI SO . . . . .	245
7.16. NAZWY UŻYWANE W ZDANIACH SUP, ZMI, UZU 1 DEK . . . . .	246
7.17. WYKAZ NIEKTÓRYCH OŚRODKÓW WYPOSAŻONYCH W MASZYŃĘ ZAM 41 . . . . .	247

## WSTĘP

Celem niniejszej książki jest zebranie w zwartej postaci podstawowych informacji dotyczących maszyny ZAM 41, które mogą być przydatne dla jej użytkowników; przy czym przez użytkowników rozumie się w tym przypadku programistów i operatorów, a częściowo także projektantów systemów. Ze względu na szeroki zakres poruszanych zagadnień książka ma niejednorodny charakter. Większa część książki jest rodzajem vademecum, tzn. zawiera informacje bez dodatkowych komentarzy. Niektóre partie mają charakter bardziej opisowy, co powinno ułatwić czytelnikowi zrozumienie podanych wiadomości.

Książka nie wyczerpuje całości rozważanych zagadnień - podstawowym założeniem było danie czytelnikowi informacji prawdziwych, ale nie zawsze omówienie całości zagadnienia. Tam, gdzie wymagała tego zwięzłość i jasność tekstu, wprowadzono pewne ograniczenia, nie pokazując wszystkich możliwości rozwiązania danego zagadnienia.

Książką tą należy posługiwać się razem z innymi książkami dotyczącymi poszczególnych języków programowania. Jako uzupełnienie niniejszego kompendium, przewiduje się wydanie w ramach tej serii książki na temat systemu operacyjnego, JOM, biblioteki systemu, metod generacji systemu itp. a przeznaczonej dla bardziej zaawansowanych programistów maszyny ZAM 41.

Pisząc tę książkę autor miał trudności z wyborem odpowiedniej terminologii. Przyjęty w środowisku programistów żargon zmuszał do akceptowania niektórych neologizmów. Przykładem może



być "wciskanie klucza" trwałe, "naciskanie klucza" chwilowe oraz "wyciskanie klucza" (czynność odwrotna do wciskania). Autorowi zależało na wywołaniu u Czytelników pewnych skojarzeń i dlatego ma nadzieję, że wybaczą mu oni nie zawsze właściwie dobraną terminologię.

Autor dziękuje wszystkim osobom, które pomagały mu w trakcie pisania tej książki, a zwłaszcza kolegom: Janowi Borowcowi, który poddał myśl napisania kompendium, Jerzemu Swianiewiczowi, który wielokrotnie konsultował autora oraz Krzysztofowi Bytnerowiczowi, Zbigniewowi Kosowskiemu i Marianowi Skupińskiemu, którzy w całości opracowali punkty 5.14 i 5.16, a ponadto dostarczyli materiałów, informacji oraz własnych uwag, bez których niemożliwe byłoby osiągnięcie tego zakresu tematyki, który książka obejmuje.



## 1. OGÓLNA ORGANIZACJA MASZYNY

Maszyna ZAM 41 jest maszyną tranzystorową drugiej generacji, zbudowaną na obwodach drukowanych. Pamięć operacyjna jest zbudowana z rdzeni ferrytowych; średnia szybkość maszyny przy obliczeniach stałoprzecinkowych wynosi 40 000 operacji na sekundę. Podstawowe cechy maszyny:

- modułowa konstrukcja oraz standardowy interfejs (sposób współpracy różnych modułów), umożliwiający zestawianie różnych konfiguracji maszyny w zależności od rodzaju zastosowań i potrzeb użytkownika;
- arytmetyka binarna (tzn. oparta na dwójkowym systemie liczenia) stało- i zmiennieprzecinkowa;
- słowo 24-bitowe, będące podstawową jednostką informacji; jednostką przesyłania infor-

macji jest na ogół słowo 24-bitowe, ciąg 8 bitów lub 1 bit;

- rozkaz zawarty zawsze w jednym słowie maszyny;
- rozkazy jednoadresowe, trzy rodzaje modyfikacji adresu;
- wbudowany system przerwań, umożliwiający pracę wieloprogramową;
- wbudowany zegar;
- rejestry dolnej i górnej blokady, zapewniające ochronę danych w ramach poszczególnych programów;
- dwie grupy rozkazów (legalne i nielegalne) oraz dwa reżimy pracy dla rozkazów nielegalnych, umożliwiające stworzenie systemu wieloprogramowego;
- układy kontrolne, wbudowane do niektórych modułów, zapewniające niezawodną pracę.

Maszyna ZAM 41 składa się:

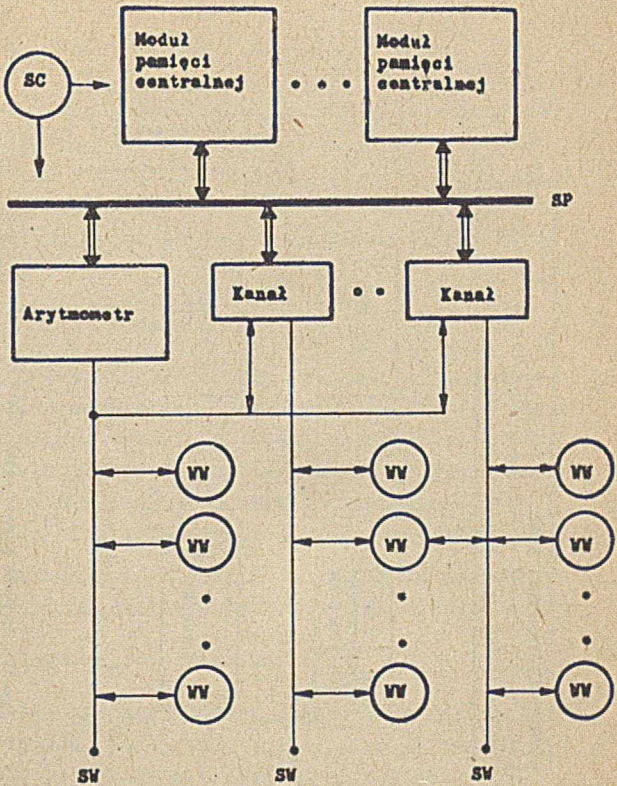
- z jednostki centralnej,
- kanałów,
- urządzeń wejścia i wyjścia,
- pamięci zewnętrznych.

Ponadto można wyróżnić tzw. szynę pamięci i szynę wejścia-wyjścia, stanowiące standardowe łącza między poszczególnymi modułami.

Schemat blokowy maszyny pokazany jest na rys. 1a.

Fizycznie maszyna ZAM 41, w zależności od konfiguracji, składa się z kilku szaf o standardowych wymiarach ok. 60 cm x 67 cm x 170 cm oraz wolnostojących modułów urządzeń zewnętrznych. Wśród modułów tych szczególną rolę odgrywa stół operatora, przy którym operator wykonuje większość swoich funkcji.

Przykład rozmieszczenia modułów maszyny podany jest na rys. 1b.



SC - sterowanie centralne  
 SP - szyna pamięci  
 WV - moduły wejściowo-wyjściowe  
 SW - szyny wejścia-wyjścia

Rys. 1a. Schemat blokowy maszyny ZAM 41

(wg L. Łukaszczyka: Rodzina maszyn matematycznych ZAM - Organizacja ogólna)

ARITMOMETRY STEROWANIE	PAO 4k	KAWAL	PAO 4k	PB 32k
PAO 4k	PAO 4k	PAO 4k	PAO 4k	PB 32k

1	2	3	4	5	6
	TASMA		MAGNETYCZNA		

MONITOR

CZYTNIA  
KART

DRUKARKA  
WIRSZOVA

DRUGI  
CZYTNIAK  
TASMY

DRUGI  
PERFORA-  
TOR  
TASMY

STOLIK  
OPERATORA

Fig. 13. Przykład rozmieszczenia modułów maszyny

## 1.1. JEDNOSTKA CENTRALNA

Jednostka centralna maszyny ZAM 41 składa się z arytmometru, pamięci operacyjnej oraz jednostki sterowania centralnego. Ponadto w skład jednostki centralnej wchodzi zegar, który ze względu na swoją rolę zostanie opisany oddzielnie.

### 1.1.1. Arytmometr

Arytmometr jest zespołem rejestrów, służących przede wszystkim do wykonywania obliczeń arytmetycznych stało- i zmiennoprzecinkowych oraz operacji logicznych. Arytmometr służy również do przekazywania informacji między pamięcią operacyjną i urządzeniami wejścia i wyjścia.

Podstawowymi rejestrami arytmometru są: akumulator, mnożnik i rejestr modyfikacji (B-rejestr).

### 1.1.2. Pamięć operacyjna (PAO)

Pamięć operacyjna jest zbudowana z rdzeni ferrytowych. Każdy moduł pamięci ma pojemność 4k (4096) słów 24 bitowych. Cykl pamięci wynosi 10  $\mu$ s.

Każde słowo pamięci posiada określony adres. 15-bitowa część adresowa rozkazów maszyny pozwala bezpośrednio adresować 32k słów, tj. 8 modułów. Zastosowanie B-modyfikacji lub pośredniego adresowania, pozwala rozszerzyć adres do 18 bitów, tj. 256k słów. Jest to maksymalna (teoretyczna) pojemność pamięci operacyjnej.

Praktycznie większość maszyn jest wyposażona w pamięć operacyjną o pojemności 20k (20480) słów.

### 1.1.3. Sterowanie centralne

Sterowanie centralne służy do koordynacji wymiany informacji między poszczególnymi modułami maszyny. W szczególności zadaniem sterowania jest pobieranie kolejnych rozkazów do wykonania. Podstawowymi rejestrami sterowania są: rejestr rozkazów (jego zawartość określa aktualnie wykonywaną operację) oraz licznik rozkazów (jego zawartość określa miejsce w PAO, z którego należy pobrać następny rozkaz). Zadaniem sterowania centralnego jest również obsługa przerw programów. Zasady systemu przerw są omówione w punkcie 1.5.

### 1.1.4. Zegar

Zegar jest rejestrem 6-bitowym służącym do odmierzenia czasu rzeczywistego w maszynie. Zawartość zegara zostaje co 0.02 s (wg częstotliwości sieci) zwiększona o 1. Po osiągnięciu maksymalnej wartości (63) zawartość zegara zostaje wyzerowana, zgłaszając jednocześnie przerwanie. W ten sposób co 1.28 s następuje przerwanie programu, co umożliwia przejście wybranych wskaźników, przekazanie sterowania do programu o wyższym priorytecie itp. Jest to tylko jeden ze sposobów wykorzystania zegara. Inny sposób polega na korzystaniu z możliwości wpisania do zegara określonej zawartości, pobrania z niego aktualnej zawartości oraz zablokowania wysyłanych przez zegar przerw.



W niektórych (nietypowych) przypadkach, maszyna może pracować z odłączonym zegarem.

## 1.2. KANAŁY

Kanały służą do sterowania pracą poszczególnych urządzeń wejścia i wyjścia, a w szczególności do autonomicznego przesyłania informacji (bloków słów) do lub z pamięci ferrytowej. Przesyłanie informacji odbywa się blokami, niezależnie od pracy arytmometru.

Każdy kanał posiada zestaw rejestrów i wskaźników, w których przechowywane są informacje sterujące dla urządzeń wejścia i wyjścia. Zawartość tych rejestrów można ustawiać i pobierać przez wykonanie odpowiednich rozkazów maszyny (grupa rozkazów OWW). W trakcie pracy kanału zawartość rejestrów jest automatycznie zmieniana. W określonych sytuacjach (np. po zakończeniu przesłania) kanał powoduje zgłoszenie przerwania.

## 1.3. URZĄDZENIA WEJŚCIA I WYJŚCIA


Do maszyny ZAM 41 można dołączać różne urządzenia wejścia i wyjścia (moduły), byleby spełniały one przyjęty standard współpracy. W praktyce stosuje się następujące rodzaje urządzeń:

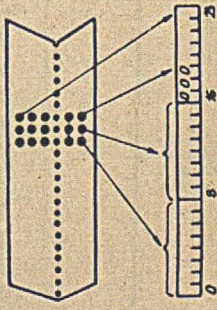
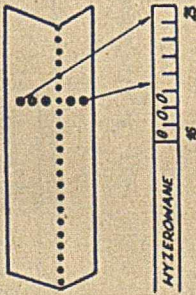
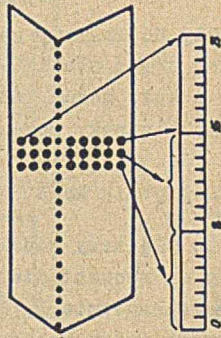
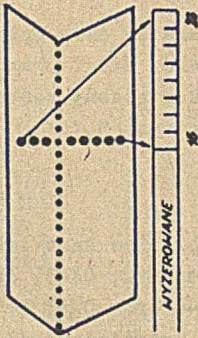
- czytnik taśmy dziurkowanej,
- czytnik kart,
- monitor dalekopisowy,
- drukarka wierszowa,
- dziurkarka taśmy papierowej.

W pewnym sensie stolik operatora odgrywa podobną rolę do urządzeń wejścia i wyjścia, dlatego będzie opisany w dalszej części pracy.

Wymienione wyżej rodzaje urządzeń można teoretycznie zastąpić innymi, jednak w praktyce to się nie zdarza.

### 1.3.1. Czytnik taśmy dziurkowanej

Najczęściej stosowanymi czytnikami taśmy dziurkowanej są: czytnik CT 1001 produkcji Zakładów Mechaniczno-Precyzyjnych "Błonie" oraz TR-5B produkcji firmy Ferranti lub TR-6 produkcji firmy ICL. Wszystkie wymienione czytniki czytają taśmę 5-8 kanałową; przełączenie z jednej szerokości na inną trwa zaledwie kilkanaście sekund. Czytane rzędkie są zawsze uzupełniane zerami do 8 bitów, przy czym mogą one być wprowadzane pojedynczo na najniższe pozycje słowa lub akumulatora (zerując pozycje starsze) albo umieszczane w słowie lub akumulatorze trójkami; graficznie przedstawia 



a)

b)

Rys.2 a) Czytanie po jednym rzędku do słowa lub akumulatora  
 b) Czytanie po 3 rzędki do słowa lub akumulatora

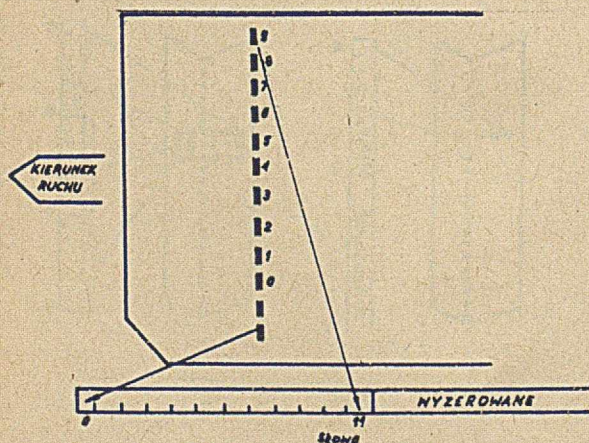
Szybkość czytnika CT 1001 wynosi ok. 1000 rzęd-  
ków/s, szybkość TR-5B i TR-6 - około 300 rząd-  
ków/s.

Z reguły do maszyny podłączane są dwa czytniki  
- jeden przez stolik operatora, drugi przez od-  
rębny moduł. Moduł ten posiada własny zestaw  
kluczy i lampek, pokazany na rys. 3.

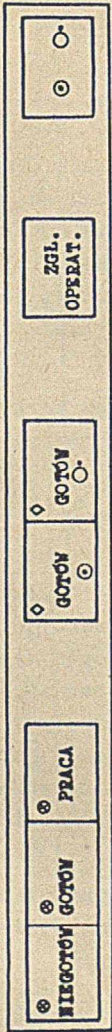
### 1.3.2. Czytnik kart

Stosuje się trzy rodzaje czytników kart: CK-2,  
oparty na mechanizmie Elliott B42, CK-3, opar-  
ty na mechanizmie Elliott B46 oraz CK-3 beta,  
oparty na innym mechanizmie, również firmy  
Elliott. Czytniki te pracują z szybkością 400  
kart/min.

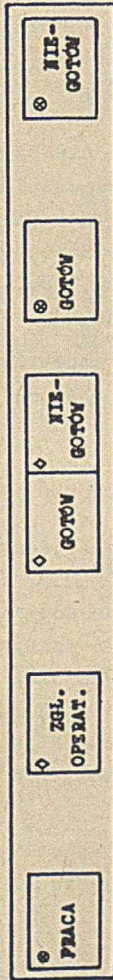
Obraz czytanych kolumn jest przesyłany do ko-  
lejnych słów pamięci, tak jak to pokazuje po-  
niższy rysunek:



Koniec karty jest sygnalizowany przesłaniem  
obrazu kolumny zawierającej 12 dziurek.



Rys. 3. Klawiatura modułu czytnika i dsierkarki taśmy



◇ klucz jednostabilny }  
 ⊗ lampka } oznaczenia dotyczą rys. 3-7

Rys. 4. Klawiatura modułu czytnika kart CK-2

Zasadniczą różnicę między czytnikami stanowi sposób rozpoznawania kolumny na karcie:

CK-2 rozpoznaje kolumnę tylko wówczas, gdy jest w niej co najmniej jedna dziurka, natomiast CK-3 i CK-3 beta mają wbudowany zegar, wyznaczający położenie kolumn na kartach. Dlatego czytnik CK-2 nie czyta kolumn pustych, ale może bez przełączania czytać karty 80- i 90-kolumnowe; czytnik CK-3 czyta tylko karty 80-kolumnowe, natomiast CK-3 beta jest wyposażony w przełącznik, umożliwiający czytanie zarówno kart 80- jak i 90-kolumnowych.

Zestaw kluczy i lampek modułu czytnika kart jest pokazany na rys. 4 (dla CK-2) i na rys. 5 (dla CK-3).

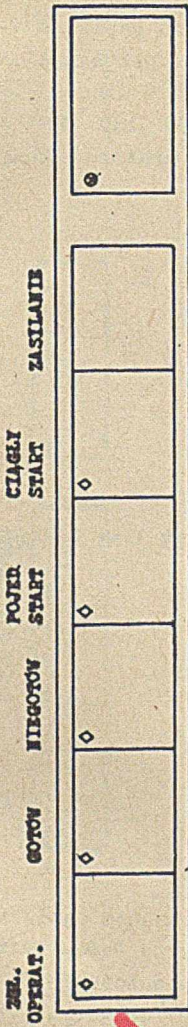
### 1.3.3. Monitor dalekopisowy

Jako monitor dalekopisowy stosuje się dalekopis firmy LORENTZ L015B lub L0133, pracujący w kodzie międzynarodowym nr 2. Dalekopis wyposażony w czytnik i dziurkarke taśmy papierowej może służyć zarówno jako wejście do maszyny, jak i jej wyjście. Ze względu na bardzo małą szybkość pracy, dalekopis jest stosowany do wprowadzania lub wyprowadzania niewielkiej ilości informacji.

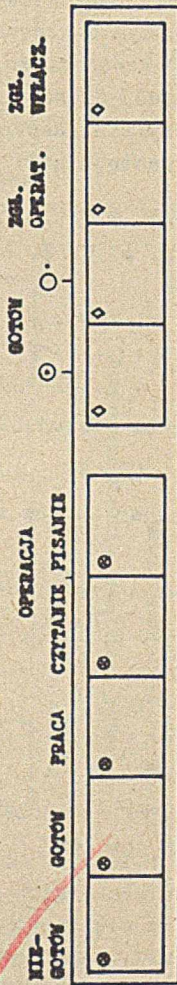
Niezależnie od klawiatury dalekopisu, moduł monitora posiada zestaw kluczy i lampek, pokazany na rys. 6.

### 1.3.4. Drukarka wierszowa

Aktualnie stosuje się niemal wyłącznie drukarki wierszowe typu DW-2 lub DW-2E, oparte na mechanizmie ICL typ 666. Uprzednio była również



Rys. 5. Klawiatura modułu czynnika kart CK-3



Rys. 6. Klawiatura modułu monitora

stosowana drukarka DW-1.

Drukarki DW-2 i DW-2E drukują na papierze 1-6 warstwowym o szerokości do 120 znaków w wierszu. Gęstość druku wynosi 6 lub 8 wierszy/cal, szybkość drukowania - ok. 600 lub 1100 wierszy/min. Repertuar znaków drukarki DW-2 jest następujący:

a) znaki graficzne (64 znaki):

A-Z, ą, ć, ę, ł, ń, ó, ś, ź, ż  
 0-9  
 +, -, /, \*, :, ;, ., ,  
 (, ), [ , ], <, =, >  
 Œ, ´, ¯, spacja

b) znaki sterujące:

NL, CR, FF

Repertuar znaków drukarki DW-2E jest następujący:

a) znaki graficzne (64 znaki):

A-Z, ą, ć, ę, ł, ó, ś, ź  
 0-9  
 +, -, /, \*, :, ;, ., ,  
 (, ), [ , ], <, =, >  
 |, #, %, ", ´, spacja

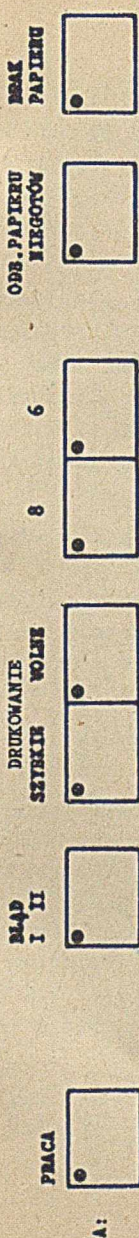
b) znaki sterujące:

NL, CR, FF

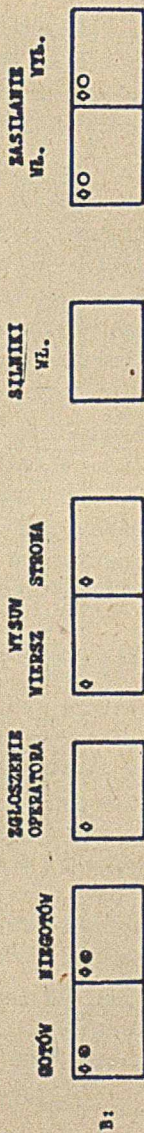
W obu drukarkach sterowanie wysuwem papieru odbywa się za pomocą 8-kanalowej taśmy perforowanej, zamkniętej w pętlę i założonej do specjalnego czytnika (wewnątrz drukarki), sprzężonego elektrycznie z mechanizmem wysuwu.

Zestaw kluczy i lampek drukarki DW-2 i DW-2E pokazano na rys. 7.





A:



B:

Rys.7. Klawiatura drukarki wierszowej

A - lewa strona

B - prawa strona

### 1.3.5. Dziurkarka taśmy papierowej

Jako dziurkarkę taśmy papierowej stosuje się przede wszystkim dziurkarkę firmy Facit typu PE 1500, czasami również dziurkarkę D 102 produkcji Zakładów Mechaniczno-Precyzyjnych "Ene".

Dziurkarka typu Facit dziurkuje taśmę 5-8 kanałową z szybkością 150 rzędków/s. Przesłanie taśmy z jednej szerokości taśmy na inną wymaga wymiany prowadnicy i elektrycznego przełączenia; czynność ta, trwająca kilkanaście minut, powinna być wykonywana przez konwojatora.

Z reguły stosuje się dwie dziurkarki taśmy: jedną dołączoną do maszyny przez stolik operatora, drugą - przez specjalny moduł. Zestaw kluczy i lampek tego modułu pokazano na rys. 1.3.5. Sposób wyprowadzania rzędków taśmy jest odwrócony do sposobu wprowadzania, opisanego w p. 1.1.4.

### 1.3.6. Stolik operatora

Stolik operatora służy do bezpośredniego nadzoru pracy maszyny. W skład stolika wchodzi między innymi zestaw kluczy i lampek pokazany na rys. 1.3.6. Ponadto przez stolik operatora podłączone są do maszyny: jeden czytnik i jedna dziurkarka taśmy papierowej. W stoliku jest również zamontowany zegar (opisany w punkcie 1.1.4) oraz głośnik, którego dźwięk charakteryzuje w pewien sposób aktualną pracę maszyny.

Poniżej opisano funkcje znajdujących się na stoliku operatora kluczy i lampek, które otwierają możliwości tego stolika.

0	B	P	32	46	8	4	2	1	16304	8107	4036	2046	4024	512	256	128	64	32	16	8	4	2	1
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
0			3		6		9		12		15		18		21		24		27		30		33
<input checked="" type="checkbox"/>																							
CZYNIK	F	LADUJ	A	M	B	L	R	ZŁOŻENIE	OPERATORA	PRL	CYKLSTOPSTART	VPROWADZ.PRACA											

lampka  klucz dwustabilny  klucz jednostabilny

rys. 8. Stolik operatora

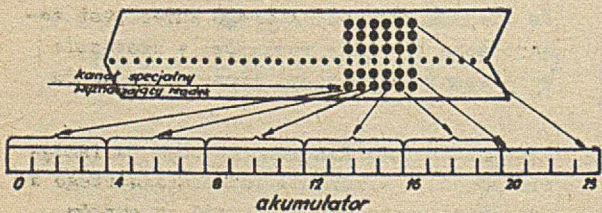


- a) Górny rząd składa się z 24 lampek, wyświetlających informacje dla operatora. W czasie pracy maszyny wyświetlane są informacje załadowane do odpowiedniego rejestru lampek przez program; gdy maszyna nie pracuje - lampki wyświetlają zawartość rejestru, wybranego jednym z kluczy oznaczonych literami A, M, B, L, R (por. punkt f). Napisy przy lampkach ułatwiają interpretację wyświetlanej informacji.
- b) Środkowy rząd obejmuje 24 klucze, umożliwiające ustawienie informacji przeznaczonych do wprowadzenia do maszyny (wciśnięty klucz oznacza 1, wyciśnięty - 0). Podczas pracy maszyny informacja ustawiana na kluczach może być wprowadzona do maszyny tylko przez wykonanie odpowiedniego rozkazu (CKA). Kiedy maszyna nie pracuje, wówczas informacja ustawiona na kluczach może być wprowadzona do wybranego rejestru A, M, B, L lub R (por. punkt f) przez naciśnięcie klucza ŁADUJ.
- c) CZYTNIK składa się z dwóch kluczy: wciśnięcie lewego powoduje włączenie zasilania dla podłączonego przez stół operatora czytnika taśmy papierowej; naciśnięcie prawego powoduje wyłączenie zasilania.
- d) F jest kluczem określającym reżim pracy maszyny: wciśnięty klucz zapala wskaźnik legalności i powoduje inny sposób wykonywania pewnych rozkazów. W niektórych egzemplarzach maszyny klucz ten został odłączony od maszyny. W pozostałych egzemplarzach (w czasie działania pod nadzorem Systemu Operacyjnego) klucz ten nie powinien być wciśnięty.

- e) ŁADUJ jest kluczem powodującym:
- o wprowadzenie informacji ustawionej na kluczach do wybranego rejestru A, M, B, L lub R jeśli jeden z nich był wybrany;
  - o wyzerowanie określonych rejestrów maszyny - jeśli żaden z kluczy A, M, B, L i R nie został wciśnięty.
- f) A, M, B, L i R są kluczami wybierającymi odpowiednio następujące rejestry: akumulator, mnożnik, B-rejestr, licznik rozkazów, rejestr rozkazów; celem wybrania jest wyświetlenie zawartości rejestru na lampkach lub wprowadzenie do niego informacji z kluczy (por. punkty a, b i e). Wciskanie kilku kluczy jednocześnie jest w zasadzie niedozwolone.
- g) ZGŁOSZENIE OPERATORA jest kluczem, którego naciśnięcie powoduje przerwanie pracy programu (jeśli maszyna pracuje i odpowiednie wskaźniki pozwalają na przyjęcie tego przerwania); umożliwia to zbadanie żądań operatora przez odpowiedni program obsługi tego przerwania.
- h) PRL jest kluczem, którego naciśnięcie powoduje przesłanie informacji z akumulatora do miejsca pamięci, którego adres jest zawartością licznika rozkazów, a następnie zwiększenie zawartości licznika rozkazów o 1.
- i) WYK.ROZKAZ jest kluczem, którego naciśnięcie powoduje wykonanie rozkazu zawartego w rejestrze rozkazów. Zawartość rejestrów nie związanych z wykonywanym rozkazem nie ulega zmianie; w szczególności nie ulega zmianie zawartość rejestru rozkazów, a jes-

li rozkaz nie był rozkazem skokowym, to nie zmienia się również zawartość licznika rozkazów.

- j) **CYKL** jest kluczem, którego naciśnięcie powoduje wykonanie jednego cyklu pracy maszyny, tj.:
- pobranie do rejestru rozkazów następnego rozkazu z miejsca PAO wskazanego przez zawartość licznika rozkazów,
  - zmianę zawartości licznika rozkazów,
  - wykonanie rozkazu zawartego w rejestrze rozkazów.
- k) **STOP** jest kluczem, którego naciśnięcie powoduje zatrzymanie pracy maszyny (jeśli maszyna pracowała).
- l) **START** jest kluczem, którego naciśnięcie powoduje uruchomienie pracy maszyny (jeśli maszyna była zatrzymana).
- m) **WPROWADZ.** jest kluczem, którego naciśnięcie powoduje:
- pobranie z czytnika taśmy 6 rzędów taśmy 5-kanalowej oraz złożenie ich w akumulatorze, tak jak to pokazano na rysunku:



za rządki taśmy (w tym przypadku) uważa się taki rządki, który ma dziurkę w zaznaczonym na rysunku kanale specjalnym;

- przesłanie zawartości akumulatora do miejsca pamięci, którego adres jest zawartością licznika rozkazów;
- zwiększenie zawartości licznika rozkazów o 1.

n) PRACA jest lampką, która jest zapalona, gdy maszyna pracuje.

#### 1.4. PAMIĘCI ZEWNĘTRZNE

Jako pamięci zewnętrzne w maszynie ZAM 41 stosuje się pamięć bębnową i pamięć taśmową.

##### 1.4.1. Pamięć bębnowa

Pamięć bębnowa składa się zwykle z 1-3 (lub więcej) bębnow magnetycznych stanowiących oddzielne moduły. Każdy bęben zawiera 128 ścieżek po 256 lub 512 słów. Łączna pojemność bębna wynosi więc 32 lub 64k. Ponadto każdy bęben ma kilka ścieżek zapasowych, które mogą być wykorzystane w przypadku awarii ścieżek podstawowych. Słowo na bębnie składa się z 24 bitów informacyjnych, 5 bitów kontrolnych, zapewniających poprawność zapisu i odczytu informacji oraz 2 bitów technicznych.

Każde słowo na bębnie ma określony adres (od 0 do 32767). Przesłanie informacji z pamięci operacyjnej na bęben lub odwrotnie odbywa się blokami; blok może się zaczynać od dowolnego słowa i może zawierać dowolną ilość słów.

Bęben obraca się z szybkością 1500 obrotów/min, stąd czas jednego obrotu wynosi 40 ms, a średni czas oczekiwania - 20 ms. Czas przesyłania 1 słowa wynosi około 156  $\mu$ s.

Dla uniknięcia pomyłkowego wymazania informacji na bębnie, poszczególne grupy ścieżek mają tzw. blokadę zapisu. Wciśnięcie odpowiedniego klucza powoduje blokadę zapisu; z zablokowanych ścieżek na bębnie można czytać informacje, natomiast zapis na tych ścieżkach nie jest możliwy.

#### 1.4.2. Pamięć taśmowa

Pamięć taśmowa jest oparta na jednostkach taśmy magnetycznej PT-2, których do maszyny jest zwykle podłączonych 5 do 7. Jednostki PT-2 pracują na standardowej 1/2 calowej taśmie magnetycznej i charakteryzują się następującymi parametrami:<sup>1)</sup>

- ilość ścieżek zapisu: 9 (8 informacyjnych, 1 kontrolna);
- gęstość zapisu 8 lub 16 rzędów/mm (przełączana kluczem),
- szybkość zapisu/odczytu 16 lub 32k rzędów/s,
- szybkość przesuwu taśmy 200 cm/s,
- szybkość przewijania taśmy 500 cm/s,
- przerwa międzyblokowa 38 mm;
- czas startu 6 ms,
- czas stopu 8 ms,
- pojemność pełnej szpuli - rzędu 10 milionów znaków.

Zapis i odczyt informacji na taśmie lub z taśmy odbywa się blokami zmiennej długości (co najmniej 3 słowa). Kolejne słowa bloku są pobierane z pamięci operacyjnej i zapisywane w 3 kolejnych rzędach taśmy lub odwrotnie.

<sup>1)</sup> Klepacz W.: Pamięci masowe maszyn cyfrowych. Warszawa, 1970, WNT, s. 90.



Aby uniemożliwić przypadkowe wymazanie informacji z taśmy, każda jednostka ma tzw. blokadę zapisu; włączenie blokady zapisu przez wciśnięcie odpowiedniego klucza umożliwia odczyt informacji z taśmy, nie pozwala natomiast na zapis informacji lub wykasowanie taśmy.

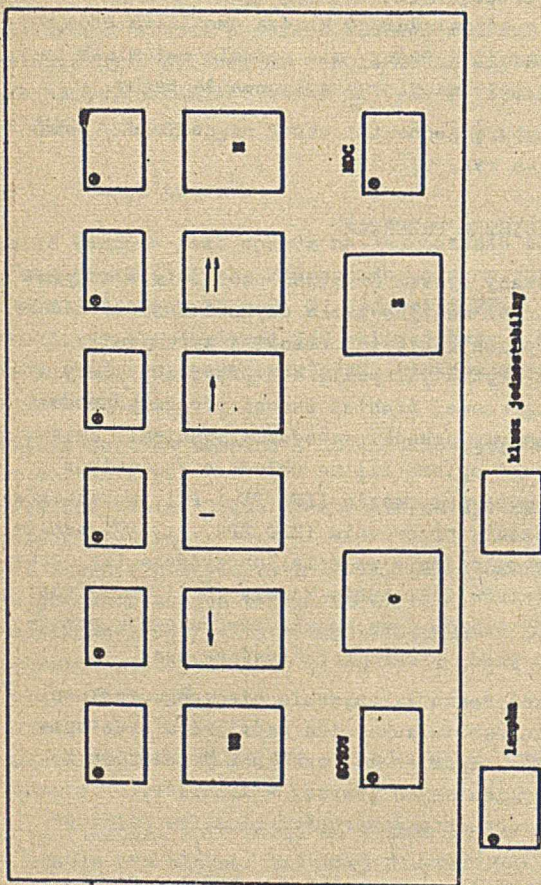
Klawiaturę jednostki taśmy magnetycznej pokazano na rys. 9.

### 1.5. SYSTEM PRZERWAŃ

Wbudowany system przerwania umożliwia wykonywanie przez maszynę wielu czynności jednocześnie (np. przesyłanie informacji i wykonywanie operacji arytmetycznych), a w przypadku pracy wieloprogramowej również chroni programy przed wzajemnym, szkodliwym oddziaływaniem. System przerwania opiera się na układzie wskaźników przyjęcia przerwania (PP, PD i PT), wskaźników zgłoszenia przerwania (ZPO, ZP1, ..., ZP7), wskaźników zgłoszenia wejścia lub wyjścia (ZW) oraz wskaźników zgłoszenia kanału (ZK). Określone układy stanów tych wskaźników powodują przerwanie pracy programu polegające na:

- zakończeniu wykonywania bieżącego rozkazu, wyzerowaniu wskaźnika nadmiarów N oraz wpisaniu odpowiednich wartości do niektórych z wymienionych powyżej wskaźników,
- zmniejszeniu zawartości licznika rozkazów o 1,
- wykonaniu rozkazu SSL do określonego dla danego rodzaju przerwania miejsca pamięci.

W przypadku jednoczesnego zgłoszenia kilku przerwania określona jest kolejność ich przyjmowania.



Zys. 9. Klasiatura jednostki tady magnetnyj P22

Jeśli układ stanów wymienionych wyżej wskaźników nie powoduje przerwania programu, wówczas maszyna pobiera i wykonuje rozkazy z miejsc pamięci wskazywanych kolejno przez zawartość licznika rozkazów.

Przerwanie pracy programu może być spowodowane jedną z następujących przyczyn:

- zgłoszenie przerwania przez urządzenie zewnętrzne lub kanał (np. po zakończeniu wykonywania jakiejś operacji),
- zgłoszenie przerwania przez operatora (naciśnięcie klucza ZGŁOSZENIE OPERATORA na stoliku operatora lub dowolnym innym module),
- zgłoszenie przerwania przez zegar.

W maszynie istnieje grupa rozkazów nazywanych nielegalnymi. Rozkazy te wykonywane są różnie, w zależności od stanu tzw. wskaźnika legalności  $F$ . Pbdany niżej opis działania tych rozkazów dotyczy tylko sytuacji, w których wskaźnik legalności jest zapalony ( $F=1$ ). Gdy wskaźnik legalności jest zgaszony ( $F=0$ ), wtedy próba wykonania rozkazu nielegalnego powoduje zapamiętanie ULR<sup>1)</sup> w zerowym miejscu pamięci, zapalenie wskaźnika legalności oraz skok do 40 miejsca pamięci. Działanie to jest więc podobne do przerwania pracy programu.

Do rozkazów nielegalnych należą:

- rozkazy typu OWW (operacje wejścia-wyjścia),
- rozkazy STO (stop), PDG (pisz w D oraz G) i PLA (pisz lampki z A),
- rozkazy zmieniające zawartość miejsc pamięci poza obszarem przeznaczonym dla programu, a

<sup>1)</sup> Opis ULR podany jest w rozdziale 3.

wyznaczonym pomnożoną przez 128 zawartością rejestrów D i G,

- rozkazy, w których pośrednie adresowanie powtarza się więcej niż 7 razy (takie rozkazy są traktowane jako nielegalne nawet przy zapalonym wskaźniku legalności).

#### 1.6. ADRESOWANIE SŁÓW W PAMIĘCI

Ze względu na budowę rozkazy maszyny ZAM 41 można podzielić na trzy grupy:

- rozkazy posiadające 15-bitową część adresową, która może przyjmować wartości od 0 do 32767,
- rozkazy posiadające 8-bitową część adresową, przyjmującą wartości od 0 do 255,
- rozkazy bezadresowe.

Wszystkie rozkazy grupy pierwszej mają 6-bitową część operacyjną, rozkazy grupy drugiej mają 12-bitową część operacyjną, natomiast rozkazy grupy trzeciej mają 6- lub 12-bitową część operacyjną. Część adresowa rozkazu określa adres pamięci lub parametr.

Dla wygody dalszego opisu ostatnie 15 bitów, obejmujące część adresową, a w niektórych przypadkach również fragment części operacyjnej - będziemy nazywali pseudoadresem.

Przed wykonaniem rozkazu pseudoadres może zostać zmodyfikowany i rozszerzony do 18 bitów. Jeżeli pseudoadres określa adres pamięci, to rozszerzenie takie powoduje, że może on przyjmować wartości od 0 do 262143.

Modyfikacja i rozszerzenie pseudoadresu odbywa się przez indeksowanie oraz pośrednie adresowanie. Indeksowanie polega na dodaniu do pseudo-

adresu zawartości ostatnich 18 bitów B-rejestru (z uwzględnieniem znaku B-rejestru); w niektórych przypadkach dodawana jest podwójna zawartość ostatnich 18 bitów B-rejestru. Pośrednie adresowanie polega na zastąpieniu pseudoadresu ostatnimi 15 lub 18 bitami (w zależności od spełnienia dodatkowych warunków), ze słowa pamięci wskazanego przez pseudoadres.

Wykonanie indeksowania oraz pośredniego adresowania jest uzależnione od wartości drugiego i trzeciego bitu w rozkazie; operacje te mogą być wykonywane kilkakrotnie (jeśli są spełnione odpowiednie warunki), jednak pośrednie adresowanie nie może być wykonane więcej niż 7 razy.

Otrzymany w wyniku indeksowania oraz pośredniego adresowania pseudoadres jest nazywany adresem efektywnym. Jeśli adres efektywny jest adresem pamięci (a nie parametrem), to może być jeszcze zwiększony o zawartość rejestru blokady dolnej D. Zwiększenie takie następuje wtedy, gdy wskaźnik legalności jest zgaszony, lub gdy wskaźnik legalności jest zapalony a jednocześnie zerowy bit w rozkazie ma wartość 1.

Wszystkie opisane powyżej modyfikacje pseudoadresu nie zmieniają zawartości pamięci, a jedynie zawartość odpowiednich rejestrów sterowania centralnego.

Powyższy opis nie jest dostatecznie precyzyjny. Ścisły opis indeksowania (modyfikacji przez B-rejestr) oraz pośredniego adresowania podany jest w punkcie 3.3.



## 2. OGÓLNE INFORMACJE O PROGRAMOWANIU

Opisaną w poprzednim rozdziale maszynę ZAM 41 programiści nazywają maszyną "surową". Programowanie na takiej maszynie jest dość skomplikowane, przede wszystkim ze względu na konieczność obsługi urządzeń zewnętrznych i zegara. Dla wygody pisania i uruchamiania programów użytkowych skonstruowano tzw. system operacyjny, realizujący m.in. obsługę wymienionych urządzeń. Niemal wszystkie programy użytkowe działają pod nadzorem systemu operacyjnego; na maszynie surowej działa system operacyjny i niektóre testy.

System operacyjny wchodzi w skład tzw. systemu oprogramowania maszyny ZAM 41, który zawiera również bibliotekę systemu oraz system wejść-wyjść (SWW). Ideowy schemat systemu oprogramo-

wania SO-141 przedstawiono na rys. 10. W dalszym ciągu pracy zostaną opisane niektóre elementy tego systemu.

## 2.1. SYSTEM OPERACYJNY I BIBLIOTEKA

### 2.1.1. System operacyjny (SO-141)

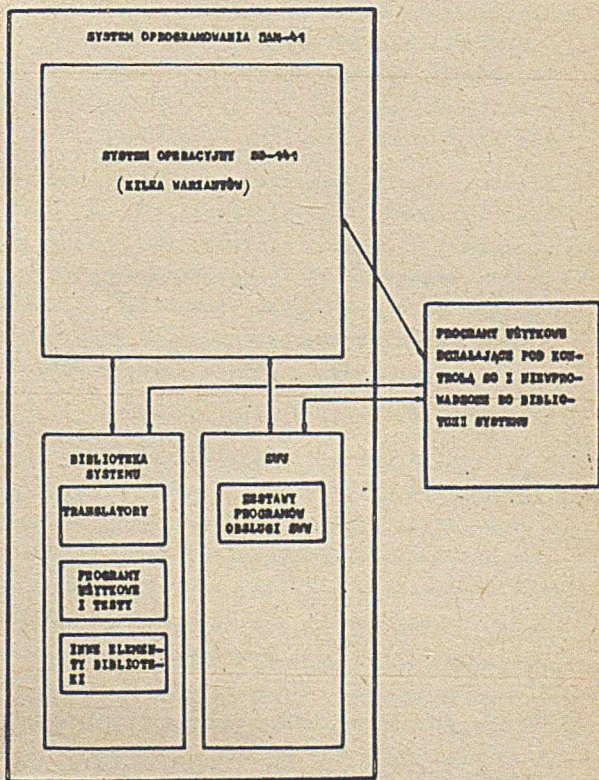
Programy systemu operacyjnego SO-141 (zwanego w skrócie SO) zostały napisane w języku PJEG (patrz 2.2.2) i w tej postaci zapisane za pomocą języka SMAD (patrz 2.2.12) na taśmach magnetycznych, w tzw. postaci źródłowej. Za pomocą specjalnych środków przekształcono postać źródłową na tzw. postać binarną. W czasie tego procesu przekształcania, zwanego generowaniem, ustawiono niektóre parametry tworzonego systemu.

Jednym z głównych elementów systemu operacyjnego jest tzw. "superwizor", tj. program zarządzający obsługą urządzeń wejścia i wyjścia, i realizujący podział czasu między poszczególne programy, jednocześnie załadowane do pamięci operacyjnej.

System operacyjny w postaci binarnej jest zapisany na taśmie magnetycznej, zwanej w dalszym ciągu taśmą systemu. Przed rozpoczęciem pracy system operacyjny zostaje załadowany na bęben, a jego część tzw. rezydent - do pamięci operacyjnej. Elementy przechowywane na bębnie są sprowadzane do pamięci operacyjnej wtedy, gdy mają być wykonywane.

Powyższy opis dotyczy systemu operacyjnego SO-141. Wcześniej istniał system SO-41, który jest jeszcze czasem eksploatowany, ale już od dość dawna nie jest rozwijany.





Dys. 90. Schemat ogólnej struktury oprogramowania SAM-44

### 2.1.2. Biblioteka systemu oprogramowania

Biblioteka systemu oprogramowania, zwana w skrócie biblioteką systemu, składa się ze zbiorów, które mogą być translatorami, programami użytkowymi lub tekstami. Każdy z wymienionych typów zbiorów ma określone własności.

Elementy biblioteki systemu są zapisane w postaci wynikowej (tj. binarnej), lub do niej zbliżonej na taśmie systemu, tzn. na tej samej taśmie magnetycznej co system operacyjny. Natomiast wersje źródłowe, zwane też plikami źródłowymi tych elementów, są zapisane w różnych językach na różnych taśmach magnetycznych.

W skład biblioteki systemu wchodzi:

- translatory języków programowania,
- funkcje języków,
- programy manipulacyjne,
- niektóre programy użytkowe,
- testy.

Każdy element biblioteki systemu ma swoją nazwę, przez którą jest wywoływany.

### 2.1.3. Edycje SO-141

System operacyjny wraz z biblioteką są stale rozwijane. Od czasu do czasu ukazują się kolejne edycje taśmy systemu. Są one oznaczone trzema literami, stanowiącymi tzw. numer edycji systemu. W chwili pisania tych słów przygotowywana jest edycja BCW, co oznacza, że od wprowadzenia SO-141 wydano przeszło 50 edycji. Generalną zasadą przestrzeganą przy wprowadzaniu nowej edycji jest zachowanie wszystkich możliwości starej edycji. W stosunku do poprzedniej

edycji systemu, nowa edycja może zawierać nowe elementy oprogramowania, nie zawiera natomiast błędów dostrzeżonych w starej edycji. Zapewnia to działanie wszystkich dotychczas działających programów. Może się jednak zdarzyć, że wskutek rozbudowy systemu, zmniejsza się obszar pamięci operacyjnej lub bębnowej przeznaczony dla programu.

#### 2.1.4. Wieloprogramowość

W trakcie generowania systemu operacyjnego określa się liczbę programów, jakie mogą być wykonywane jednocześnie, a ściślej mówiąc, które jednocześnie znajdują się w pamięci operacyjnej i są wykonywane na przemian, pod nadzorem systemu operacyjnego. Ze względu na stosunkowo małą pamięć operacyjną oraz niewielką liczbę urządzeń zewnętrznych, na ogół generuje się systemy dopuszczające maksymalnie pracę dwuprogramową. Podział maszyny na zestawy obsługujące poszczególne programy jest przeważnie następujący:

- a) reżim A: zestaw 0 zawiera połowę pamięci operacyjnej i bębnowej oraz połowę każdego rodzaju urządzeń, zaokrągloną w razie potrzeby w górę (np. gdy są trzy jednostki taśmy magnetycznej, to zestaw 0 zawiera dwie z nich); pozostała część pamięci operacyjnej i bębnowej oraz pozostałe urządzenia są przydzielane do zestawu 1;
- b) reżim B: zestaw 0 zawiera 4 jednostki taśmy magnetycznej, zestaw 1 - pozostałe; pamięć operacyjna i bębnowa oraz pozostałe urządzenia zewnętrzne są tak dzielone, jak w reżimie A;

- c) reżim C: całość maszyny obejmuje zestaw O, co oznacza pracę jednoprogramową (nie licząc systemu operacyjnego).

O ile wybór dopuszczalnych reżimów następuje w momencie generowania systemu operacyjnego, to wybór właściwego reżimu pracy następuje w momencie jego startowania (patrz punkt 5.1).

Adresowanie pamięci operacyjnej jest niezależne od tego, czy program działa samodzielnie, czy dzieli pamięć z innymi programami. Wynika to z automatycznego dodawania w określonych warunkach, zawartości rejestru dolnej blokady do adresu rozkazu. Niezależne adresowanie pamięci bębnowej oraz niezmienny sposób odwoływania się do urządzeń zewnętrznych realizuje system operacyjny. Dzięki temu programista pisząc program musi się jedynie troszczyć o ilość dostępnych słów w pamięci operacyjnej i bębnowej oraz liczbę urządzeń zewnętrznych, natomiast nie interesuje go, w którym zestawie program będzie działał.

## 2.2. TRANSLATORY

Jednym z zasadniczych elementów biblioteki systemu jest grupa translatorów dla różnych języków programowania. Poniżej podany jest bardzo krótki przegląd tych języków.

### 2.2.1. Język JOM

JOM (język operacyjny maszyny) służy do opisywania wykonywanych na maszynie problemów (wyjaśnienie pojęcia PROBLEM - patrz p. 2.4). W języku tym opisuje się zadania wchodzące w skład problemu oraz podaje się inne informacje

o problemie, takie jak tytuł problemu, urządzenia, podprogramy i dekodery stosowane dla symbolicznych wejść i wyjść, ograniczenia czasu oraz wyjścia itp.

### 2.2.2. Języki PJES i PJEG

Języki PJES (Podstawowy Język Symboliczny) i PJEG (Podstawowy Język Generowania) są językami symbolicznymi, w których każdy rozkaz odpowiada jednemu słowu maszyny. Języki te umożliwiają dostęp do każdego słowa lub nawet bitu, zarówno w pamięci operacyjnej jak i bębnowej. Obydwa języki dopuszczają różne sposoby adresowania symbolicznego, przy czym PJEG, który jest rozszerzeniem języka PJES, stwarza w tym zakresie szczególnie dużo udogodnień. Liczne dyrektywy (szczególnie w języku PJEG) umożliwiają optymalną gospodarkę pamięcią, wypisywanie informacji o programie wynikowym, omijanie części programu w czasie translacji itp. Język PJEG umożliwia tworzenie makrodefinicji i wykonywanie makrorozkazów; własności te ułatwiają tworzenie programów wynikowych.

### 2.2.3. Język SAS

Język SAS (System Adresów Symbolicznych) jest, podobnie jak PJES i PJEG, językiem symbolicznym. SAS jest wyposażony w rozbudowany aparat wbudowanych makrodefinicji i odpowiadających im makrodowołań. Aparat ten został pomyślany jako narzędzie umożliwiające łatwe korzystanie z obszernej biblioteki funkcji tego języka (stąd pochodzi jego formalna nazwa MSAS - Makro SAS). Ponadto SAS posiada własny system

gospodarki pamięcią operacyjną, co w połączeniu z podziałem programu na sekcje, w dużej mierze ułatwia programowanie.

#### 2.2.4. Język EOL

Język EOL (Expression Oriented Language) jest oryginalnym polskim rozwiązaniem, przeznaczonym do przetwarzania tekstów. EOL był przede wszystkim realizowany z myślą o pisaniu translatorów innych języków.

#### 2.2.5. Język COBOL

Język COBOL (Common Business Oriented Language) jest oparty na amerykańskiej propozycji standardu tego języka z 1967 roku. Zmiany w stosunku do tej propozycji dotyczą przede wszystkim opisu postaci danych na nośnikach zewnętrznych (karty, taśma papierowa, tabulogramy itp.) oraz częściowo opisu urządzeń zewnętrznych. Fragmenty programów w języku COBOL mogą być pisane w języku SAS, co pozwala na korzystanie z obszernej biblioteki funkcji tego języka.

Rozbudowany aparat do opisu złożonych struktur danych oraz operacje wejścia-wyjścia, przesyłania, sortowania itp. umożliwiają stosowanie języka COBOL (jak zresztą wszystkie realizacje tego języka) do programowania (a czasem nawet opisu) zagadnień z zakresu zastosowań gospodarczych.

#### 2.2.6. Język ALGOL

Język ALGOL (Algorithmic Language) jest wersją języka ALGOL 60, uwzględniającą specyfikę ma-

szyny ZAM 41, szczególnie w zakresie operacji wejścia-wyjścia. Język ALGOL, ze względu na rozbudowaną część proceduralną, jest przede wszystkim stosowany do programowania zagadnień naukowo-badawczych, konstrukcyjnych, ekonomicznych itp.

Translator języka ALGOL jest zrealizowany metodą interpretacyjną, co wprawdzie wydłuża czas obliczeń, ale znacznie skraca czas translacji. Rozmieszczanie danych w pamięci operacyjnej odbywa się dynamicznie.

#### 2.2.7. Język SAKO

Język SAKO (System Automatycznego Kodowania) został przeniesiony z maszyn ZAM 2 i ZAM 21 tak, aby programy z tamtych maszyn mogły działać na maszynie ZAM 41 po wprowadzeniu minimalnych zmian. Język SAKO jest rozwiązaniem polskim, zbliżonym do języka FORTRAN. Zawiera on rozbudowaną część proceduralną i dlatego, podobnie jak ALGOL, szczególnie nadaje się do programowania zagadnień naukowo-technicznych, konstrukcyjnych, ekonomicznych itp. Ponadto język SAKO zrealizowany na maszynie ZAM 41, posiada możliwość współpracy z taśmą magnetyczną, co znacznie rozszerza zakres zastosowań tego języka. Fragmenty programów w języku SAKO mogą być pisane w języku SAS, co umożliwia korzystanie z obszernej biblioteki funkcji tego języka.

#### 2.2.8. Język GPSS

Język GPSS (General Purpose System Simulator, wersja na maszynie ZAM 41, znana również pod

nazwą MODEST) jest wzorowany na rozpowszechnionym za granicą, szczególnie w USA, języku o tej samej nazwie. GPSS służy do symulacji różnych procesów dyskretnych, zależnych od przypadkowych lub zdeterminowanych zdarzeń zewnętrznych. Wynikiem symulacji są różne dane statystyczne o zachowaniu się badanego procesu. W trakcie działania symulacji zapewniona jest wygodna komunikacja operatora z systemem.

#### 2.2.9. Język CEMMA

Język CEMMA jest przeznaczony do rozwiązywania zadań symulacji procesów ciągłych i impulsowo-ciągłych, opisywanych równaniami różniczkowymi zwyczajnymi liniowymi i nieliniowymi. Język CEMMA umożliwia dokonanie zwięzłego opisu bloków całkujących, działań arytmetycznych, operacji logicznych itp. Możliwe jest również zadanie kryterium optymalizacji, według którego nastąpi optymalizacja do 6 parametrów modelu.

#### 2.2.10. Język ASTEK

Język ASTEK (Analiza Statystyczna Eksperymentów) służy do statystycznego opracowania zbiorów danych eksperymentalnych, uzyskiwanych z różnego rodzaju doświadczeń technicznych, przemysłowych, laboratoryjnych itp. W szczególności w języku tym można testować statystycznie zbiory danych, liczyć średnie, wariancje i dystrybuanty, przeprowadzać analizę regresji i korelacji, wyznaczać gęstość widmową itp.



### 2.2.11. Języki BIGRAF i PLEXOL

Języki BIGRAF i PLEXOL są przeznaczone do rozwiązywania zagadnień z zakresu sztucznej inteligencji, np. pisania fragmentów translatorów dla języków wyższych rzędów. Obiektem działania programów w języku BIGRAF są dane w postaci grafów o dwóch gałęziach wychodzących z każdego wierzchołka. Język PLEXOL jest uogólnieniem BIGRAF-u i dopuszcza dowolną liczbę gałęzi wychodzących z poszczególnych wierzchołków.

Obydwa języki są iteracyjne i mają strukturę podobną do języka ALGOL-60, natomiast nie są językami rekursywnymi.

### 2.2.12. Język SMAD

Język SMAD (System Magazynowania i Aktualizacji Dokumentów) znany również pod nazwą POPR lub POPRAWEK służy do magazynowania programów na taśmie magnetycznej. Programy te, traktowane jak teksty, mogą być uzupełniane i poprawiane, co w istotny sposób upraszcza proces programowania. Ponadto SMAD umożliwia wykonanie dodatkowych czynności, jak np. sporządzenie dokumentacji programu.

### 2.2.13. Języki BIB1 i BIB2

Języki BIB1 i BIB2 (Bibliotekarz 1 i 2) służą do wprowadzania nowych zbiorów do biblioteki systemu (zob. rozdział 4). Dla programisty nie przewidującego włączenia programów użytkowych do biblioteki systemu, języki te nie mają zastosowania.

## 2.2.14. Translatory WEBIN, WYBIN i WYSE

Translatory WEBIN (wejście binarne) i WYBIN (wyjście binarne) służą do operowania tzw. taśmą binarną, tj. taśmą z programem całkowicie lub częściowo przetłumaczonym. Taśmy binarne stosuje się przede wszystkim dla programów napisanych w językach o stosunkowo długim procesie translacji (np. SAS, COBOL). Taśma taka jest wprowadzana do maszyny znacznie szybciej, a zapisane na niej sumy kontrolne zapewniają ponadto prawidłowość jej wprowadzenia.

Ponieważ translator języka SAS prowadzi własną gospodarkę pamięcią bębnową, rozmieszczenie programu na bębnie jest uzależnione od aktualnej konfiguracji systemu sprowadzonego na bęben. W tej sytuacji programy binarne z języka SAS mogłyby nie działać prawidłowo. Aby temu zaradzić wprowadzono translator WYSE, który przed wypisaniem taśmy binarnej przenosi program wynikowy na ustalone miejsce oraz dołącza do niego sekcję startową. Po wyprowadzeniu taśmy binarnej i wprowadzeniu jej ponownie zaczyna działać sekcja startowa, która z powrotem przenosi program na właściwe miejsce.

## 2.2.15. Język SMAO

Język SMAO (System Magazynowania Opisów) służy do przechowywania opisów poszczególnych translatorów, programów itp. na taśmie magnetycznej. Zmagazynowane opisy mogą być bieżąco uzupełniane i poprawiane, co gwarantuje ich aktualność nawet jeśli zmiany w dokumentacji następują często. Za pomocą języka SMAO produkowana jest tzw. informacja ekspresowa dotycząca maszyny

ZAM 41, a obejmująca te fragmenty opisów oprogramowania, które jeszcze nie zostały opublikowane.

### 2.3. PROGRAMY UŻYTKOWE I TESTY

Oprócz wymienionych w p. 2.2 translatorów w skład biblioteki systemu wchodzi kilka programów użytkowych, które można podzielić na następujące trzy kategorie:

- programy manipulacyjne,
- programy do przetwarzania danych,
- testy.

#### 2.3.1. Programy manipulacyjne

Programami manipulacyjnymi nazwano często używane programy, służące do wykonywania pomocniczych czynności typu organizacyjnego. Czynności te dotyczą na ogół operowania taśmą magnetyczną, sporządzania dokumentacji, uruchamiania programów itp.

Programy manipulacyjne są podzielone na cztery grupy: MAN, MAN1, VMAN oraz inne. W skład tych grup m.in. wchodzi (podane poniżej nazwy, chociaż żargonowe, zostały już zaakceptowane przez środowisko programistów i użytkowników i dlatego są przytoczone w oryginalnym brzmieniu):

##### a) grupa MAN:

- EDYTOR-PJES (CIA) - dzieli programy napisane w języku PJES na strony, zgodnie z wymaganiami SMAD,
- SPRAWDZATOR (SPR) - porównuje i wykrywa niezgodności między dwoma taśmami z tym samym programem napisanym w języku PJES,

- WYPISYWANIE TM (WTM) - wypisuje określone bloki z taśmy magnetycznej,
- WYSZUKIWANIE TEKSTU (SZUK) - wyszukuje określoną sekwencję znaków w tekście zbudowanym zgodnie z wymaganiami języka POPRAWEK;

b) grupa MAN1:

- WYPISYWACZ SPISU STRON SMAD (STR) - wypisuje spis stron z tekstu zapisanego w języku SMAD na taśmie magnetycznej,
- POWIELACZ TMS (TMS) - powiela taśmę magnetyczną systemu,
- WYPISYWACZ SMAD (WYPS) - wypisuje kolejne strony tekstu zapisanego w języku SMAD na taśmie magnetycznej,
- DOKUMENT (DOK) - sporządza dokumentację tekstu wejściowego,
- METRYKOWACZ TAŚMY MAGNETYCZNEJ (MKZ) - metrykuje taśmę magnetyczną zgodnie z przyjętym standardem,
- WYPISYWACZ SPISU STRON SMAO (STR8) - wypisuje spis stron z tekstu zapisanego w języku SMAO na taśmie magnetycznej;

c) grupa VMAN:

- METRYKOWACZ TM W STANDARDZIE (VMKZ) - metrykuje taśmę magnetyczną zgodnie z przyjętym standardem,
- UNIWERSALNY PROGRAM WYDAWNICZY (VWYD) - redaguje tekst wypisując go na różnych urządzeniach,
- METRYKOWACZ TM W STANDARDZIE SMAD (VMEP, VMKD) - zapisuje na początku lub w miejscu zatrzymania taśmy magnetycznej znacznik końca dokumentu, zgodnie z wymaganiami języka SMAD,

- SCALATOR TAŚM ZAPISANYCH W STANDARDZIE SMAD (VSCP) - łączy dwie taśmy magnetyczne z zapisanymi tekstami w języku SMAD,
- WYPISYWACZ SPISU STRON TM ZAPISANEJ W STANDARDZIE SMAD (VWYP) - wypisuje spis stron z taśmy magnetycznej zapisanej w języku SMAD,
- PERFORATOR ZAWARTOŚCI TM ZAPISANEJ W STANDARDZIE SMAD (VPP0) - perforuje zawartość taśmy magnetycznej zapisanej w języku SMAD,
- WYPISYWACZ TAŚMY SYSTEMU (VWTS) - powie-la lub sprawdza taśmę systemu, wypisuje katalogi, perforuje wybrane zbiory z taśmy systemu,
- KONTROLA ZAPISU PLIKU (VKZP) - kontroluje i reprodukuje pliki zapisane na taśmie magnetycznej,
- ZAPIS BINARNEJ NA MAGNETYCZNEJ (VZBT) - zapisuje informacje z binarnej taśmy papierowej na taśmie magnetycznej zgodnie z wymaganiami języka SMAD;

d) inne:

- EDYTOR NA TAŚMĘ PAPIEROWĄ - SAS-EOL (PTSE) - dzieli programy napisane w języku SAS-EOL na strony, zgodnie z wymaganiami języka SMAD; wyniki wypisuje na taśmie papierowej,
- EDYTOR NA DRUKARKĘ - SAS-EOL (DWSE) - dzieli programy napisane w języku SAS-EOL na strony, zgodnie z wymaganiami języka SMAD; wyniki wypisuje na drukarce wierszowej, uzupełniając je skorowidzem etykiet,
- SPRAWDZATOR - SAS-EOL (SPSE) - porównuje i wykrywa niezgodności między dwiema taś-

mami z tym samym programem, napisanym w języku SAS-EOL,

- EDYTOR-ALGOL (EDAL) - dzieli programy napisane w języku ALGOL na strony, zgodnie z wymaganiami języka SMAD; wyniki wypisuje na perforatorze oraz na drukarce wraz ze skorowidzem zadeklarowanych nazw,
- EDYTOR-COBOL (EDCO) - dzieli programy napisane w języku COBOL na strony, zgodnie z wymaganiami języka SMAD; wyniki wypisuje na taśmie papierowej.

### 2.3.2. Programy do przetwarzania danych

W bibliotece systemu znajduje się grupa programów parametryzowanych znanych pod nazwą OPUS, służących do wykonywania niektórych typowych czynności z zakresu przetwarzania danych. Najważniejszymi programami z tej grupy są:

- PPO1 - czyta dane z kart perforowanych, kontroluje ich poprawność, poprawne dane zapisuje na taśmie magnetycznej,
- PPO3 - czyta dane z taśmy perforowanej, kontroluje ich poprawność, poprawne dane zapisuje na taśmie magnetycznej,
- PPO9 i PP12 - czytają dane z taśm magnetycznych, przekształcają je i wynik zapisują na taśmach magnetycznych,
- PP10 - kontroluje zbiory na taśmie magnetycznej,
- PP11 - powiela zbiory na taśmie magnetycznej,
- PP13 - czyta dane z taśmy magnetycznej, redaguje je, wykonuje odpowiednie operacje arytmetyczne i wyniki wypisuje na drukarce wierszowej,

- PP15 - sortuje dane zapisane na taśmie magnetycznej oraz łączy dwa posortowane zbiory danych.

### 2.3.3. Testy

Włączone do biblioteki systemu testy tworzą tzw. system testów K-500. Składa się on z dyrygenta oraz z autonomicznych testów, sprawdzających poszczególne urządzenia. Ponieważ K-500 pracuje pod nadzorem systemu operacyjnego zakłada się, że w pewnym zakresie maszyna jest sprawna, szczególnie część pamięci operacyjnej, podstawowe operacje arytmetyczne i logiczne itp. K-500 sprawdza więc tylko część pamięci operacyjnej i bębnowej oraz urządzenia zewnętrzne. Ponieważ K-500 pracuje pod nadzorem systemu operacyjnego, sprawdzenie urządzeń zewnętrznych nie jest zupełne. W celu pełniejszego sprawdzenia maszyny należy zastosować istniejące testy działające poza systemem operacyjnym.

W skład systemu testów K-500 wchodzi:

- test arytmometru zmiennego przecinka,
- test pamięci operacyjnej,
- test pamięci bębnowej,
- test pamięci taśmowej,
- test perforatora i czytnika taśmy papierowej,
- test czytnika taśmy papierowej,
- test drukarki DW-1,
- test drukarki DW-2,
- test czytnika kart.

## 2.4. JEDNOSTKA PRZETWARZANIA

Jednostką przetwarzania, tzn. taką jednostką, którą można oddzielnie operować, jest tzw. problem. Problem dzieli się na kroki, z których każdy oznacza bądź wykonanie programu stanowiącego element biblioteki systemu, bądź programu będącego wynikiem działania translatora.

### 2.4.1. Język operacyjny maszyny (JOM)

Parametry wykonywanego problemu opisuje się w specjalnym języku, zwanym Językiem Operacyjnym Maszyny (JOM). Opis problemu stanowi tzw. czołówkę, która musi być wyperforowana na taśmie papierowej w kodzie M-2 i musi być wprowadzona na początku problemu z pierwszego czytnika taśmy papierowej w zestawie, w którym dany problem ma być wykonywany. Czołówka składa się ze zdań, które mają następujące znaczenie:

- a) zdanie PROBLEM określa kolejne kroki, które w ramach danego problemu będą wykonywane; w szczególności jednym z kroków może być JOM, co oznacza, że po wykonaniu poprzednich kroków należy powtórnie wczytać czołówkę, która może zmieniać poprzednio ustalone parametry problemu;
- b) zdanie WEJSCIA (PROGRAMOW, TRANSLATOROW) określa symboliczne wejście, stosowane w danym problemie (patrz 2.4.2);
- c) zdanie WYJSCIA (PROGRAMOW, TRANSLATOROW) określa symboliczne wyjście, stosowane w danym problemie (patrz 2.4.2);
- d) zdanie OGRANICZ uniemożliwia zbyt długie działanie programów w ramach danego proble



mu i wyprowadzanie zbyt dużej liczby wyników; ograniczenia te stosuje się przede wszystkim wówczas, gdy program nie jest jeszcze uruchomiony i istnieje obawa jego "zapętlenia";

- e) zdanie TYTUL powoduje wypisanie określonego tekstu na początku wyprowadzanych za pomocą symbolicznego wyjścia O wyników.

#### 2.4.2. Symboliczne wejścia i wyjścia (SWW)

Jednym z założeń systemu operacyjnego było umożliwienie takiego pisania programów, aby były one niezależne od rodzaju stosowanych urządzeń, nośników z danymi lub wynikami oraz kodów. W tym celu wprowadzono specjalny aparat zwany systemem obsługi wejścia-wyjścia (SOWW). System SOWW składa się z tzw. symbolicznych wejść i wyjść, które są definiowane w czołówce problemu, a wykorzystywane w programie. Każde symboliczne wejście i wyjście ma przypisany przez programistę piszącego program jednoznaczny numer. Aby umożliwić pełną niezależność programu od stosowanych nośników zewnętrznych wprowadzono jeszcze pojęcie kodu wewnętrznego, tj. wewnętrznej reprezentacji każdego znaku. Stosowane są dwa kody wewnętrzne - sześciobitowy (KW-6) i ośmiobitowy (KW-8). Przy pisaniu programu wystarczy więc, w zasadzie, znać stosowany kod wewnętrzny oraz numer symbolicznego wejścia lub wyjścia. Wyniesiona do czołówki problemu, a więc poza program, definicja symbolicznych wejść i wyjść może być łatwo zmieniona, co nie wpływa na sposób wykonania samego programu. Definicja symbolicznego wejścia lub wyjścia obejmuje: symbol urządze-

nia (np. CPB - drugi w danym zestawie czytnik taśmy papierowej), rodzaj podprogramu (np. podprogram czytania taśmy po jednym znaku) oraz symbol dekodera (np. z kodu M-2 na KW-6). Symboliczne wejścia i wyjścia są definiowane oddzielnie dla translatorów, oddzielnie dla programów problemów, które nie są translatorami, a które nazywać będziemy po prostu programami. Liczba wejść i wyjść, zarówno dla translatorów jak i programów, może wynosić do 16, co daje bardzo dużą elastyczność ich stosowania.

### 3. PROGRAMOWANIE

#### 3.1. POSTAĆ INFORMACJI W MASZYNIE

Przez postać informacji w maszynie rozumie się ogólnie przyjętą interpretację znaczenia ciągu 24 lub 48 bitów, zapisanych w jednym lub dwóch słowach maszyny. Ciągi te mogą być interpretowane jako:

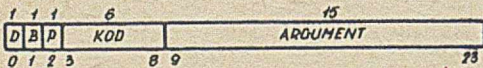
- rozkazy,
- liczby pojedyncze (krótkie),
- liczby podwójne (długie),
- liczby zmiennoprzecinkowe,
- teksty w KW-6,
- teksty w KW-8,
- inne.

Postać informacji przedstawiono graficznie; na górze rysunku zaznaczono liczbę bitów, a na dole podano numerację bitów.

### 3.1.1. Rozkaz

Rozkaz może mieć jedną z dwóch postaci:

a) rozkazy z nierozszerzoną częścią operacyjną



D oznacza bit D-modyfikacji; nie ma on wpływu na wykonanie rozkazu w programie użytkownika; może być wykorzystany np. dla zaznaczenia konieczności przeadresowania rozkazu (por. opis rozkazu SDY przeadresuj);

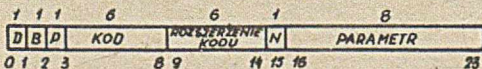
B oznacza bit B-modyfikacji (modyfikacji B-rejestrem);

P oznacza bit P-modyfikacji (pośredniego adresowania);

KOD oznacza wartość części operacyjnej rozkazu; dopuszczalne wartości dziesiętne części operacyjnej, odpowiadające im wartości oktalne i nazwy podano w tabeli 7.10.1;

ARGUMENT jest albo adresem, albo parametrem rozkazu;

b) rozkazy z rozszerzoną częścią operacyjną



Znaczenie D, B i P jest takie samo jak w punkcie a.

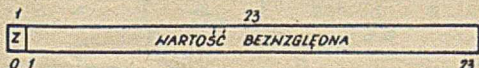
Znaczenie KOD-u jest analogiczne, jak w punkcie a, ale dopuszczalne są tylko dwie wartości (48 i 63) odpowiadające operacjom ZZR i OWW.

KOD nie wyznacza jednoznacznie operacji, lecz grupę operacji. Operacja jest wyznaczona dopiero przez KOD i ROZSZERZENIE KOD-u łącznie. Wartości dziesiętne i oktalne ROZSZERZENIA KOD-u oraz nazwy operacji podano w tabelach 7.10.2 i 7.10.3.

N jest bitem nie mającym wpływu na wykonanie operacji.

PARAMETR jest parametrem operacji.

### 3.1.2. Liczba pojedyncza (krótka)



Z oznacza znak liczby; Z=0 oznacza +, Z=1 oznacza -.

WARTOŚĆ BEZWZGLĘDNA jest liczbą całkowitą nie mniejszą niż 0 i nie większą niż 8388607.

Liczba pojedyncza (krótka) x może więc przybierać wartości z przedziału

$$-8388607 \leq x \leq 8388607$$

### 3.1.3. Liczba podwójna (długa)

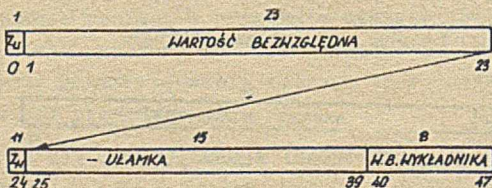


Z oznacza znak liczby; znaki w obu słowach powinny być jednakowe.  $Z=0$  oznacza +,  $Z=1$  oznacza -.

WARTOŚĆ BEZWZGLĘDNA, składająca się z części starszej (x) oraz części młodszej (y) jest równa  $8388608x+y$ ; x i y są liczbami całkowitymi nie mniejszymi od 0 i nie większymi od 8388607. Dlatego liczba podwójna (długa) może przybierać wartość z przedziału

$$-70368744177663 \leq z \leq 70368744177663$$

### 3.1.4. Liczba zmiennoprzecinkowa



$Z_u$  oznacza znak części ułamkowej;  $Z_u=0$  oznacza +,  $Z_u=1$  oznacza -.

$Z_w$  oznacza znak wykładnika;  $Z_w=0$  oznacza +,  $Z_w=1$  oznacza -.

WARTOŚĆ BEZWZGLĘDNA UŁAMKA jest liczbą 38 bitową; przyjmuje się, że jest to liczba ułamkowa, przy czym najstarszy jej bit jest pierwszym bitem po przecinku. Jeśli przez x oznaczyć liczbę całkowitą zapisaną w bitach 1-23 i 25-39, to WARTOŚĆ BEZWZGLĘDNA UŁAMKA =  $x \cdot 2^{-38}$ . Dokładność przedstawienia ułamka (tj. najmniejsza różnica między dwoma różnymi ułamkami) wynosi  $2^{-38} \approx 4 \cdot 10^{-12}$

WARTOŚĆ BEZWZGLĘDNA WYKŁADNIKA (W.B.WYKŁADNIKA) jest 8 bitową liczbą całkowitą, nie mniejszą od 0 i nie większą od 255.

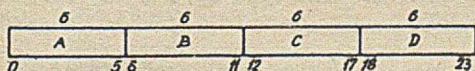
Jeśli przez  $u$  oznaczymy część ułamkową (wraz ze znakiem), a przez  $w$  - wykładnik (również ze znakiem), to liczba zmiennoprzecinkowa  $z$  wyrażona jest wzorem:

$$z = 2^w \cdot u$$

Zakres liczb zmiennoprzecinkowych jest określony wzorem:

$$2 \cdot 10^{-77} \approx 2^{-255} (1 - 2^{-38}) \leq |z| \leq 2^{255} (1 - 2^{-38}) \approx 5 \cdot 10^{76}$$

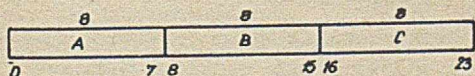
### 3.1.5. Tekst w KW-6



A, B, C, D oznaczają znaki zapisane w 6-bitowym kodzie wewnętrznym KW-6.

Kod KW-6 jest podany w punkcie 6.2.

### 3.1.6. Tekst w KW-8

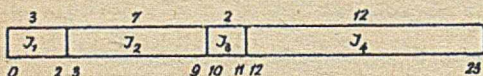


A, B, C oznaczają znaki zapisane w 8-bitowym kodzie wewnętrznym KW-8. Kod KW-8 jest podany w punkcie 6.4.

### 3.1.7. Inne

Programista może również w dowolny inny sposób zapisywać informacje w słowach maszyny. Przykła-

dem może być:



$I_1, I_2, I_3, I_4$  stanowią informacje interpretowane przez program.

### 3.2. REJESTRY, WSKAŹNIKI, ULR

Rejestry są urządzeniami służącymi do przechowania w maszynie niektórych informacji. Rozróżnia się dwa rodzaje rejestrów:

- rejestry zrealizowane za pomocą sprzętu,
- rejestry zrealizowane za pomocą programów.

Dla programisty rodzaj realizacji rejestru jest nieistotny.

Każdy rejestr składa się z określonej ilości bitów. Rejestry jednobitowe nazywane są wskaźnikami.

Funkcję podobną do rejestrów spełnia uzupełniony licznik rozkazów (ULR), który będzie opisany razem z rejestrami.

Krótki opis poszczególnych rejestrów dostępnych dla programisty podaje poniższa tabelka:

<u>Symbol</u>	<u>Nazwa i funkcja</u>	<u>Liczba bitów</u>	<u>Numeracja bitów</u>
A	Akumulator, podstawowy rejestr do działań arytmetycznych i logicznych na liczbach i słowach krótkich	1+23	0-23



<u>Symbol</u>	<u>Nazwa i funkcja</u>	<u>Liczba bitów</u>	<u>Numeracja bitów</u>
M	Mnożnik, podstawowy (obok A) rejestr do działań arytmetycznych na liczbach krótkich.	1+23	0-23
AM	Akumulator długi, złożony z rejestrów A i M. Podstawowy rejestr do działań arytmetycznych na liczbach stałoprzecinkowych 46-bitowych oraz liczbach zmiennoprzecinkowych.	2+46	0-47
B	B-rejestr, służy do modyfikacji adresów.	1+23	0-23
LR	Licznik rozkazów, zawiera adres bieżąco wykonywanego rozkazu.	15	9-23
RR	Rejestr rozkazów, zawiera rozkaz pobrany do wykonania.	24	0-23
N	Wskaźnik nadmiaru, zawiera informację o przekroczeniu zakresu rejestru A lub B.	1	
F	Wskaźnik legalności, określa jeden z dwóch stanów maszyny: działanie superwizora (F=1) lub programu użytkowego (F=0). Może być ustalany programowo (tylko przez superwizor) lub	1	

<u>Symbol</u>	<u>Nazwa i funkcja</u>	<u>Liczba bitów</u>	<u>Numeracja bitów</u>
	na stoliku operatora. Niniejsza publikacja dotyczy stanu programu użytkowego, dlatego w dalszym ciągu zakłada się, że $F=0$ .		
DATA	Rejestr daty zawiera- jący aktualną datę, wprowadzoną przez ope- ratora w czasie ściąga- nia systemu z bębna; da- ta zapisana jest nastę- pująco: ROK - bity 0-8 ( $0 < \text{ROK} < 100$ ) DZIEŃ - bity 9-23 ( $0 < \text{DZIEŃ} < 367$ ).	24	0-23
ZEG	Rejestr zegara zawiera czas realny mierzony w jednostkach równych 0.02s; co 0.02 s do za- wartości rejestru doda- wana jest liczba 1. Re- jestr zegara jest usta- wiany przez operatora w czasie ściągania systemu z bębna.	29	1-29
ZX[I] I=0,1, ...31	Rejestry warunków związa- ne z realizacją przesłań zewnętrznych.	7	11-17
BTM	Licznik bloków na taśmie magnetycznej, prowadzony jest oddzielnie dla każ- dej jednostki taśmy ma-	24	0-23

<u>Symbol</u>	<u>Nazwa i funkcja</u>	<u>Liczba bitów</u>	<u>Numera- cja bitów</u>
	gnetycznej. Zawiera on numer bloku, do którego odczytu lub zapisu ustawiona jest taśma.		
ULR	<p>Uzupełniony licznik rozkazów, może być zapisany w dowolnym słowie pamięci; zawiera on następujące informacje:</p> <p>bit 2 - zawiera "1",</p> <p>bit 5 - zawiera bieżącą zawartość wskaźnika N,</p> <p>bity 9-23 - zawierają bieżącą zawartość rejestru LR,</p> <p>pozostałe bity zawierają zera.</p>	24	0-23

Niektóre inne rejestry i wskaźniki, wprowadzone wyłącznie dla wygody opisu, zostały omówione w innych punktach niniejszej publikacji.

### 3.3. SPOSÓB REALIZACJI ROZKAZU

Czynności maszyny, związane z wykonaniem każdego rozkazu są następujące:

- pobranie rozkazu z miejsca pamięci, określonego przez aktualną zawartość licznika rozkazów i wpisanie go do rejestru rozkazów;

- wykonanie pobranego rozkazu, tj.:
  - obliczenie efektywnego adresu określonego przez część adresową rozkazu,
  - wykonanie operacji określonej przez kod rozkazu.

W trakcie wykonywania operacji określonej przez rozkaz, ustalana jest nowa zawartość licznika rozkazów. Gdy nie jest to rozkaz sterujący, wówczas zawartość licznika rozkazów zwiększa się o jeden.

Sposób obliczania adresu efektywnego zostanie przedstawiony schematycznie. W tym celu wprowadzono następujące rejestry:

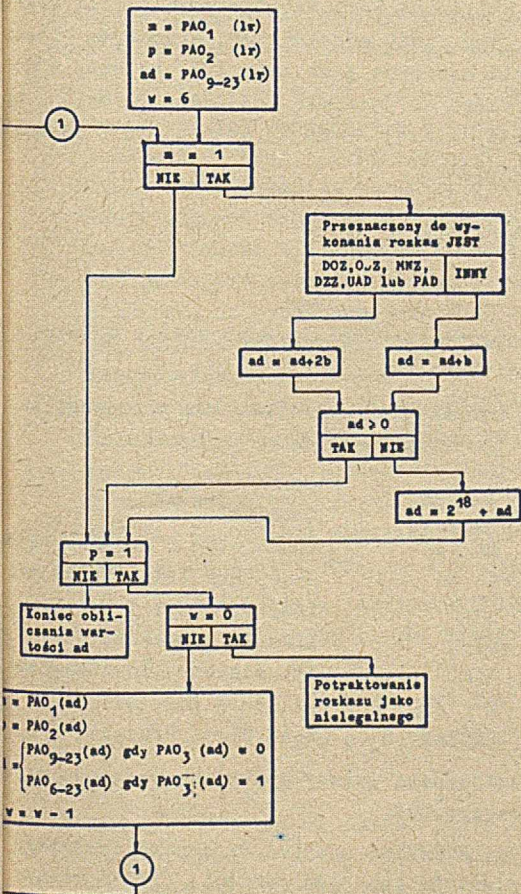
- rejestr adresu, zawierający 18 bitów ponumerowanych od 6 do 23,
- wskaźnik modyfikacji,
- wskaźnik pośredniego adresowania,
- wskaźnik liczby pośrednich adresowań.

Zawartość tych rejestrów będzie oznaczana odpowiednio przez ad, m, p oraz w.

Ponadto przyjęto następujące oznaczenia:

- lr            oznacza zawartość licznika rozkazów,
- $PAO_m(n)$    oznacza m-ty bit n-tego słowa PAO,
- $PAO_{m-k}(n)$  oznacza ciąg od m-tego do k-tego bitu n-tego słowa PAO
- b            oznacza zawartość B-rejestru, bez uwzględnienia bitów 1 do 5 (tzn. tylko bit znaku oraz bity 6 do 23).

W sieci działań na str.67 przedstawiono sposób obliczenia wartości ad.



Rys.11. Schemat obliczania adresu efektywnego rozkazu

Jeśli wykonywana operacja nie jest typu Z to obliczona wartość ad stanowi adres efektywny, tzn. jest adresem lub parametrem rozkazu. Jeśli operacja jest typu ZZR, to bity 9 do 15 wartości ad stanowią rozszerzenie kodu operacji, a bity 16 do 23 wyznaczają parametry operacji.

### 3.4. LISTA ROZKAZÓW

#### 3.4.1. Symbolika

Przy opisie rozkazów zastosowano opisaną w poprzedniej symbolikę.

a) Dużymi literami oznaczono poszczególne rejestry oraz miejsca pamięci:

A	akumulator,
M	mnożnik,
AM	akumulator i mnożnik razem,
B	B-rejestr (rejestr modyfikacji),
LR	licznik rozkazów,
ULR	uzupełniony licznik rozkazów,
N	wskaźnik nadmiaru,
PAO(x)	x-te słowo pamięci.

Ponadto dla wygody opisu wprowadzono rejestry robocze ROB.

Tymi samymi symbolami oznaczono zawartość rejestrów lub słów pamięci, traktowaną jako ciąg bitów lub liczby całkowite, których wyrażeniem w układzie binarnym są te ciągi bitów; bit stojący na pozycji zerowej w rejestrach A, M i B oraz w słowach pamięci jest więc traktowany jako współczynnik przy najmniejszej potęgze 2 ( $2^{23}$  lub  $2^{47}$ ), a nie jako znak.

Oznaczenie tym samym symbolem trzech różnych pojęć (rejestru, ciągu bitów lub liczby) nie wywołuje nieporozumień, gdyż z kontekstu zawsze wynika znaczenie omawianego pojęcia.

- b) Małymi literami a, m, b i pao(x) oznaczono zawartość odpowiednio A, M, B i PAO(x) traktowaną jako liczby całkowite, których rozwinięciem w układzie binarnym jest ciąg bitów stojących na pozycjach 1-23, a znakiem jest "+" jeśli bit zerowy jest zerem, lub "-" jeśli bit zerowy jest jednością. Podobnie przez am oznaczono zawartość rejestru AM, traktowaną jako liczba całkowita, której rozwinięciem w układzie binarnym jest ciąg bitów stojących na pozycjach 1-23 akumulatora i 1-23 mnożnika; znakiem tej liczby jest "+" jeśli zerowy bit akumulatora jest zerem, a "-", jeśli ten bit jest jednością. Zerowy bit mnożnika jest przy tym oznaczeniu nieistotny.

Jeśli Y jest dowolną z liczb A, M, B lub PAO(x), a y - odpowiadającą jej liczbą a, m, b lub pao(x), to

$$Y = \begin{cases} y & \text{gdy } y > 0 \text{ lub } y = +0 \\ 2^{23} - y & \text{gdy } y < 0 \text{ lub } y = -0 \end{cases}$$

Podobna równość dla liczby AM nie zachodzi.

- c) Indeksy dopisane do dużych liter oznaczają część rejestru, zawierającą pozycje wskazane przez te indeksy; duża litera z indeksami może również oznaczać zawartość części rejestru, traktowaną tak, jak to objaśniono w punkcie a.

Małe litery z indeksami oznaczają liczby całkowite wg interpretacji podanej w punkcie b, ale w odniesieniu do części rejestru.

- d) Przez  $am_z$  oznaczono liczbę zmiennoprzecinkową (patrz p. 3.1.4) zapisaną w AM; jej część ułamkową oznaczono przez  $am_u$ , a wykładnik - przez  $am_w$ .

Zachodzi więc związek

$$am_z = am_u \cdot 2^{am_w}$$

Przez liczbę znormalizowaną rozumie się taką liczbę, że  $\frac{1}{2} \leq am_u < 1$  lub  $am_u = +0$  i  $am_w = -255$ .

Podobne oznaczenia zastosowano do liczb zapisanych w pamięci ( $pao_z(x)$ ,  $pao_u(x)$ ,  $pao_w(x)$ ).

- e) W opisie rozkazów zastosowano symbolikę ALGOL-u, np.  $LR:=LR+1$  oznacza: zwiększ zawartość licznika rozkazów o jeden.
- f) Jako symboli działań użyto:

$+, -, \cdot$  oraz  $:$  dla działań arytmetycznych (suma, różnica, iloczyn i iloraz)

$\vee, \wedge$  oraz  $\oplus$  dla działań logicznych (suma, iloczyn i różnica symetryczna).

Wartość bezwzględną liczby  $y$  oznaczono przez  $|y|$ , a część całkowitą - przez  $E(y)$  (częścią całkowitą liczby ujemnej  $y$  jest najmniejsza liczba całkowita nie mniejsza od  $y$ ).

Kolejność wykonywania działań jest zgodna z ogólnie przyjętą; tam, gdzie było potrzeba - wprowadzono nawiasy.

- g) Jeśli wynik działania przekracza zakres rejestru, w którym ma być umieszczony, wówczas najstarsze bity wyniku są obcinane. Powsta-



jące przy wykonywaniu działań przeniesienia są oznaczone apostrofem, np.  $A'=1$  jeśli przeznaczony do zapisania w akumulatorze wynik przekroczył zakres akumulatora,  $A'=0$  w przeciwnym przypadku; podobnie, jeśli przeznaczony do zapisania w mnożniku wynik przekracza zakres mnożnika, to  $m'=1$  jeśli  $m>0$  lub  $m=+0$ ,  $m'=-1$  jeśli  $m<0$  lub  $m=-0$ ; jeśli natomiast wynik nie przekracza zakresu mnożnika, to  $m'=0$ .

- h) W opisie podano tylko te rejestry lub ich części, których zawartość zmienia się w trakcie wykonywania rozkazu. Korzystając z opisu należy pamiętać, że zawartość rejestrów może być albo taka, jaka była przed wykonaniem rozkazu, albo mogła zostać zmieniona zgodnie z wcześniej podanymi wyrażeniami; np.  $A:=B$ ,  $M:=A$  oznacza: zawartość rejestru B przenieś do rejestru A, a potem nową zawartość rejestru A przenieś do rejestru M.
- i) Przez  $x$  oznaczono efektywny adres rozkazu lub jego parametr (patrz p. 3.1.1).
- j) Dla zaznaczenia znaku liczby, tj. bitu na zerowej pozycji pisano  $+0$  lub  $-0$ ; gdy znak ten jest nieistotny, wówczas pisano  $0$ .
- k) W opisie rozkazów podano następujące informacje:
- symbol rozkazu i jego kod dziesiętny,
  - nazwę,
  - opis działania rozkazu.

W przypadku rozkazów typu ZZR, kod rozkazu podano w postaci dwóch liczb oddzielonych kreską ukośną: kodu ZZR (48) i rozszerzenia, np. 48/55.

Opis rozkazów podano w kolejności alfabetycznej symboli. Spis symboli w kolejności ich kodów podany jest w tablicach 7.10.1, 7.10.2 i 7.10.3.

- 1) W większości przypadków opis działania jest sformalizowany, choć w niektórych przypadkach podano opis słowny. Ponadto czasami podano w nawiasach komentarz lub kilka słów zwracających uwagę Czytelnika na określone fakty.

Podany niżej przykład wraz z komentarzem wyjaśnia sposób interpretowania przyjętego formalizmu. Przykład:

a) jeśli A , lub B, to

C,

D;

jeśli E, to F;

jeśli G, to H.

b) jeśli J, to

K,

L,

M;

jeśli N, to P.

Komentarz: warunki (A lub B) oraz J są rozłączne i uzupełniają się. Jeśli zachodzi np. A lub B, to wykonaj C i D. Ponadto jeśli zachodzi warunek E, to wykonaj F oraz jeśli zachodzi warunek G, to wykonaj H (warunki E i G nie muszą być rozłączne). Podobnie w przypadku b (tzn. jeśli zachodzi warunek J), wówczas wykonaj K,L i M, a jeśli zachodzi ponadto N, to wykonaj P.

## 3.4.2. Rozkazy legalne

<u>Symbol</u> <u>i kod</u>	<u>Nazwa</u> <u>rozkazu</u>	<u>Opis</u> <u>działań</u>
CKA 48/15	Czytaj klucze do akumulatora	Wpisz do akumulatora zawartość rejestru kluczy na stoliku operatora, $LR:=LR+1$ .
DOA 38	Dodaj do akumulatora	$a:=a+pao(x)$ , $N:=N \vee A'$ , $LR:=LR+1$ . (Znak zera - patrz 3.5).
DOB 54	Dodaj do B-rejestru	$b:=b+pao(x)$ , $N:=N \vee B'$ , $LR:=LR+1$ . (Znak zera - patrz 3.5).
DOL 44	Dodaj logicznie	$A:=A \vee PAO(x)$ , $LR:=LR+1$ .
DOM 40	Dodaj do mnożnika	$m:=m+pao(x)$ , $a:=a+m'$ , $N:=N \vee A'$ , $LR:=LR+1$ . ( $m'$ - przeniesienie ze znakiem; znak zera - patrz 3.5).
DOP 4	Dodaj do pamięci	$ pao(x)  :=  pao(x)  + 1$ , $N:=N \vee PAO'(x)$ , $LR:=LR+1$ . ( $PAO_0(x)$ nie zmienia się).

- DOZ 12 Dodaj w zmiennym przecinku Rozkaz jest określony jeśli  $am_z$  i  $pao_z(x)$  są znormalizowane lub  $pao_z(x)$  jest znormalizowane i różne od zera;  
 $am_z := am_z + pao_z(x)$ ,  
 $am_z$  - znormalizowane;  
 jeśli  $|am_w| > 255$  to  
 $LR := -4$ ;  
 jeśli  $|am_w| \leq 255$  to  
 $LR := LR + 1$ .
- DZD 43 Dziel akumulator i mnożnik a) Jeśli  $|pao(x)| > |a|$ , to  $ROB := AM$ ,  
 $m := E(am : pao(x))$ ,  
 $M_o := A_o \div PAO_o(x)$ ,  
 $a := rob - m \cdot pao(x)$ ,  
 $A_o := ROB_o$ ,  
 $LR := LR + 2$ .  
 b) Jeśli  $|pao(x)| \leq |a|$ , to  $LR := LR + 1$ .
- DZZ 15 Dziel w zmiennym przecinku a) Jeśli  $am_z$  i  $pao_z(x)$  są znormalizowane i  $pao_z(x) \neq 0$ , to  
 $am_z := am_z : pao_z(x)$ ,  
 $am_z$  - znormalizowane;  
 jeśli  $|am_w| > 255$  to  
 $LR := -4$ ;  
 jeśli  $|am_w| \leq 255$  to  
 $LR := LR + 1$ .  
 b) Jeśli  $am_z$  nie jest znormalizowane,  $pao_z(x)$  jest znormalizowane i  $pao_z(x) \neq 0$ , to  
 $am_z := 0$ ,  
 $am_z$  - znormalizowane,  
 $LR := LR + 1$ .

- c) Jeśli  $pao_z(x)$  nie jest znormalizowane, lub  $pao_z(x)=0$ , to  $LR:=5$ .
- LAR** Przesuń akumulator i mnożnik w lewo arytmetycznie  
 48/21  $am:=am \cdot 2^x$ ,  
 $N:=N \vee A'$ ,  
 $A_0:=M_0$ ,  
 $LR:=LR+1$ .  
 $(x \leq 63)$ .
- LCA** Przesuń akumulator w lewo cyklicznie  
 48/25  $LR:=LR+1$ , wykonaj  $x$  razy operacje:  
 $ROB_0:=A_0$ ,  
 $A_{0-22}:=A_{1-23}$ ,  
 $A_{23}:=ROB_0$ .  
 $(x \leq 63)$ .
- LCD** Przesuń akumulator i mnożnik w lewo cyklicznie  
 48/17  $LR:=LR+1$ , wykonaj  $x$  razy operacje:  
 $ROB_0:=A_0$ ,  
 $A_{0-22}:=A_{1-23}$ ,  $A_{23}:=M_0$ ,  
 $M_{0-22}:=M_{1-23}$ ,  $M_{23}:=ROB_0$ .  
 $(x \leq 63)$ .
- LCM** Przesuń mnożnik w lewo cyklicznie  
 48/29  $LR:=LR+1$ , wykonaj  $x$  razy operacje:  
 $ROB_0:=M_0$ ,  
 $M_{0-22}:=M_{1-23}$ ,  $M_{23}:=ROB_0$ .  
 $(x \leq 63)$ .
- LMB** Mnóż logicznie mnożnik i przesłij do B-rejestru  
 48/7  $B:=M \wedge x$ ,  
 $LR:=LR+1$ .  
 $(x \leq 255)$ .
- MNL** Mnóż logicznie  
 46  $A:=A \wedge PAO(x)$ ,  
 $LR:=LR+1$ .

MNM 42	Mnóż mnożnik	$am := m \cdot pao(x),$ $A_0 := M_0 \div PAO_0(x),$ $M_0 := A_0,$ $LR := LR + 1.$
MNZ 14	Mnóż w zmien- nym przecinku	<p>a) Jeśli <math>am_z</math> i <math>pao_z(x)</math> są znormalizowane to</p> $am_z := am_z \cdot pao_z(x);$ jeśli $ am_w  > 255,$ to $LR := 4;$ jeśli $ am_w  \leq 255,$ to $LR := LR + 1.$ <p>b) Jeśli <math>am_z</math> lub <math>pao_z(x)</math> nie jest znormalizowane, to <math>am_z := 0,</math>  <math>am_z</math> - znormalizowane,  <math>LR := LR + 1.</math></p>
NNR 48/63 i inne	Nic nie rób	$LR := LR + 1.$
NOR 48/49	Normalizuj	<p>a) Jeśli <math>a = m = 0,</math> to  <math>A_0 := M_0,</math>  <math>LR := LR + 1.</math></p> <p>b) Jeśli <math>a \neq 0</math> lub <math>m \neq 0,</math> to znajdź liczbę rob, taką, że  <math>2^{rob} \leq  am  &lt; 2^{rob+1},</math>            następnie oblicz  <math>am := am \cdot 2^{46-rob},</math>  <math>A_0 := M_0,</math>  <math>b := b + (45 - rob),</math>  <math>N := N \vee B',</math>  <math>LR := LR + 2;</math>            jeśli <math>b = 0,</math> to <math>B_0 := 1.</math></p>
NZB 48/11	Neguj znak B-rejestru	$B_0 := 1 - B_0,$ $LR := LR + 1.$

OAM 48/53	Zaokrąglj w akumulatorze i mnożniku	$M_{23} := 1,$ $LR := LR + 1;$ jeśli $M_1 = 1,$ to $A_{1-23} := A_{1-23} + 1,$ $N := N \vee A'.$
ODA 39	Odejmij od akumulatora	$a := a - pao(x),$ $N := N \vee A',$ $LR := LR + 1.$ (Znak zera - patrz 3.5).
ODB 55	Odejmij od B-rejestru	$b := b - pao(x),$ $N := N \vee B',$ $LR := LR + 1.$ (Znak zera - patrz 3.5).
ODM 41	Odejmij od mnożnika	$m := m - pao(x),$ $a := a - m',$ $N := N \vee A',$ $LR := LR + 1.$ ( $m'$ - przeniesienie ze znakiem; znak zera - patrz 3.5).
ODP 5	Odejmij od pamięci	a) Jeśli $pao(x) = 0,$ to $LR := LR + 1.$ b) Jeśli $pao(x) \neq 0,$ to $ pao(x)  :=  pao(x)  - 1,$ $PAO_0(x) := PAO_0(x),$ $LR := LR + 2.$
ODZ 13	Odejmij w zmiennym przecinku	Rozkaz jest określony jeśli $am_z$ i $pao_z(x)$ są znormalizowane, lub jeśli $pao_z(x)$ jest znorma- lizowane i $pao_z(x) \neq 0;$ $am_z := am_z - pao_z(x),$ $am_z$ - znormalizowane;

- jeśli  $|am_w| > 255$  to  
 $LR := 4$ ;
- jeśli  $|am_w| \leq 255$  to  
 $LR := LR + 1$ .
- OKZ 48/51 Zaokrąglj w zmiennym przecinku
- $M_{1-16} := M_{1-16} + 1$ ,  
 $A_{1-23} := A_{1-23} + m'$ ;  
 $LR := LR + 1$ ;
- jeśli  $a = 0$ , to  $A_0 := M_0$ ;  
 jeśli  $A' = 1$ , to  $A_1 := 1$ ,  
 $A_{2-23} := 0, M_{1-15} := 0$ ,  
 $b := b + 1$  (w tym ostatnim przypadku jeśli  $b = 0$ , to  $B_0 := 0$ ).
- Uwaga: ewentualny nadmiar  $b$  nie jest sygnalizowany.
- OSL 45 Odejmij symetrycznie (logicznie)
- $A := A \div PAO(x)$ ,  
 $LR := LR + 1$ .
- PAD 34 Pamiętaj akumulator i mnożnik długo
- $PAO(x) := A$ ,  
 $PAO(x+1) := M$ ,  
 $LR := LR + 1$ .
- (Modyfikacja B-rejestrem - podwójna).
- PAM 33 Pamiętaj akumulator i mnożnik
- $PAO(x) := A$ ,  
 $PAO(x+1) := M$ ,  
 $LR := LR + 1$ .
- PAR 48/23 Przesuń akumulator i mnożnik w prawo arytmetycznie
- a) Jeśli  $46 \leq x \leq 63$ , to  
 $a := +0$ ,  
 $m := +0$ ,  
 $LR := LR + 1$ .
- b) Jeśli  $x < 46$ , to  
 $am := am : 2^x$ ,  
 $M_0 := A_0$ ,  
 $LR := LR + 1$ .



PCA 48/27	Przesuń akumulator w prawo cyklicznie	$LR:=LR+1$ , wykonaj $x$ razy operacje: $ROB_{23}:=A_{23}$ , $A_{1-23}:=A_{0-22}$ , $A_0:=ROB_{23}$ . $(x \leq 63)$ .
PCD 48/19	Przesuń akumulator i mnożnik w prawo cyklicznie	$LR:=LR+1$ , wykonaj $x$ razy operacje: $ROB_{23}:=M_{23}$ , $M_{1-23}:=M_{0-22}$ , $M_0:=A_{23}$ , $A_{1-23}:=A_{0-22}$ , $A_0:=ROB_{23}$ . $(x \leq 63)$ .
PCM 48/31	Przesuń mnożnik w prawo cyklicznie	$LR:=LR+1$ , wykonaj $x$ razy operacje: $ROB_{23}:=M_{23}$ , $M_{1-23}:=M_{0-22}$ , $M_0:=ROB_{23}$ . $(x \leq 63)$ .
PEB 48/1	Przeskocz przy większym od B-rejestru	a) Jeśli $ b  > x$ to $LR:=LR+1$ . b) Jeśli $ b  \leq x$ to $LR:=LR+2$ . $(x \leq 255)$ .
PMB 51	Pamiętaj B-rejestr	$PAO(x) := B$ , $LR:=LR+1$ .
PMM 49	Pamiętaj mnożnik	$PAO(x) := M$ , $LR:=LR+1$ .

- POB 53 Porównaj B-rejestr
- a) Jeśli  $b < \text{pao}(x)$  lub  $b = -0$  i  $\text{pao}(x) = +0$ , to  $\text{LR} := \text{LR} + 1$ .
- b) Jeśli  $b > \text{pao}(x)$  lub  $b = +0$  i  $\text{pao}(x) = -0$ , to  $\text{LR} := \text{LR} + 2$ .
- c) Jeśli  $b = \text{pao}(x) \neq 0$  lub  $b = \text{pao}(x) = -0$  lub  $b = \text{pao}(x) = +0$ , to  $\text{LR} := \text{LR} + 3$ .
- (b i pao, a nie B i PAO).
- POL 47 Porównaj logicznie
- a) Jeśli  $A < \text{PAO}(x)$ , to  $\text{LR} := \text{LR} + 1$ .
- b) Jeśli  $A > \text{PAO}(x)$ , to  $\text{LR} := \text{LR} + 2$ .
- c) Jeśli  $A = \text{PAO}(x)$ , to  $\text{LR} := \text{LR} + 3$ .
- (A i PAO a nie a i pao)
- PPB 48/3 Przeskocz przy plusie B-rejestru
- a) Jeśli  $B_0 = 0$ , to  $\text{LR} := \text{LR} + 2$ .
- b) Jeśli  $B_0 = 1$ , to  $\text{LR} := \text{LR} + 1$ .
- PPM 48/35 Przeskocz przy plusie mnożnika
- a) Jeśli  $M_0 = 0$ , to  $\text{LR} := \text{LR} + 2$ .
- b) Jeśli  $M_0 = 1$ , to  $\text{LR} := \text{LR} + 1$ .
- PRB 48/45 Przeskocz przy różnym B-rejestrze
- a) Jeśli  $B = x$ , to  $\text{LR} := \text{LR} + 1$ .
- b) Jeśli  $B \neq x$ , to  $\text{LR} := \text{LR} + 2$ .
- (B a nie b,  $x \leq 255$ ).
- PRM 48/37 Przeskocz przy różnym od zera mnożniku
- a) Jeśli  $m = 0$ , to  $\text{LR} := \text{LR} + 1$ .
- b) Jeśli  $m \neq 0$ , to  $\text{LR} := \text{LR} + 2$ .

PZA 32	Pamiętaj i zeruj akumulator	PAO(x) := A, A := 0, LR := LR + 1.
PZN 48/5	Przeskocz przy zerze nadmiaru	a) Jeśli N=0, to LR := LR + 2. b) Jeśli N=1, to N := 0, LR := LR + 1.
Pkl 6 < k1 < 11 lub 16 < k1 < 31	Wykonaj rozkaz programowany k1	a) Jeśli w PAO(x) jest rozkaz programowany P <sub>rs</sub> , to LR := LR + 1. b) Jeśli nie zachodzi przypadek a, to nie zmieniając LR pobierz i wykonaj rozkaz z PAO(x); Uwaga: jeśli w PAO(x) jest rozkaz DOZ, ODZ, MNZ lub DZZ, to LR := 4 lub LR := 5 nie zostanie wykonane, nawet jeśli by to wynikało z działania rozkazu DOZ, ODZ, MNZ lub DZZ.
SDY 1	Skocz do dyrygenta	Rozkaz służący do przekazania sterowania do systemu operacyjnego. Część adresowa interpretowana jest przez system operacyjny. Rozkaz szczególnie opisano w p. 3.4.4.
SKO 0	Skocz	LR := x.
SLR 58	Skocz pamiętając ULR	PAO(1) := ULR, LR := x.

SMA 60	Skocz przy minusie akumulatora	a) Jeśli $A_0=1$ , to LR:=x. b) Jeśli $A_0=0$ , to LR:=LR+1.
SOB 57	Skocz odejmując od B-rejestru	a) Jeśli $b=0$ , to LR:=LR+1. b) Jeśli $b \neq 0$ , to B:=B-1, LR:=x. (B a nie b).
SSE 59	Skocz do segmentu	LR:=x+B <sub>6-15</sub> , B <sub>6-15</sub> :=0 (B <sub>6-15</sub> ≤ .1023).
SSL 2	Skocz ze śladem	PAO(x) :=ULR, LR:=x+1.
STO 48/33	Stop	Rozkaz nielegalny, ale interpretowany przez system operacyjny jako zakończenie działania programu z przyczyną x (x obejmuje 9 ostatnich bitów); nie może mieć żadnej modyfikacji.
SUB 56	Skocz i umieść w B-rejestrze	B:=PAO(2), LR:=x.
SZA 61	Skocz przy zerze akumulatora	a) Jeśli $a=0$ , to LR:=x. b) Jeśli $a \neq 0$ , to LR:=LR+1.
UAD 35	Umieść w akumulatorze i mnożniku długo	A:=PAO(x), M:=PAO(x+1), LR:=LR+1. (Modyfikacja B-rejestrem - podwójna).

UAM 36	Umieść w akumulatorze i mnożniku	$A:=PAO(x),$ $M:=PAO(x+1),$ $LR:=LR+1.$
UEB. 62	Umieść adres efektywny	$B:=x,$ $LR:=LR+1.$
UMA 37	Umieść w akumulatorze	$A:=PAO(x),$ $LR:=LR+1.$
UMB 52	Umieść w B-rejestrze	$B:=PAO(x),$ $LR:=LR+1.$
UMM 50	Umieść w mnożniku	$M:=PAO(x),$ $LR:=LR+1.$
WBM 48/39	Wykładnik z B-rejestru do mnożnika	$M_0:=B_0,$ $M_{16-23}:=B_{16-23},$ $LR:=LR+1.$
WMB 48/41	Wykładnik z mnożnika do B-rejestru	$B_0:=M_0,$ $B_{1-15}:=0, B_{16-23}:=M_{16-23},$ $M_0:=A_0,$ $M_{16-23}:=0,$ $LR:=LR+1.$
WRO 3	Wróć	$N:=N \vee PAO_5(x),$ $LR:=PAO_{9-23}(x) + 1.$
ZAM 48/59	Zamień akumulator z mnożnikiem	$ROB:=A,$ $A:=M,$ $M:=ROB,$ $LR:=LR+1.$
ZBA 48/57	Zamień B-rejestr z akumulatorem	$ROB:=B, B:=A,$ $A:=ROB,$ $LR:=LR+1.$
ZMB 48/43	Zamień mnożnik z B-rejestrem	$ROB:=M,$ $M:=B,$ $B:=ROB,$ $LR:=LR+1.$

ZZN 48/55	Zrównaj zna- ki	$am := a \cdot 2^{23+m},$ $LR := LR + 1;$ jeśli $am = 0$ , to $A_0 := 0,$ $M_0 := 0.$
ZZR 48	Zmiana zawar- tości rejestrów	Grupa rozkazów bezadre- sowych lub posiadają- cych 8-bitowy parametr; wszystkie rozkazy z tej grupy mają swoje symbo- le oraz nazwy; zostały one opisane w tym pun- kcie w odpowiednich dla nich miejscach.

### 3.4.3. Rozkazy nielegalne

Rozkazy nielegalne charakteryzują się tym, że sposób ich działania zależy od stanu wskaźnika legalności F. Przy zgaszonym wskaźniku legalności ( $F=0$ ) działanie wszystkich rozkazów nielegalnych jest identyczne i zostało opisane w punkcie 1.5. Poniższy opis dotyczy więc sytuacji, gdy wskaźnik legalności jest zapalony ( $F=1$ ). W opisie poza symbolami określonymi w punkcie 3.4.1 użyto następujących oznaczeń:

D oznacza 11-bitowy rejestr blokady dolnej,  
 G oznacza 11-bitowy rejestr blokady górnej,  
 L oznacza 8-bitowy rejestr lampek.

<u>Symbol</u>	<u>Nazwa</u>	<u>Opis działania</u>
<u>i kod</u>	<u>rozkazu</u>	
CBP 63/7	Czytaj równo- legle blok słów	Operacja OWW (kanał).
CRA 63/3	Czytaj równo- legle do aku- mulatora	Operacja OWW (arytmo- metr).

CRP 63/3	Czytaj równoległe do pamięci	Operacja OWW (kanał).
CTA 63/5	Czytaj trzy znaki do akumulatora	Operacja OWW (arytmometr).
CTB 63/5	Czytaj po trzy znaki blok słów	Operacja OWW (kanał).
NAS 63/0	Nadaj sygnał	Operacja OWW (kanał i arytmometr).
NNR 63/6 63/7	Nic nie rób	Operacja OWW (arytmometr).
OWW 63	Operacje wejścia-wyjścia	Grupa 16 rozkazów służących do współpracy z urządzeniami zewnętrznymi przez kanał lub arytmometr.
PBP 63/6	Pisz równoległe blok słów	Operacja OWW (kanał).
PDG 48/9	Pisz w rejestrach D i G	$G:=A_{6-16}$ , $D:=M_{6-16}$ , $LR:=LR+1$ .
PLA 48/13	Pisz lampki z akumulatora	$L:=A_{16-23}$ , $LR:=LR+1$ .
PRA 63/2	Pisz równoległe z akumulatora	Operacja OWW (arytmometr).
PRP 63/2	Pisz równoległe z pamięci	Operacja OWW (kanał).

PTA 63/4	Pisz trzy znaki z akumulatora	Operacja OWW (arytmometr).
PTB 63/4	Pisz po trzy znaki blok słów	Operacja OWW (kanał).
PZS 63/1	Przeskocz przy zerze sygnału	Operacja OWW (kanał i arytmometr).
STO 48/33	Stop	LR:=LR+1, zatrzymaj działanie maszyny. Przy F=0 rozkaz ten, traktowany jako nielegalny, powoduje przerwanie pracy programu i przejście do działania systemu operacyjnego; system interpretuje to jako zakończenie programu z przyczyną x, gdzie x stanowi 9-bitowy parametr rozkazu STO. ( $x \leq 511$ ).

#### 3.4.4. Rozkazy typu SDY

Działanie rozkazu SDY jest następujące:

<u>Symbol</u> <u>i kod</u>	<u>Nazwa</u> <u>rozkazu</u>	<u>Opis działania</u>
SDY 1	Skocz do dyrygenta	PAO (0) :=ULR, F:=1, LR:=41.

Należy zwrócić uwagę, że w powyższym opisie zapis PAO (0) oznacza zerowe słowo pamięci przez-



naczonej dla programu użytkownika, natomiast LR:=41 oznacza skok do 41 słowa całej pamięci, ponieważ w tym momencie F=1 i adresy nie są automatycznie zwiększane o zawartość rejestru blokady dolnej (por. p. 1.6). Część adresowa rozkazu (parametr) nie ma wpływu na sposób wykonania rozkazu. Dopiero system operacyjny pobiera ten parametr i wykonuje czynności określone tym parametrem.

Parametry rozkazu SDY mogą formalnie przyjmować wartości od 0 do 32767 i można je podzielić na 4 grupy:

- parametry, które można stosować w rozkazach SDY we wszystkich programach, m.in. w programach użytkowych (tzw. SDY legalne - L);
- parametry, których nie można stosować w programach użytkowych, ale można stosować w innych programach, m.in. w translatorach (tzw. SDY T-legalne - T);
- parametry, które można stosować tylko w systemie operacyjnym (tzw. SDY nielegalne - N);
- inne parametry, które nie określają żadnej czynności do wykonania przez system operacyjny.

Rozkazy SDY legalne są opisane w punkcie 3.4.5. Poniższe zestawienie obejmuje wszystkie rozkazy SDY legalne, T-legalne i nielegalne; dla rozkazów SDY legalne - wskazano, gdzie znajduje się dokładniejszy opis.

<u>Parametr</u>	<u>Legalność</u>	<u>Nazwa rozkazu</u>	<u>Opisany w</u>
3/10+ W/17+np	L	Czekaj i sprawdź warunek	3.4.5.a
2/10+ W/17+np	L	Sprawdź warunek	3.4.5.a
1/10+uz/17+np	L	Zdefiniuj przesłanie dla uz	3.4.5.a
1/10+ 1/11+us/17+np	L	Zdefiniuj przesłanie dla us	3.4.5.a
1/11+np/17+7/20+4	L	Rozdefiniuj przesłanie	3.4.5.a
1/11+ 0/17+7/20+3	L	Zawieś maszynę	3.4.5.h
1/11+ 0/17+7/20+2	L	Pisz na monitorze z M-2	3.4.5.e
1/11+uz/17+7/20+1	L	Wymień program	3.4.5.f
1/11+us/17+7/20+0	N	Wymień program z przeadresowaniem	
1/11+uz/17+3/20+7	L	Zwolnij uz	3.4.5.b
1/11+tm/17+3/20+6	L	Zamknij TM	3.4.5.c
1/11+tm/17+3/20+5	L	Zamknij pisanie na TM	3.4.5.c
1/11+tm/17+3/20+3	L	Otwórz TM	3.4.5.c
1/11+tm/17+3/20+2	L	Sprawdź otwarcie TM	3.4.5.c
1/11+tm/17+3/20+1	L	Podaj BTM do B	3.4.5.c
1/11+tm/17+3/20+0	L	Ustaw TM na bloku B	3.4.5.c
1/11+uz/17+2/20+5	N	Zdefiniuj absolutnie przesłanie O dla urządzenia	
1/11+tm/17+2/20+4	T	Wpisz otwarcie urządzenia	
1/11+tm/17+2/20+2	T	Odwiń absolutnie TM	
1/11+tm/17+2/20+0	T	Załaduj BTM wg B	
1/11+np/17+1/20+5	L	Podaj definicję przesłania	3.4.5.a
1/11+tm/17+1/20+4	L	Sprowadź masyw z TM	3.4.5.c
1/11+tm/17+1/20+3	L	Zapisz masyw na TM	3.4.5.c
1/11+ 0/17+1/20+2	L	Odłącz wszystkie SWW	3.4.5.d
1/11+sw/17+1/20+1	L	Odłącz SWW	3.4.5.d
1/11+sw/17+1/20+0	L	Dołącz SWW	3.4.5.d

1/11+tm/17+0/20+7	L	Zbadaj blokadę zapisu TM	3.4.5.o
1/11+tm/17+0/20+6	L	Zbadaj niegotowość TM	3.4.5.o
1/11+tm/17+0/20+5	L	Zbadaj zajętość TM	3.4.5.o
1/11+tm/17+0/20+4	L	Czekaj na wolną TM	3.4.5.o
1/11+tm/17+0/20+3	L	Kasuj TM	3.4.5.o
1/11+tm/17+0/20+2	L	Odwiń TM	3.4.5.o
1/11+tm/17+0/20+1	L	Posuń TM	3.4.5.o
1/11+tm/17+0/20+0	L	Cofnij TM	3.4.5.o
1/12+us/17+3/20+1	N	Podaj BTM TMS do B	
1/12+us/17+3/20+0	N	Ustaw TMS na bloku B	
1/12+us/17+2/20+5	N	Zdefiniuj absolutnie przesłanie O dla us	
1/12+us/17+2/20+2	N	Odwiń absolutnie us	
1/12+us/17+2/20+0	N	Załaduj BTM us wg B	
1/12+us/17+0/20+2	N	Odwiń us	
1/12+us/17+0/20+1	N	Posuń us	
1/12+us/17+0/20+0	N	Cofnij us	
1/17+5/20+6	L	Nadaj wartość T-etykiecie	3.4.5.h
1/17+5/20+5	L	Podaj T-etykietę do A	3.4.5.h
1/17+5/20+4	L	Podaj datę kalendarzową do A	3.4.5.g
1/17+5/20+3	L	Podaj godzinę do A	3.4.5.g
1/17+5/20+2	L	Podaj zegar do AM	3.4.5.g
1/17+5/20+1	L	Czytaj monitor	3.4.5.e
1/17+5/20+0	L	Pisz na monitorze	3.4.5.e
1/17+4/20+1	N	Zwolnij bibliotekę	
1/17+4/20+0	T	Pisz metrykę do EM	
1/17+1/20+7	T	Zapal legalność	
1/17+1/20+6	T	Ustaw na podzbiorze	
1/17+1/20+5	T	Ustaw na zbiorze	
1/17+1/20+4	T	Przejdź do podzbioru	
1/17+1/20+3	L	Przejdź do zbioru	3.4.5.f
1/17+1/20+2	T	Podaj podzbiór	
1/17+1/20+1	L	Podaj zbiór	3.4.5.f
1/17+1/20+0	T	Podaj metrykę do PAO	

## 1/17+0/20+7 L Przeadresuj

1/17+0/20+7	L	Przeadresuj	3.4.5.
1/17+0/20+6	L	Przejdź do podzbioru legalnie	3.4.5.
1/17+0/20+5	L	Podaj do B adres nazwy z A	3.4.5.
1/17+0/20+4	L	Podaj podzbiór legalnie	3.4.5.
1/17+0/20+3	L	Wybierz z katalogu	3.4.5.
1/17+0/20+2	L	Podaj datę do A	3.4.5.
1/17+0/20+1	L	Podaj opis urządzenia	3.4.5.
1/17+0/20+0	L	Podaj uz do B	3.4.5.
np	L	Inicjuj przesłanie	3.4.5.

### 3.4.5. SDY legalne

Podany poniżej skrócony opis rozkazów składa się z dwóch części: postaci rozkazu wraz z parametrami i graficznie zaznaczonym miejscem powrotu oraz z opisu danych dla rozkazu i opisu działania rozkazu. Pełny opis rozkazów znajduje się w "ZAM 41 Oprogramowanie (System 141), cz. I - Opis maszyny użytkowej". Numery uz i us podane są w punkcie 6.1. We wszystkich rozkazach apao oznacza adres w pamięci operacyjnej, abeb - adres bębnowy.

#### a) Przesłania

Przez ZX(np) oznaczono rejestr warunków przesłania np; zawartość tego rejestru jest sumą liczb, które odpowiadają zaistniałym faktom, związanym z przesłaniem np, zgodnie z poniższą tabelką:

- 2 - zbliżający się koniec TM,
- 4 - przesłanie błędne,
- 8 - przesłanie puste,
- 16 - przesłanie niepełne,
- 32 - nadmiar,
- 64 - zakończono przesłanie.

#### Zdefiniuj przesłanie dla uz

SDY 1/10+uz/17+np	<u>Działanie:</u>
apao	zdefiniuj przesłanie n
n=ilość słów	słów do lub z urządzenia
→ powrót	uz,
	LR:=LR+3.

#### Zdefiniuj przesłanie dla us

SDY 1/10+1/11+us/17+np	<u>Działanie:</u>
apao	zdefiniuj przesłanie n
n=ilość słów	słów do lub z bębna, ja-
→ powrót	ko us,
	LR:=LR+3.

Inicjuj przesłanie

dla bębna:

SDY np

1/3+abeb

→ powrót

dla pozostałych

urządzeń:

SDY np

→ powrót

Dane:

wykonane SDY zdefiniuj przesłanie dla uz lub us.

Działanie:

po zakończeniu poprzedniego przesłania o numerze np i po zwolnieniu

urządzenia (jeśli urządzenie nie jest czytnikiem kart):  $ZX(np)=0$ ,

zainicjuj przesłanie o numerze np; po zakończeniu przesłania ustaw

 $ZX(np)$ . Dla urządzeń us

dozwolony tylko bęben

użytkownika. Dla TM:

a) jeśli wystąpił koniec taśmy, to  $BTM=BTM$ ;b) jeśli taśma odłączona, to  $BTM=-0$ ;c) w pozostałych przypadkach  $BTM=BTM+1$ .

Dla bębna:

 $LR=LR+2$ .

Dla urządzeń poza bębna:

 $LR=LR+1$ .Sprawdź warunek

SDY 2/10+W/17+np

→ warunek spełniony

→ warunek niespełniony

Dane:

W jest sumą liczb odpowiadających badanym warunkom:

2 - zbliżający się koniec TM

4 - przesłanie błędne,

8 - przesłanie puste,

- 16 - przesłanie niepełne,  
 32 - nadmiar,  
 64 - zakończenie przesłania.

Działanie:

- a) Jeśli  $ZX(np) \wedge W=0$ , to  
 $LR:=LR+2$ .  
 b) W przeciwnym przypadku  
 $LR:=LR+1$ .

Czekaj i sprawdź warunek

SDY 3/10+W/17+np

Dane:

- warunek spełniony jak w opisie rozkazu  
 → warunek niespełniony "sprawdź warunek".

Działanie:

- a) Jeśli przesłanie np nie jest zdefiniowane, to  $LR:=LR+2$ .  
 b) Inaczej czekaj na zakończenie przesłania, a następnie sprawdź warunek.

Podaj definicję przesłania

SDY 1/11+np/17+1/20+5

Działanie:

apao

→ powrót

- a) Jeśli przesłanie np nie jest zdefiniowane, to  
 $PAO(apao):=+0$ ,  
 $LR:=LR+2$ .  
 b) Jeśli przesłanie jest zdefiniowane, to  
 $PAO(apao):=ZX(np) \cdot 2^{17} + 1$ ,  
 $PAO(apao+1):=uz$  lub  $64+us$ ,  
 $PAO(apao+2):=adres$  z definicji przesłania np,  
 $PAO(apao+3):=ilość$  z definicji przesłania np,  
 $LR:=LR+2$ .

## Rozdefiniuj przesłanie

SDY 1/11+np/17+7/20+4 Działanie:

→ powrót

- a) Jeśli przesłanie jest w toku, to rozkaz ten jest nielegalny.
- b) Inaczej przestaje obowiązywać definicja przesłania np,  
LR:=LR+1.

b) Współpraca z różnymi urządzeniami

Przez symbol urządzenia rozumie się skrót: CP, DP, DW, TM, CK, BB (bęben), natomiast przez nazwę urządzenia: CPA, CPB, DPA itd. (por. 6.1).

Podaj uz do B

SDY 1/17

→ brak urządzenia

→ jest urządzenie

Dane: $A_{0-5}=0,$  $A_{6-23}$ =nazwa urządzenia w KW-6.Działanie:

a) Jeśli w zestawie brak urządzenia, to

B:=0,

LR:=LR+1.

b) Jeśli jest urządzenie, to

B:=uz.2<sup>6</sup>,

LR:=LR+2.

Podaj opis urządzenia

SDY 1/17+1

apao

→ brak urządzenia

→ jest urządzenie

Dane: $A_{0-5}$ =uz.Działanie:

a) Jeśli w zestawie brak urządzenia, to

PAO (apao+7):=nazwa dzia-



łającego superwizora w  
KW-6 adjustowana do pra-  
wej strony,

LR:=LR+2.

b) Jeśli urządzenie jest  
bębnem, to zawartość  
PAO(apao+1),...,PAO  
(apao+5) jest przypadko-  
wa, zawartość pozosta-  
łych słów i powrót -  
tak jak dla innych urzą-  
dzeń.

c) Jeśli w zestawie wys-  
tępuje urządzenie uz i  
nie jest ono bębniem, to:

PAO<sub>8-23</sub>(apao):=symbol  
urządzenia w KW-8,

PAO<sub>16-23</sub>(apao+1):=nr ka-  
nału w KW-8 lub -0 dla  
arytmometru,

PAO<sub>8-23</sub>(apao+2):=nr mo-  
dułu w KW-8,

PAO<sub>0-14</sub>(apao+3):=nr ka-  
nału binarnie,

PAO<sub>15-23</sub>(apao+3):=nr mo-  
dułu binarnie,

PAO(apao+4):=1 dla DW1,  
2 dla DW2, 4 dla DW2E,  
przypadkowa dla innych  
uz,

PAO(apao+5):=liczba  
wierszy na stronie DW  
(inne - przypadkowe),

PAO<sub>6-23</sub>(apao+6):=nazwa  
urządzenia w KW-6,

PAO(apao+7):=nazwa dzia-  
łającego superwizora w

KW-6 adjustowana do prawej strony,  
 PAO (apao+8) := numer problemu (binarnie),  
 LR:=LR+3.

Zwolnij uz

SDY 1/11+uz/17+3/20+7 Działanie:  
 — powrót LR:=LR+1.

c) Współpraca z TM

Jeśli nie powiedziano inaczej, to symbol  $tm$  oznacza dowolny numer taśmy (do pisania lub czytania). Każda taśma jest otwarta do czytania, otwarta do czytania-pisania lub zamknięta; na początku każdego problemu taśmy są otwarte - można więc, ale nie trzeba ich otwierać. Jednak po każdym zamknięciu konieczne jest otwarcie taśmy, jeśli ma ona być ponownie czytana lub pisana. Przy końcu problemu taśmy otwarte i nie zamknięte w tym problemie, są automatycznie zamykane.

Odwiń TM

SDY 1/11+tm/17+2  
 — powrót

Działanie:

a) Jeśli taśmy brak w zestawie, to

BTM:=-0,

LR:=LR+1.

b) Jeśli taśma jest odwinięta (BTM=0), to

LR:=LR+1.

c) Inaczej rozpocznij odwijanie TM,

LR:=LR+1;

po odwinięciu BTM:=0.

Cofnij TM

SDY 1/11+tm/17

→ powrót

Działanie:

- a) Jeśli taśmy brak w zestawie, to  
 $BTM:=-0$ ,  
 $LR:=LR+1$ .
- b) Jeśli taśma odwinęta ( $BTM=0$ ), to  
 $LR:=LR+1$ .
- c) Inaczej rozpocznij cofanie TM o 1 blok,  
 $LR:=LR+1$ ,  
 po cofnięciu  $BTM:=BTM-1$ .

Posuń TM

SDY 1/11+tm/17+1

→ powrót

Działanie:

- a) Jeśli taśmy brak w zestawie, to  
 $BTM:=-0$ ,  
 $LR:=LR+1$ .
- b) Jeśli taśma jest przy końcu, to  
 $LR:=LR+1$ .
- c) Inaczej rozpocznij posuwanie TM o 1 blok,  
 $LR:=LR+1$ ,  
 po posunięciu  $BTM:=BTM+1$ ,  
 ewentualny nadmiar uz jest wypisywany po zakończeniu problemu.

Kasuj TM

SDY 1/11+tm/17+3

→ powrót

Działanie:

- a) Jeśli taśmy brak w zestawie, to  
 $BTM:=-0$ ,  
 $LR:=LR+1$ .

- b) Jeśli taśma jest przy końcu, to  
LR:=LR+1.
- c) Inaczej rozpocznij kasowanie ok. 8 cm taśmy,  
LR:=LR+1.

### Otwórz TM

SDY 1/11+tm/17+3/20+3

wskaźnik dostępu  
apao

- nie otwarto
- otwarto do czytania
- otwarto do czytania-pisania

Dane:

pole 27 słów począwszy od apao, przeznaczono na metryki.

Działanie:

- a) Jeśli taśmy brak w zestawie, to  
B:=2,  
LR:=LR+3.
- b) Jeśli taśma jest w zestawie i jest zablokowana, to  
odwiń taśmę,  
BTM:=0,  
otwórz taśmę do czytania,  
wypisz na monitorze tekst:  
OTW TM M-K VOL 123456,  
przenieś metrykę do PAO  
od słowa apao,  
LR:=LR+4.
- c) Jeśli taśma jest w zestawie i nie jest zablokowana, to  
odwiń taśmę,  
BTM:=0,  
otwórz taśmę do czytania i pisania,  
wypisz na monitorze tekst:  
OTW TM M-K VOL 123456,

przenieś metrykę do PAO  
od słowa apao,

LR:=LR+5.

Dla szpul ogólnodostęp-  
nych - wskaźnik dostę-  
pu = 64.

### Sprawdź otwarcie TM

SDY 1/11+tm/17+3/20+2 Działanie:

- |                     |                         |
|---------------------|-------------------------|
| → nie otwarto       | a) Jeśli taśma nie jest |
| → otwarto do czyta- | otwarta, to             |
| nia                 | LR:=LR+1.               |
| → otwarto do czyta- | b) Jeśli taśma jest     |
| nia-pisania         | otwarta do czytania, to |
|                     | LR:=LR+2.               |
|                     | c) Inaczej              |
|                     | LR:=LR+3.               |

### Zamknij TM

SDY 1/11+tm/17+3/20+6 Dane:

apao tekst w KW-8 zapisan, po 3  
→ powrót znaki w kolejnych słowach  
PAO, począwszy od apao;  
tekst może zawierać co  
najwyżej 6 słów, ostat-  
nie słowo = 0.

Działanie:

- a) Jeśli taśma nie jest  
otwarta, to  
LR:=LR+2.  
b) Inaczej  
odwiń taśmę,  
zamknij taśmę,  
wypisz na monitorze tekst  
ZAM TM M-K VOL 123456 1  
tekst zapisany od apao,  
LR:=LR+2.

Zamknij pisanie na TMSDY 1/11+tm/17+3/20+5 Działanie:

— powrót

zamknij taśmę do pisania,  
o ile była otwarta do pi-  
sania,

LR:=LR+1.

Uwaga: rozkaz można sto-  
sować do innych urządzeń  
niż tm, wówczas jest roz-  
kazem NIC NIE RÓB.Podaj BTM do BSDY 1/11+tm/17+3/20+1 Działanie:

— powrót

a) Jeśli taśmy brak w  
zestawie, to

B:=-0,

LR:=LR+1.

b) Inaczej

B:=BTM,

LR:=LR+1.

Ustaw TM na bloku B

SDY 1/11+tm/17+3/20

— powrót

Dane:

B=nr bloku.

Działanie:a) Jeśli taśmy brak w  
zestawie, to

BTM:=-0,

LR:=LR+1.

b) Jeśli taśma jest w  
zestawie i  $B \leq 0$ , to  
ustaw taśmę na bloku 0,

BTM:=0,

LR:=LR+1.

c) Jeśli  $B \geq BTM$  odpowia-  
dającemu ostatniemu blo-  
kowi taśmy, to

ustaw taśmę na bloku  
ostatnim i odpowiednio  
zmodyfikuj BTM,  
zapisz ewentualny nad-  
miar (wypisywany po za-  
kończeniu problemu),  
LR:=LR+1.

d) Inaczej  
ustaw taśmę na bloku B,  
BTM:=B,  
LR:=LR+1.

### Zapisz masyw na TM

SDY 1/11+tm/17+1/20+3 Dane:

apao

n=liczba słów

→gdy koniec tm

→gdy poprawnie

n słów w PAO począwszy od  
apao, przeznaczone do za-  
pisania na taśmie.

#### Działanie:

podziel obszar n, słów, po-  
czynając od apao na bloki  
po 1024 słowa (ostatni -  
reszta), dopisz metrykę  
zgodnie ze standardem  
(por. 3.6) i zapisz bloki  
te na taśmie. Ponadto  
B:=nr pierwszego z zapi-  
sanych bloków,  
AM:= suma kontrolna zapi-  
sanych informacji;  
jeśli w trakcie pisania  
pojawił się koniec taśmy,  
to

LR:=LR+3;

inaczej

LR:=LR+4.

Uwaga: 1) tm musi być nume-  
rem do pisania; 2) rozkaz  
powoduje rozdefiniowanie  
przesłania O.

Sprowadź masyw z TM

SDY 1/11+tm/17+1/20+4

apao

n=liczba słów

m=nr bloku

→gdy poprawnie

→gdy błąd

Działanie:

czytaj z taśmy n słów  
począwszy od bloku m,  
sprawdź sumy kontrolne  
wg przyjętego standardu  
(por. 3.6);  
jeśli wystąpił błąd w  
odczytanie, złe sumy kon-  
trolne, złe numery blo-  
ków, koniec taśmy itp., to  
LR:=LR+5;

jeśli odczyt poprawny, to  
zapisz informacje z blo-  
ków (bez metryk) do pa-  
mięci poczynając od apao,  
LR:=LR+4.

Uwaga: 1) tm musi być nu-  
merem taśmy do czytania,  
2) n musi obejmować peł-  
ną liczbę kolejnych blo-  
ków, 3) rozkaz powoduje  
rozdefiniowanie przesła-  
nia 0.

Zbadaj blokadę zapisu TM

SDY 1/11+tm/17+7

→zapis zablokowany

→zapis odblokowany

Działanie:

a) Jeśli zapis tm jest  
zablokowany, to  
LR:=LR+1.

b) Inaczej  
LR:=LR+2.

Zbadaj niegotowość TM

SDY 1/11+tm/17+6

→niegotowa

→gotowa

Działanie:

a) Jeśli taśma jest nie-  
gotowa, to  
LR:=LR+1.

b) Inaczej  
LR:=LR+2.



Zbadaj zajętość TM

SDY 1/11+tm/17+5

→ zajęta

→ wolna

Działanie:

a) Jeśli trwa operacja dotycząca taśmy, to  
 $LR:=LR+1$ .

b) Inaczej

 $LR:=LR+2$ .

Uwaga: taśma musi być otwarta.

Czekaj na wolną TM

SDY 1/11+tm/17+4

→ powrót

Działanie:

zawieś program aż do zakończenia trwającej operacji dotyczącej taśmy,

 $LR:=LR+1$ .

Uwaga: taśma musi być otwarta.

d) Współpraca z SWW

We wszystkich rozkazach symbol sw oznacza numer symbolicznego wejścia lub 16+numer symbolicznego wyjścia.

Dołącz SWW

SDY 1/11+sw/17+1/20

→ powrót

Dane:

PAO(38)= adres dołączanego symbolicznego wejścia lub wyjścia.

Działanie:

sprowadź symboliczne wejście lub wyjście do pamięci poczynając od adresu zapisanego w PAO(38),

przygotuj symboliczne wejście lub wyjście do pracy (zaadresuj, przydziel numery przesłań itd.),

$PAO(38) := PAO(38) + \text{długość podprogramu} + \text{długość dekodera},$

$LR := LR + 1.$

Uwaga: jeśli w trakcie dołączania symbolicznego wejścia lub wyjścia zostanie przekroczony adres w PAO zawarty w 57 słowie PAO, wówczas następuje przerwanie pracy programu.

#### Odłącz SWW

SDY 1/11+sw/17+1/20+1

→ powrót

Działanie:

odłącz symboliczne wejście lub wyjście,

$LR := LR + 1.$

#### Odłącz wszystkie SWW

SDY 1/11+1/20+2

→ powrót

Działanie:

odłącz wszystkie dołączone symboliczne wejścia i wyjścia oprócz wejścia i wyjścia o numerze 0,

$LR := LR + 1.$

#### e/ Współpraca z monitorem

##### Pisz na monitorze z M-2

SDY 1/11+7/20+2

→ powrót

Dane:

tekst o długości do 100

znaków, zakończony znakiem "pusty", zapisany po cztery znaki w słowie w kodzie M-2

(1. znak - bity 1-5,  
2. znak - bity 7-11, itd.,  
bity 0,6,12,18 zawierają 0),

B= adres tekstu.

Działanie:

czekaj na zakończenie operacji z monitorem, wypisz na monitorze tekst, którego adres podany jest w B,  
LR:=LR+1.

Pisz na monitorze

SDY 1/17+5/20

→ powrót

Dane:

tekst o długości do 100 znaków, zakończony znakiem NL, zapisany po trzy znaki w słowie w KW-8 (1. znak - bity 0-7, 2. znak - bity 8-15, 3. znak - bity 16-23),

B = adres tekstu.

Działanie:

wypisz na monitorze znak NL, numer priorytetu oraz tekst, którego adres podany jest w B,

LR:=LR+1.

Czytaj monitor

SDY 1/17+5/20+1

→ powrót

Dane:

B = adres pola, do którego należy czytać tekst z monitora.

Działanie:

wypisz na monitorze znak NL i numer priorytetu,

wczytaj z monitora wprowadzane znaki, dekodując je na KW-8 i umieszczając po trzy, w kolejnych słowach, poczynając od adresu podanego w B,

LR:=LR+1.

Uwagi: 1) Wprowadzanie znaku NL powoduje rozpoczęcie wczytywania od początku. Znaki - i \* powodują adjustowanie ostatniego słowa do prawej strony, tak aby - względnie \* znalazły się na bitach 16-23.

Ewentualne uzupełnienie początku tego słowa odbywa się znakami NU. Ponadto znak \* jest dekodowany na NL i kończy wprowadzanie tekstu. Po wypełnieniu 31 słów wprowadzanym tekstem, następne znaki będą pomijane, a ostatnie 2,1

lub 0 znaków i znak NL  
będą wprowadzone do 31  
słowa.

2) Niewprowadzenie teks-  
tu z monitora w ciągu  
około 2 minut, spowoduje  
wprowadzenie do pierwsze-  
go słowa buforu wartości  
84 (znak \* w KW-8) i  
przejęcie do dalszego  
działania programu.

#### f/ Współpraca z biblioteką

Wszystkie nazwy zbiorów i podzbiorów napisane  
w KW-6 są przez system adjustowane do lewej  
strony. Nie ma więc znaczenia sposób adjusto-  
wania nazw w łącznikach do rozkazu SDY.

#### Wymień program

SDY 1/11+uz/17+7/20+1 Działanie:  
apao zdefiniuj przesłanie nr 0  
n=liczba słów na lub z bębna począwszy  
abeb od adresu abeb n słów z  
adres startu lub do pamięci operacyj-  
nej, począwszy od adresu  
apao,  
wykonaj przesłanie 0,  
LR:=adres startu.  
Uwaga: uz musi być nume-  
rem bębna.

#### Przeadresuj

SDY 1/17+7 Działanie:  
apao od każdego słowa ujemne-  
n=liczba słów go leżącego w sekwencji  
→powrót przeadresowywanej odej-  
mij apao,  
LR:=LR+3.  
Uwaga: sekwencją przea-  
dresowywaną są takie cią-

gi słów, które: 1) zaczynają się w apao lub w słowie między apao i  $apao+n-1$  mającym wartość +8388607, 2) kończą się w  $apao+n-1$  lub w słowie pomiędzy apao i  $apao+n-1$  mającym wartość -8388607, 3) nie zawierają słowa o wartości -8388607.

Podaj zbiór

SDY 1/17+1/20+1

nazwa zbioru w KW-6

— brak zbioru

— jest zbiór

Działanie:

a) Jeśli w bibliotece istnieje zbiór zwarty o podanej nazwie, a jego długość nie przekracza obszaru bębnowego, to sprowadź ten zbiór na bęben, a jego sekcję startową do PAO,  
LR:=LR+3.

b) Inaczej

LR:=LR+2

Uwaga: sposób sprowadzenia zbioru określony jest w metryce zbioru.

Przejdź do zbioru

SDY 1/17+1/20+3

nazwa zbioru w KW-6

Działanie:

a) Jeśli w bibliotece istnieje zbiór zwarty o podanej nazwie, a jego długość nie przekracza obszaru bębnowego, to sprowadź ten zbiór na bęben, a jego sekcję star-

tową do PAO, przejdź do wykonania zbioru.

b) Inaczej - rozkaz nielegalny.

Uwaga: sposób sprowadzenia zbioru oraz adres, od którego rozpoczyna się jego wykonanie określone są w metryce zbioru.

Podaj podzbiór legalnie

SDY 1/17+4

nazwa zbioru w KW-6

nazwa podzbioru w  
KW-6

→ brak podzbioru

→ jest podzbiór

Działanie:

a) Jeśli w bibliotece istnieje zbiór złożony o podanej nazwie zawierający podzbiór o podanej nazwie oraz długość tego podzbioru nie przekracza obszaru pamięci, to sprowadź wskazany podzbiór do PAO,

LR:=LR+4.

b) Inaczej

LR:=LR+3.

Uwaga: sposób sprowadzenia podzbioru określony jest w metryce zbioru.

Przejdź do podzbioru legalnie

SDY 1/17+6

nazwa zbioru w KW-6

nazwa podzbioru w  
KW-6

Działanie:

a) Jeśli w bibliotece istnieje zbiór złożony o podanej nazwie, zawierający podzbiór o podanej nazwie i długość tego podzbioru nie przekracza obszaru w pamięci opera-

cyjnej, to  
sprowadź wskazany pod-  
zbiór do PAO, przejdź do  
wykonania podzbioru.

b) Inaczej rozkaz niele-  
galny.

Uwaga: sposób sprowadza-  
nia podzbioru oraz adres,  
od którego rozpoczyna się  
jego wykonywanie, określo-  
ne są w metryce zbioru.

Wybierz z katalogu

SDY 1/17+3

→brak zbioru

→jest zbiór

Dane:

A = nazwa zbioru w KW-6,

B = apao.

Działanie:

a) Jeśli w bibliotece  
istnieje zbiór o nazwie  
podanej w akumulatorze, to  
 $PAO_{0-11}(apao)$  :=wymagana  
minimalna PAO (w jednost-  
kach po 128 słów),

$PAO_{12-23}(apao)$  :=wymagana  
minimalna pamięć bębnowa  
(w jednostkach po 128  
słów),

$PAO(apao+1)$  :=długość me-  
tryki zbioru,

$PAO(apao+2)$  :=adres bi-  
blioteczny metryki zbioru,

LR:=LR+2.

b) Jeśli w bibliotece brak  
wskazanego zbioru, to

LR:=LR+1.



g) Współpraca z kalendarzem i zegaremPodaj datę do A

SDY 1/17+2

→ powrót

Działanie: $A_{0-8} := \text{rok},$  $A_{9-23} := \text{dzień w roku},$  $LR := LR+1.$ Podaj datę kalendarzową do A

SDY 1/17+5/20+4

→ powrót

Działanie: $A_{0-7} := \text{dzień},$  $A_{8-15} := \text{miesiąc},$  $A_{16-23} := \text{rok},$  $LR := LR+1.$ Podaj godzinę do A

SDY 1/17+5/20+3

→ powrót

Działanie:

pobierz z rejestru zegara godziny, minuty, sekundy,

umieść w akumulatorze:

 $A_{0-11} := \text{godziny},$  $A_{12-17} := \text{minuty},$  $A_{18-23} := \text{sekundy},$  $LR := LR+1.$ 

Uwaga: czas realizacji rozkazu - do 40 ms.

Podaj zegar do AM

SDY 1/17+5/20+2

→ powrót

Działanie: $A_{1-23} := RZ_{1-23},$  $A_0 := 0,$  $M_{1-6} := RZ_{23-29},$  $M_0 := 0,$  $M_{7-23} := 0,$  $LR := LR+1.$

Uwagi: 1) przez RZ oznaczono rejestr zegara,  
2) czas realizacji rozkazu - do 40 ms.

#### h) Inne

##### Zawieś maszynę

SDY 1/11+7/20+3

→ powrót

##### Działanie:

zawieś pracę programu do czasu wprowadzenia startu z monitora (tzn. tekstu: nr priorytetu - S\*) po wprowadzeniu tekstu  
LR:=LR+1.

##### Podaj do B adres nazwy z A

SDY 1/17+5

apao

X=szablon

p=krok

n=ilość

→ brak nazwy

→ jest nazwa

##### Działanie:

a) Jeśli  $p \leq 0$ , to

LR:=LR+5.

b) Jeśli  $p > 0$ , to

porównuj A kolejno z iloczynami  $X \wedge PAO(apao)$ ,  $X \wedge PAO(apao+p)$ , ...;

jeśli znajdzie równość to B:=adres słowa w PAO, dla którego równość zachodzi,

LR:=LR+6;

jeśli w czasie działania rozkazu adres w pamięci przekroczy  $apao+n-1$  lub obszar w pamięci dostępny dla programu, to  
LR:=LR+5.

Nadaj wartość T-etykiecie

SDY 1/17+5/20+6

T-etykieta

→ powrót

Działanie:

a) Jeśli T-etykieta występuje na liście T-etykiet lub gdy są wolne miejsca na liście T-etykiet, to

T-etykieta:=A,

LR:=LR+2.

b) Inaczej rozkaz nielegalny.

Podaj T-etykietę do A

SDY 1/17+5/20+5

T-etykieta

→ brak T-etykiety

→ jest T-etykieta

Działanie:

a) Jeśli na liście T-etykiet znajduje się T-etykieta, to

A:=T-etykieta,

LR:=LR+3.

b) Inaczej

A:=liczba T-etykiet na liście T-etykiet,

LR:=LR+2.

Uwaga: rozkaz powoduje rozdefiniowanie przesłania O.

## 3.5. DZIAŁANIA Z WYNIKIEM 0

Jeśli po wykonaniu rozkazów DOB, DOA lub DOM w wyniku otrzymujemy liczbę 0, to znak tej liczby jest równy znakowi liczby pobieranej z pamięci. Jeśli natomiast po wykonaniu rozkazów ODB, ODA lub ODM w wyniku otrzymujemy liczbę 0, to znak tej liczby jest przeciwny znakowi liczby pobranej z pamięci.

W myśl powyższej reguły, dla dowolnej liczby całkowitej  $X > 0$  zachodzą równości:

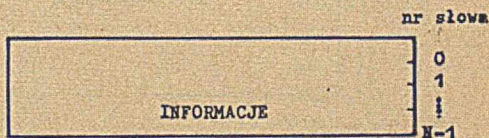
Rejestr	PAO	Wynik
(+X)	+ (-X)	= -0
(-X)	+ (+X)	= +0
(+X)	- (+X)	= -0
(-X)	- (-X)	= +0
(+0)	+ (+0)	= +0
(+0)	+ (-0)	= -0
(+0)	- (+0)	= -0
(+0)	- (-0)	= +0
(-0)	+ (+0)	= +0
(-0)	+ (-0)	= -0
(-0)	- (+0)	= -0
(-0)	- (-0)	= +0

### 3.6. STANDARD ZAPISU INFORMACJI NA TAŚMIE MAGNETYCZNEJ

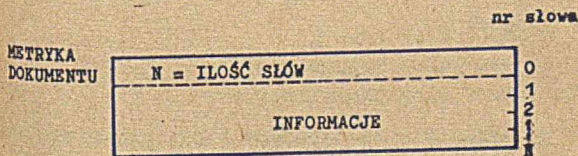
Istnieją różne sposoby zapisu informacji na taśmach magnetycznych. Niżej opisany sposób został przyjęty jako standard dla maszyny ZAM 41. Do tego standardu dostosowane są między innymi systemy TM-3 (tzw. SZMATA, w języku SAS i PJEG), TM-4 (w języku SAS) oraz współpraca z taśmą magnetyczną w języku COBOL.

Informacje przeznaczone do zapisania na taśmie są grupowane w dokumenty (rekordy), które stanowią logiczną jednostkę zapisu. Dokumenty mogą być stałej lub zmiennej długości.

Dokumenty stałej długości ( $N$  słów) mają następującą budowę:



Dokumenty zmiennej długości mają następującą budowę:



W celu zapisania na taśmie, dokumenty są łączone w bloki, stanowiące fizyczną jednostkę zapisu. Długość każdego bloku wynosi co najmniej 5 i co najwyżej 2048 słów. Bloki mogą mieć zmienną długość, a ich budowa jest następująca:

R	NR BLOKU	0
-----		
	N=ILOŚĆ SŁÓW INFORM.W BLOKU	1
-----		
	SUMA ARYTMET.SŁÓW 0 i 1	2
-----		
	SUMA KONTROLNA CZĘŚCI INFORMACYJNEJ	3-4
-----		
	METRYKA LUB KOLEJNE DOKUMENTY Z INFORMACJAMI	5 : : : N+

#### Objaśnienia

Bloki numerowane są na taśmie kolejno poczynając od 0. Rozróżnia się bloki organizacyjne i bloki informacyjne. Rodzaj bloku jest zaznaczony w bicie 0 zerowego słowa (na rysunku bit 0 oznaczono symbolem R). Znaczenie R jest następujące:

R=0 blok informacyjny

R=1 blok organizacyjny.

Wyróżnia się następujące bloki organizacyjne:

MEP - metryka początku pliku

MEK - metryka końca pliku

MKZ - metryka końca zapisu

MKS - metryka końca szpuli

RER - metryka rerunu.

Budowę metryk opisano w punktach 3.6.1-3.6.5.

Suma kontrolna jest liczona i sprawdzana wg następującego algorytmu zapisanego w języku . PJEG (A oznacza adres początku bloku, N - liczbę słów).

<u>liczenie</u>	<u>sprawdzanie</u>
PAR 46	PAR 46
UEB N-1	UEB N-1
* DOM A+5	* DOM A+5
SOB Q-1	SOB Q-1
ZZN	ZZN
PAM A+3	POL A+3
	SKO błąd
	SKO błąd
	ZAM
	POL A+4
	SKO błąd
	SKO błąd
	SKO dobrze

## 3.6.1. Metryka początku pliku

## 3.6.1. Metryka początku pliku

			nr słowa	nr uwagi
M	S	P	5	1
NR SEKCJI PLIKU			6	2
NAZWA PLIKU			7-12	3
DATA ZAPISANIA PLIKU			13	4
DATA WAŻNOSCI PLIKU			14	4
NR GENERACJI PLIKU			15	5
W	ILOSC POWT. ZAPISU		16	6
D	MAKS. DŁ. DOK.	MAKS. DŁ. BLOKU	17	7,8
0 1	11 12	23		

## Uwagi

- 1) Tekst w KW-8 (oktalnie 46505240).
- 2) Liczba binarna, dodatnia.
- 3) Tekst złożony z 18 znaków zapisanych w KW-8.
- 4) Dwie ostatnie cyfry roku - w bitach 0-8, dzień - w bitach 9-23.
- 5) Liczba binarna, dodatnia.

w KW-8.

- 4) Dwie ostatnie cyfry roku - w bitach 0-8, dzień - w bitach 9-23.
- 5) Liczba binarna, dodatnia.



- 6)  $W=0$  oznacza ostatni plik na taśmie,  
 $W=1$  oznacza, że mogą wystąpić dalsze pliki.
- 7)  $D=0$  oznacza, że dokumenty są zmiennej długości,  
 $D=1$  oznacza, że dokumenty są stałej długości.
- 8) Długości liczone wraz z metrykami.

### 6.2. Metryka końca pliku

			nr słowa	nr uwagi
M	E	K	5	1
DATA WAŻNOSCI PLIKU			6	2
SUMA KONTROLNA PLIKU			7-8	3

uwagi

- 1) Tekst w KW-8 (oktalnie 46505226).
- 2) Dwie ostatnie cyfry roku - w bitach 0-8,  
dzień - w bitach 9-23.
- 3) Suma sum kontrolnych wszystkich bloków pliku lub ostatniej sekcji.

### 6.3. Metryka końca zapisu

			nr słowa	nr uwagi
M	K	Z	5	1
DATA WAŻNOSCI OSTATNIEGO PLIKU			6	2
W			7	3

uwagi

- 1) Tekst w KW-8 (oktalnie 46513264).
- 2) Dwie ostatnie cyfry roku - w bitach 0-8,  
dzień - w bitach 9-23.
- 3)  $W=0$  oznacza ostatni plik na taśmie,  
 $W=1$  oznacza, że mogą wystąpić dalsze pliki.

## 3.6.4. Metryka końca szpuli

			nr słowa	nr uwagi
M	K	S	5	1
SUMA KONTROLNA CZĘŚCI PLIKU			6-7	2

## Uwagi

- 1) Tekst w KW-8 (oktalnie 46513246).
- 2) Suma sum kontrolnych wszystkich bloków pliku lub ostatniej sekcji, zapisanych dotychczas.

## 3.6.5. Metryka rerunu

			nr słowa	nr uwagi
R	E	R	5	1
			6	2
			i dalsze	

## Uwagi

- 1) Tekst w KW-8 (oktalnie 51105244).
- 2) Wg organizacji użytkownika.

### 3.7. WSPÓŁPRACA Z URZĄDZENIAMI ZEWNĘTRZNYMI

#### 3.7.1. Symboliczne wejście i wyjście zerowe

W celu skorzystania z symbolicznego wejścia lub wyjścia o numerze 0 wystarczy to wejście lub wyjście zdefiniować w czołówce, a w programie stosować rozkaz P31 (dla wejścia) lub P31 16 (dla wyjścia); w językach wyższego rzędu podprogramy wejścia i wyjścia na ogół używają tych rozkazów.

Jeśli symboliczne wejście o numerze 0 dotyczy pierwszego w zestawie czytnika taśmy papierowej, podprogramu PCTP oraz kodu M-2, to definicja tego wejścia może być pominięta; podobnie może być pominięta definicja symbolicznego wyjścia o numerze 0, jeśli dotyczy ona pierwszego w zestawie perforatora taśmy, podprogramu PPTP i kodu M-2.

Definiując wejście lub wyjście o numerze 0 należy pamiętać, że łączna długość podprogramu i dekodera, zarówno dla wejścia jak i wyjścia, nie może przekraczać 300 słów pamięci, gdyż są one dołączane przez system do obszaru pamięci określonego z góry; długości podprogramów i dekodów podane są w punktach 6.5 i 6.7.

W czasie działania programu symboliczne wejście i wyjście o numerze 0 może być dołączone (rozkazem SDY 1/11+i/13+8,  $i = 0$  dla wejścia,  $i = 1$  dla wyjścia) lub odłączone (rozkazem SDY 1/11+i/13+9,  $i = 0$  dla wejścia,  $i = 1$  dla wyjścia).

Przed dołączeniem wejścia lub wyjścia należy wpisać do 38 słowa PAO adres początku obszaru przeznaczonego dla tego wejścia lub wyjścia.

Zaleca się, aby dołączenie symbolicznego wejścia lub wyjścia o numerze 0 było poprzedzone jego odłączeniem.

Symboliczne wejście o numerze 0 translatora może być określone w czołówce problemu oraz może być zdefiniowane między krokami problemu operacją WEOT.

### 3.7.2. Symboliczne wejścia i wyjścia niezerowe

Symboliczne wejście i wyjście niezerowe nie są dołączane przez system w czasie sprowadzania programu do pamięci. Dlatego, aby z nich skorzystać, należy:

- zadeklarować w czołówce symboliczne wejście lub wyjście o dowolnym numerze w  $(1 \leq w \leq 15)$ ,
- zarezerwować obszar pamięci operacyjnej o długości nie mniejszej od sumy długości podprogramu i dekodera,
- przesłać adres początku zarezerwowanego obszaru do 38 słowa pamięci,
- dołączyć symboliczne wejście lub wyjście (rozkazem SDY  $1/11+i/13+w/17+8$ ,  $i=0$  dla wejścia,  $i=1$  dla wyjścia),
- korzystać z symbolicznego wejścia (rozkazem P31 w) lub wyjścia (rozkazem P31  $16+w$ ).

Dołączone wejścia lub wyjścia można odłączać (rozkazem SDY  $1/11+i/13+w/17+9$ ,  $i=0$  dla wejścia,  $i=1$  dla wyjścia), a odłączone - można dołączać (rozkazem SDY  $1/11+i/13+w/17+8$ ,  $i=0$  dla wejścia,  $i=1$  dla wyjścia). Przy dołączaniu należy pamiętać o ustawieniu adresu zarezerwowanego obszaru w 38 słowie pamięci. Odwoływać się do wejść (rozkazem P31 w) lub wyjść (rozkazem P31  $16+w$ ) można tylko wtedy, gdy są one dołączone.

Odlączanie i dołączanie wejść i wyjść ma przede wszystkim na celu czyszczenie bufora oraz umożliwienie racjonalnej gospodarki obszarem w PAO. Zaleca się, aby dołączenie wejścia lub wyjścia (oprócz pierwszego dołączenia) było poprzedzone jego odłączeniem.

### 3.7.3. Przesyłanie informacji za pomocą rozkazów SDY

Przesyłanie informacji między PAO i dowolnym urządzeniem zewnętrznym można zrealizować za pomocą rozkazów SDY; jest to najniższy, dopuszczony dla programu użytkowego poziom, w którym można napisać program przesyłania informacji. W tym celu należy:

- określić obszar w PAO, do lub z którego informacje będą przesyłane;
- przyporządkować przesłaniu jakiś numer  $np$ ,  $0 \leq np \leq 31$ ; zaleca się przyporządkowywanie przede wszystkim numerów 31, 30, ..., ponieważ numery przesłań  $np=0, 1, \dots$  są wykorzystywane przez symboliczne wejścia i wyjścia;
- zdefiniować przesłanie, określając urządzenie  $u$  (porównaj punkt 6.1), numer przesłania ( $np$ ), adres w PAO oraz ilość słów; wykonuje się to w sposób następujący:

$$\text{SDY } 1/10+u/17+np$$

$$\frac{\text{adres w PAO}}{\text{liczba słów}}$$

definicja przesłania może być jedna dla wielu przesłań;

- zainicjować przesłanie, wykonując:

$$\text{SDY } np$$

lub, gdy przesłanie dotyczy bębna:

SDY np.

1/3+adres bębnowy

Realizacja przesłania będzie zgodna z ostatnio zdefiniowanym przesłaniem o numerze np.

Po zainicjowaniu przesłania można wykonywać inne operacje; nie należy jednak zmieniać zawartości obszaru, z którego informacje są przesyłane, ani pobierać zawartości z obszaru, do którego dokonuje się przesyłanie informacji.

W trakcie realizacji przesłania - informacje o tym przesłaniu - są wpisywane do rejestru ZX(np); do badania tego rejestru stosuje się rozkaz

SDY 2/10+W/17+np

gdzie W jest sumą liczb w, określających stawiane pytania, zgodnie z poniższą tabelką:

w	pytanie
2	czy wystąpił zbliżający się koniec taśmy magnetycznej
4	czy przesłanie było błędne
8	czy przesłanie było puste
16	czy przesłanie było niepełne
32	czy w trakcie przesłania wystąpił nadmiar urządzenia
64	czy przesłanie zakończono

Jeśli odpowiedź na co najmniej jedno z zadanych pytań jest pozytywna, wówczas  $LR:=LR+1$ ; w przeciwnym przypadku  $LR:=LR+2$ .

Można czekać na zakończenie przesłania, a następnie badać rejestr ZX(np); w tym przypadku stosuje się rozkaz

SDY 3/10+W/17+np

Znaczenie W oraz zmiana LR są takie same, jak w punkcie poprzednim.

W przypadku współpracy z taśmą magnetyczną, zaleca się otwarcie urządzenia przed pierwszym zdefiniowaniem przesłania oraz zamknięcie urządzenia po wykonaniu ostatniego przesłania.

Inne możliwości współpracy z urządzeniami zewnętrznymi są podane przy opisie rozkazów SDY (patrz p. 3.4.5).





#### 4. WŁĄCZANIE ZBIORU DO BIBLIOTEKI SYSTEMU

Zamiast pełnego opisu sposobu włączania dowolnego zbioru do biblioteki systemu podano kilka przykładów, które nie wyczerpują tematu, lecz komentują następujące zagadnienia:

- typ i organizację zbioru,
- przygotowanie pliku źródłowego zbioru,
- produkowanie taśmy binarnej,
- wprowadzenie zbioru do biblioteki systemu,
- wywoływanie zbioru z biblioteki systemu.

Oddzielny punkt omawia wprowadzenie nowych funkcji języka SAS.

Podane w dalszym ciągu czołówki zakładają, że dane są wprowadzane przez pierwszy czytnik w kodzie M-2, a wyniki są wypisywane na drukarce.

#### 4.1. TYP I ORGANIZACJA ZBIORU

Istnieją trzy typy zbiorów: program użytkowy (U), program - translator (P) i tekst (T); w dalszym ciągu będziemy rozważać tylko typ U (program użytkowy).

Istnieją dwie organizacje zbioru: zbiór zwarty (X) i zbiór złożony (Y). Z punktu widzenia użytkownika podstawowa różnica między zbiorem o organizacji zwartej i złożonej polega na sposobie ich wywoływania.

Wywołanie zbioru zwartego powoduje wykonanie następujących czynności:

- sprowadzenie zbioru z biblioteki (z taśmy systemu lub bębna systemu) na bęben użytkownika
- sprowadzenie do PAO sekcji startowej i przejście do jej wykonywania.

Wywołanie zbioru zwartego niszczy więc zawartość bębna użytkownika.

Zbiór złożony składa się z podzbiorów (w szczególnym przypadku z 1 podzbioru). Wywołanie zbioru złożonego powoduje bezpośrednie przesłanie podzbioru startowego z biblioteki systemu (taśmy systemu lub bębna systemu) do PAO i rozpoczęcie wykonywania tego podzbioru. Proces ten nie niszczy zawartości bębna użytkownika.

Wprowadzenie do biblioteki systemu programu napisanego w języku innym niż PJEK i traktowanego jako zbiór złożony jest bardzo trudne, a czasami niewykonalne.

Każdy zbiór ma nazwę złożoną z 1 do 4 znaków, przy czym pierwszy znak musi być literą, różną od litery O. Każdy podzbiór zbioru złożonego mu-

si mieć nazwę spełniającą ten sam warunek. Nazwa jednego z podzbiorów może być identyczna z nazwą zbioru.

#### 4.2. PRZYGOTOWANIE PLIKU ŹRÓDŁOWEGO ZBIORU

##### 4.2.1. Przykład 1

Program użytkowy (TYP=U), o nazwie XYZ (NZB=XYZ) napisany w języku PJEG, traktowany jest jako zbiór złożony (ORG=Y), składający się z jednego podzbioru, również o nazwie XYZ. Przygotowanie pliku źródłowego tego zbioru polega na zapisaniu za pomocą SMAD na taśmie magnetycznej następującego tekstu:

::- - - -

:[W] - XYZ -

P: BIB1;

T: TRANSLACJA XYZ.

::- - - -

:[W] - XYZ.BIB -

Z: TYP=U, ORG=Y, NZB=XYZ, PZS=XYZ;

P: NPZ=XYZ;

T: TRA=PJEG;

KONIEC;

::- - - -

:[W] - XYZ.ED -

VWY [

XYZ - POCZATEK TRANSLACJI

EDYCJA: $\alpha\beta\gamma$ - 1

]

::- - - -

tekst programu XYZ w języku PJEG  
do dyrektywy VST wyłącznie

::- - - -

:[W] - XYZ.WS -

VWP [wymagana PAO]

VWB [wymagana PB]

VPP [początek programu na bębnie]

VDP [długość programu]

VST [parametry VST]

## 4.2.2. Przykład 2

Program użytkowy (TYP=U) o nazwie XYZ (NZB=XYZ), napisany w języku PJEG, traktowany jako zbiór złożony (ORG=Y), składa się z 3 podzbiorów o nazwach A, B i C. Działanie programu zaczyna się od podzbioru B (PZS=B). Wówczas na taśmie magnetycznej należy za pomocą SMAD zapisać następujący tekst:

```
::- - - -
```

```
:[W] - XYZ -
```

```
P: BIB1;
```

```
T: TRANSLACJA XYZ.
```

```
::- - - -
```

```
:[W] - XYZ.BIB -
```

```
Z: TYP=U, ORG=Y, NZB=XYZ, PZS=B;
```

```
P: NPZ=A;
```

```
T: TRA=PJEG;
```

```
P: NPZ=B;
```

```
T: TRA=PJEG;
```

```
P: NPZ=C;
```

```
T: TRA=PJEG;
```

```
KONIEC;
```

```
::- - - -
```

```
:[W] - XYZ.ED -
```

```
VWY [
```

```
XYZ - POCZATEK TRANSLACJI
```

```
EDYCJA:  $\alpha\beta\gamma$ - 1
```

```
]
```

```
::- - - -
```

tekst programu XYZ, części A, B i C  
do dyrektywy VST wyłącznie

:- - - -

:[W] - XYZ.WS1 -

VWP [wymagana PAO dla części A]  
 VWB [wymagana PB dla części A]  
 VPP [początek części A na bębnie]  
 VDP [długość części A]  
 VST [parametry VST dla części A]

:- - - -

:[W] - XYZ.WS2 -

VWP [wymagana PAO dla części B]  
 VWB [wymagana PB dla części B]  
 VPP [początek części B na bębnie]  
 VDP [długość części B]  
 VST [parametry VST dla części B]

:- - - -

:[W] - XYZ.WS3 -

VWP [wymagana PAO dla części C]  
 VWB [wymagana PB dla części C]  
 VPP [początek części C na bębnie]  
 VDP [długość części C]  
 VST [parametry VST dla części C]

Wymienione wyżej parametry dla części B i C winny być zadane za pomocą liczb całkowitych i T-etykiet.

#### 4.2.3. Przykład 3

Wszystkie dane są takie same, jak w przykładzie 1, tylko program ma być włączony jako zbiór zwarty.

W zasadzie przygotowanie programu jest identyczne jak w przykładzie 1, zachodzą jedynie następujące drobne różnice:

- na stronie XYZ.BIB zamiast ORG=Y należy wpisać ORG=X (zbiór zwarty),
- na stronie XYZ.WS oprócz podanych dyrektyw należy jeszcze dopisać VSP [spodziewany adres programu na bębnie]. Spodziewany adres programu na bębnie określa adres bębnowy, od którego ma być wpisywany program wywołany do wykonania (wywołanie zbioru zwartego powoduje wpisanie tego zbioru na bęben użytkownika, nawet wówczas, gdy był on adaptowany na bęben systemu). Dyrektywa VSP musi być oczywiście wpisana przed dyrektywą VST,
- parametry dyrektywy VST dotyczą tylko sekcji startowej; wywołanie programu przez PROBLEM: XYZ; spowoduje wprowadzenie do PAO tylko tej sekcji.

#### 4.2.4. Przykład 4

Program użytkowy (TYP=U) o nazwie LUX (NZB=LUX), napisany w języku SAS jest traktowany jako zbiór zwarty. Wówczas należy zapisać za pomocą SMAD na taśmie magnetycznej następujący tekst:

```
;;- - - -
```

```
:[W] - LUX -
```

```
P: MSAS,WYSE,BIB1;
```

```
T: TRANSLACJA LUX.
```

```
;;- - - -
```

```
:[W] - PROG -
```

tekst programu LUX w języku SAS

```
;;- - - -
```

```
:[W] - LUX.BIB -
```

```
Z: TYP=U,ORG=X,NZB=LUX;
```

```
T: TRA=PJEG;
```

```
KONIEC;
```

:: - - - -

:[W]

- LUX.WST -

VPB [ ]

VCM [ 1 ]

VDP [ K+5 ]

VWB [ K+5 ]

VWP [wymagana PAO]

VPP [ 0 ]

VSP [ 0 ]

VKB

VST [ 0,5, Q+141, Q+143 ]

## 4.3. PRODUKOWANIE TAŚMY BINARNEJ

Po przygotowaniu pliku źródłowego zbioru w sposób opisany w punkcie 4.2 należy wykonać następujący problem (x jest jedną z liter A,B,...,F, określającą jednostkę taśmy magnetycznej, na której jest zapisany wyprowadzany zbiór):

PROBLEM: POPR, JOM;

WEJSCIA: 0 = PCTM (TMx),

1 = PCT6 (TMx),

2 = PCTP (CPA, DM2C);

WYJŚCIA: 0 = PDW (DWA, DDW1),

1 = PPTP (DPA, DM2P),

2 = PPTP (DPA, DJED);

TYTUL: WYPROWADZENIE TAŚMY BINARNEJ.

}

odcinek ok. 50 cm czystej taśmy

::OD

::PR: nazwa zbioru

::KO

Wykonanie tego problemu spowoduje wypisanie taśmy binarnej na pierwszym perforatorze taśmy w zestawie.



## 4.4. WPROWADZENIE ZBIORU DO BIBLIOTEKI SYSTEMU

W celu wprowadzenia zbioru (np. o nazwie XYZ) do biblioteki systemu należy wykonać następujący problem (x,y,z wyznaczają jednostki taśm magnetycznych):

PROBLEM: BIB2;

WEJSCIA: 0=PCTP (CPA, DM2C),  
1=PCTP (CPA, DJED),  
2=PCTM (TMx);

WYJSCIA: 0=PDW (DWA, DDW1),  
1=PDW (TMy),  
2=PDW (TMz).

E:abc, def;

R:ALFA;

W:XYZ;

} odcinek ok. 50 cm czystej taśmy

} taśma binarna

} odcinek ok. 50 cm czystej taśmy

R;

K;

W wyniku wykonania tego problemu zostanie zapisana na jednostce TMz edycja abc (numer generacyjny def, def - liczba do 3 cyfr) taśmy systemu. Nowa taśma systemu zawiera wszystkie te elementy, które zawierała taśma systemu założona na jednostce TMx (nie może to być aktualnie działająca taśma systemu) oraz zbiór XYZ, wpisany do biblioteki systemu między zbiór ALFA i zbiór następny. Taśma y jest taśmą roboczą.

Jeśli zbiór XYZ ma być zapisany na początku biblioteki, to R:ALFA; powinno być pominięte. Jeśli zbiór XYZ ma być zapisany na końcu bibliote-

ki systemu, to R:ALFA; powinno być zastąpione przez R;, zaś R; znajdujące się przed K; powinno być pominięte.

#### 4.5. WYWOŁANIE ZBIORU Z BIBLIOTEKI SYSTEMU

Wywołanie zbioru z biblioteki systemu może się odbywać przez rozkazy SDY lub przez JOM. Poniższe przykłady ilustrują wywołanie zbioru przez JOM.

##### 4.5.1. Przykład 1

Wywołanie zbioru zwartego o nazwie XYZ:

PROBLEM: XYZ;

Zbiór XYZ zostanie sprowadzony na bęben użytkownika, a jego sekcja startowa do PAO, po czym rozpocznie się wykonywanie sekcji startowej.

Jeśli zbiór XYZ jest programem napisanym w języku MSAS, to jednocześnie zbiór ten zostanie dodatkowo wprowadzony na bęben użytkownika począwszy od adresu 0; zawartość początku bębna użytkownika zostanie więc zniszczona.

##### 4.5.2. Przykład 2

Wywołanie podzbioru KSI, wchodzącego w skład zbioru złożonego o nazwie XYZ, składającego się z jednego lub kilku podzbiorów (podzbiór KSI nie musi być podzbiorem startowym):

PROBLEM: XYZ (KSI);

Podzbiór KSI można również wywołać w następujący sposób (por. opis odpowiedniego rozkazu SDY):

SDY 1/17+6

(XYZ)

(KSI)

## 4.5.3. Przykład 3

Jeśli w przykładzie 2 nazwa podzbioru KSI zostanie zmieniona na XYZ (tzn. podzbiór ten ma taką samą nazwę jak cały zbiór złożony), to podzbiór ten można wywołać następująco:

PROBLEM: XYZ;

## 4.6. WPROWADZENIE NOWYCH FUNKCJI JĘZYKA SAS

W celu wprowadzenia nowych funkcji języka SAS do biblioteki systemu trzeba mieć pliki źródłowe i odpowiadające im tabulogramy niektórych elementów biblioteki systemu. Ponadto niezbędna jest znajomość budowy tych elementów.

## 4.6.1. Budowa zbioru MAKR

Plik źródłowy zbioru MAKR jest zbudowany następująco (z prawej strony podane są komentarze):

```
::- - - -
```

```
:[W] - MAKR -
```

```
P: BIB1,POPR,JOM;
```

```
T: TRANSLACJA MAKR.
```

```
::- - - -
```

```
:[W] - MAKR.BIB -
```

```
Z: ORG=X,TYP=T,NZB=MAKR,PZS=SKMA;
```

```
T: TRA=MSAS;
```

```
KONIEC;
```

```
::- - - -
```

```
:[W] - MAKR.ED -
```

```
:MAKROSY-TRANSLACJA
```

```
:EDYCJA :edycja i numer kolejny
```

:- - - -

:[W]

- M.1 -

MAKRO;

SEST A           początek listy nazw funkcji

'DDW.E'		DW
'DDK.E'	nazwy funkcji	DK
'DDZN'	(. i do trzech	<u>DZN</u>
:	znaków) zapi-	:
:	sane jako teks-	:
:	ty w SAS	

'L.3E'           -0, koniec listy nazw funkcji  
ze zbioru FUN1

'FUN1'           nazwa zbioru, w którym są za-  
pisane funkcje DW,DK,DZN,...

'DSCT'

  :

'L.3E'           -0, koniec listy nazw funkcji  
ze zbioru FUVG

'FUVG'           nazwa zbioru, w którym są za-  
pisane funkcje ...

'4E'            +0, koniec listy nazw funkcji

:- - - -

:[W]

- M.2 -

MAKR DRUKUJ    początek pieczętki dla makro-  
PODP            rozkazu DRUKUJ

UMA ;1         ;1 oznacza pierwszy argument

UMM ;2         z listy argumentów makrorozkazu

PAD\* +6        DRUKUJ

PAD\* +10

UMM ;3

PMM\* +5

PMM\* +9

SSL .DW        odwołanie do podprogramu DW

+0

+0

+0

  :

SKO 14	
FUNK .DW	żądanie dołączenia funkcji DW
FUNK .DK	i DK
KOPO	koniec pieczętki dla makrorozkazu DRUKUJ
MAKR ....	
⋮	
KOPO	koniec pieczętki dla ostatniego makrorozkazu
KONS	

#### 4.6.2. Przygotowanie funkcji do włączenia do systemu

Dla włączenia funkcji do taśmy systemu należy zmodyfikować zbiór MAKR. W tym celu należy:

- wymienić na stronie MAKR-ED symbol edycji i numer kolejny
- dopisać na stronie M.1 (lub M.1A) nazwę funkcji do listy nazw funkcji; jeśli nowo wprowadzana funkcja ma być dołączana jako podzbiór do jednego z istniejących zbiorów (np. FUN1), to jej nazwę należy wpisać jako tekst powyżej nazwy odpowiedniego zbioru; jeśli wprowadzana funkcja ma być zapisana w nowo tworzonym zbiorze, to przed tekstem '.4E' należy dopisać tekst:

'D nazwa funkcji'

'L.3E'

'nazwa zbioru'

- jeśli funkcja ma być wywoływana jako makrorozkaz, to odpowiednią pieczętką, zbudowaną na wzór podany dla DRUKUJ w punkcie 4.6.1. należy dołączyć do zbioru MAKR w takim miejscu, żeby kolejność pieczętek była zgodna

z kolejnością nazw funkcji na liście nazw.

Dla tak zmodyfikowanego zbioru MAKR należy wyprodukować taśmę binarną w sposób opisany w punkcie 4.3. Poprawna translacja zbioru MAKR powinna być zakończona przyczyną 56.

Jeśli funkcja jest dołączana do istniejącego zbioru funkcji (np. do zbioru FUN1), to należy ją wprowadzić do tego zbioru jako oddzielny podzbiór oraz odpowiednio zmodyfikować stronę, na której są wymienione wszystkie podzbiory tego zbioru (dla funkcji XYZ stroną XYZ.BIB). Następnie należy wyprodukować taśmę binarną zmodyfikowanego zbioru.

Jeśli funkcja tworzy podzbiór nowego zbioru, to należy ten zbiór przygotować zgodnie z opisem podanym w punkcie 4.2, a następnie wyprodukować taśmę binarną tego zbioru.

Sposób pisania funkcji podany jest w opisie języka SAS.

Poprawna translacja zbioru zawierającego dogrywaną funkcję winna być zakończona przyczyną 55 dla każdej funkcji.

Należy pamiętać, że w procesie tym nie są wykrywane wszystkie błędy; niektóre błędy formalne mogą być wykryte dopiero w procesie translacji programu użytkowego, korzystającego z wprowadzanych funkcji.

Po wyprodukowaniu taśm binarnych zbioru MAKR i zbioru zawierającego funkcję należy te taśmy wprowadzić do biblioteki systemu. Jeśli funkcja jest wprowadzana do istniejącego zbioru, to należy wykonać następujący problem:

PROBLEM: BIB2;

WEJSCIA: 0 = PCTP (CPA, DM2C),

1 = PCTP (CPA, DJED),

2 = PCTM (TMx);

WYJSCIA: 0 = PDW (DWA, DDW1),

1 = PDW (TMy),

2 = PDW (TMz).

E...oc, def;

P:MAKR;

W:MAKR;

} odcinek ok. 50 cm czystej taśmy

} taśma binarna zbioru MAKR

} odcinek ok. 50 cm czystej taśmy

P:nazwa zbioru wymienianego;

W:nazwa zbioru wymienianego;

} odcinek ok. 50 cm czystej taśmy

} taśma binarna zbioru wymienianego

} odcinek ok. 50 cm czystej taśmy

R;

K;

Jeśli dana funkcja tworzy nowy zbiór, to należy wykonać następujący problem:

PROBLEM: BIB2;

WEJSCIA: 0 = PCTP (CPA, DM2C),

1 = PCTP (CPA, DJED),

2 = PCTM (TMx);

WYJSCIA: 0 = PDW (DWA, DDW1),

1 = PDW (TMy),

2 = PDW (TMz).

E:abc,def;

P:MAKR;

W:MAKR;

{ odcinek ok. 50 cm czystej taśmy

{ taśma binarna zbioru MAKR

{ odcinek ok. 50 cm czystej taśmy

R:nazwa zbioru poprzedzającego zbiór wprowadzany;

W:nazwa zbioru wprowadzanego;

{ odcinek ok. 50 cm czystej taśmy

{ taśma binarna zbioru wprowadzanego

{ odcinek ok. 50 cm czystej taśmy

R;

K;

W obu przypadkach znaczenie x,y,z oraz abc i def jest takie, jak w punkcie 4.4.



## 5. OPEROWANIE MASZYNĄ

### 5.1. STARTOWANIE SYSTEMU OPERACYJNEGO

Startowanie systemu operacyjnego jest czynnością polegającą na wykonaniu przez maszynę i przez operatora takich operacji, aby translatory i programy użytkowe mogły działać pod nadzorem systemu operacyjnego. Z punktu widzenia operatora startowanie systemu operacyjnego SO-141 można podzielić na następujące etapy:

- adaptacja SO:
  - wprowadzenie ściągaczki SO z taśmy systemu,
  - określenie elementów przenoszonych z taśmy systemu na bęben;
- określenie parametrów SO.

Wymienione czynności zostaną opisane w punktach 5.1.1, 5.1.2 i 5.1.3. W punkcie 5.2 zostaną opi-

sane uproszczone metody startowania SO.

### 5.1.1 Wprowadzenie ściągaczki SO z taśmy systemu

Ściągaczka SO z taśmy systemu znajduje się na ogół w każdym ośrodku wyposażonym w maszynę ZAM 41. Sposób przygotowania tej ściągaczki jest podany w punkcie 7.15.1.

Dla wprowadzenia ściągaczki należy wykonać następujące czynności:

- nacisnąć klucz STOP na stoliku operatora,
- zablokować zapis na wybranej jednostce taśmy magnetycznej i założyć na tę jednostkę taśmę systemu SO-141,
- odblokować bębny,
- wycisnąć na stoliku operatora klucze 0-23 oraz A, M, B, L i R,
- podłożyć taśmę ze ściągaczką do czytnika, ewentualnie włączyć gotowość czytnika,
- nacisnąć klucz ŁADUJ,
- wcisnąć klucz L,
- nacisnąć klucz ŁADUJ,
- nacisnąć 5 razy klucz WPROWADŹ,
- nacisnąć klucz ŁADUJ,
- wycisnąć klucz L,
- nacisnąć klucz START,
- po wczytaniu taśmy papierowej wcisnąć klucz, odpowiadający jednostce, na której znajduje się taśma systemu;

nr klucza podaje poniższa tabelka:

kanał	jednostka	16	17	18
1		3	4	5
2		6	7	8

- nacisnąć klucz START.

Zakończenie wprowadzania ściągaczki jest sygnalizowane zapaleniem na stoliku operatora lampki nad kluczem 23. Operator powinien wówczas wykonać drugą część adaptacji, tj. określić elementy taśmy systemu, które mają być przeniesione na bęben.

#### 5.1.2. Określenie elementów przenoszonych z taśmy systemu na bęben

Określenie elementów taśmy systemu, które mają być przenoszone na bęben obejmuje:

- a) włączenie monitora do pracy,
- b) zaproponowanie przez maszynę standardowego zestawu elementów taśmy systemu ładowanych lub przenoszonych na bęben,
- c) zaakceptowanie lub zmianę przez operatora ww zestawu,
- d) sprowadzenie żadanego zestawu z taśmy systemu na bęben,
- e) zablokowanie części bębna.

Czynności b i d są wykonywane przez maszynę, czynności a, c i e winien wykonać operator. W trakcie wykonywania wymienionych czynności na monitorze są wypisywane pewne informacje. Informacje wypisywane przez maszynę rozpoczynają się tekstem O: , jeśli natomiast maszyna żąda wprowadzenia informacji, wówczas wypisywane są znaki O; , po których operator powinien wprowadzić potrzebną informację zakończoną symbolem \* . Jeśli w trakcie pisania na monitorze, operator pomyli się, to może skasować błędną informację przez wpisanie poprawnej informacji w nowym wierszu. O ile błąd nie zostanie przez operatora zauważony, wówczas maszyna wypisze jeden z następujących tekstów:

## NIE ROZUMIEM

BLEDNA NAZWA - nazwa

co oznacza, że informacja nie została przyjęta przez maszynę; operator może wówczas poprawną informację wprowadzić ponownie.

Włączenie monitora do pracy polega na naciśnięciu na stoliku operatora kolejno kluczy: NIEGOTÓW, GOTÓW, ZGŁOSZENIE OPERATORA; jeśli przed wykonaniem tej czynności lampka GOTÓW była zapalona, wówczas można pominąć naciskanie kluczy NIEGOTÓW i GOTÓW.

Zaproponowanie przez maszynę standardowego zestawu elementów ładowanych lub przenoszonych z taśmy systemu na bęben polega na wypisaniu nazwy wariantu SO, nazwy zestawu programów obsługi symbolicznych wejść-wyjść oraz nazw wybranych elementów biblioteki systemu. Najczęściej zestawem standardowym jest:

SV, SWW, POPR, JOM, PJEG, BIB1, BIB2

Zaakceptowanie lub zmiana zestawu polega na napisaniu przez operatora na monitorze kolejno jednego lub kilku z następujących tekstów:

SUP - nazwa wariantu SO \*

jeśli zamiast proponowanego w standardzie wariantu systemu operacyjnego ma być ładowany inny wariant;

ZMI - nazwa 1 - nazwa 2 - ... - nazwa n \*

jeśli zamiast proponowanych w standardzie elementów biblioteki systemu operator chce sprowadzić na bęben elementy o nazwach: nazwa 1, nazwa 2, ..., nazwa n;

UZU - nazwa 1 - nazwa 2 - ... - nazwa n \*

jeśli oprócz już zgłoszonych do sprowadzenia

elementów biblioteki systemu operator chce sprowadzić na bęben elementy o nazwach: nazwa 1, nazwa 2, ..., nazwa n;

DEK - nazwa żadanego zestawu programów obsługi SWW\*

jeśli operator chce sprowadzić na bęben nie-standardowy program obsługi wejść-wyjść;

SYS-SO41\* lub SYS-SO42\*

jeśli operator chce działać w Systemie SO-41 lub SO-42 zamiast SO-141;

KON\*

jeśli operator wprowadził wszystkie zmiany lub jeśli akceptuje standard.

Spis nazw, których można używać w zdaniach SUP, ZMI, UZU i DEK podano w punkcie 7.16.

Sprowadzenie żadanego zestawu z taśmy systemu na bęben jest sygnalizowane wypisaniem na monitorze nazw ładowanego SO i sprowadzanego zestawu programów obsługi wejść-wyjść i elementów biblioteki systemu oraz wypisaniem tekstu ZABLOKUJ BEBS - n KL.

Jeśli łączna objętość żadanych do przeniesienia zbiorów przekracza przeznaczony na ten cel obszar bębna, to niektóre z nich mogą nie zostać przeniesione na bęben. Nazwy zbiorów przeniesionych na bęben są wypisywane na monitorze.

Zablokowania części bębna powinien dokonać operator, przy czym liczba kluczy, którymi należy zablokować bęben, jest podana na monitorze.

Po wykonaniu wymienionych czynności zapala się lampka na stoliku operatora nad kluczem 23. Można wówczas przystąpić do określenia parametrów SO.

### 5.1.3. Określenie parametrów SO

Określenie parametrów SO obejmuje:

- a) włączenie do pracy monitora;
- b) zaproponowanie następujących parametrów SO: jednostki, na której jest założona taśma systemu, podziału na zestawy, numeru problemu, drukarki, daty, godziny;
- c) zaakceptowanie lub zmiana przez operatora ww parametrów.

Czynność b wykonuje maszyna, czynności a i c powinien wykonać operator. W trakcie wykonywania powyższych czynności na monitorze są wypisywane odpowiednie informacje. Informacje wypisywane przez maszynę rozpoczynają się od tekstu O: ,jeśli natomiast maszyna żąda wprowadzenia informacji, wówczas wypisywane są znaki O; ,po których operator powinien wprowadzić potrzebną informację zakończoną znakiem \* . Jeśli w trakcie pisania na monitorze operator pomyli się, to może skasować błędną informację przez wpisanie poprawnej informacji w nowym wierszu. O ile błąd nie zostanie przez operatora zauważony, wówczas maszyna wypisze jeden z następujących tekstów:

NIE ROZUMIEM  
 identyfikator - XXX  
 BYLY BLEDY

Wypisanie tekstu NIE ROZUMIEM oznacza, że wprowadzone błędnie informacje zostały całkowicie pominięte. Poprawną informację można wprowadzić ponownie.

Jeśli maszyna wypisze identyfikator - XXX, np. DATA - XXX, DRUK - XXX, to oznacza, że zostały wprowadzone błędne parametry. Jeśli popraw-

ne parametry nie zostaną wprowadzone, wówczas zostanie wypisany tekst **BYLY BLEDY**, po którym znów można wprowadzić poprawne parametry. Poprawnie wprowadzone parametry są przez maszynę potwierdzane.

Włączenie monitora do pracy polega na naciśnięciu na stoliku monitora klucza **ZGŁOSZENIE OPERATORA**; jeśli przed wykonaniem tej czynności lampka **NIEGOTÓW** była zapalona, wówczas naciśnięcie klucza **ZGŁOSZENIE OPERATORA** powinno być poprzedzone naciśnięciem klucza **GOTÓW**.

Zaproponowanie przez maszynę parametrów **SO** polega na wypisaniu następującego tekstu:

```
O:-----
   SO 141 ...
O:ZEST- ...
   :DATA- ...
   :PROB- ...
O:DRUK- ...
   :TMS - ...
O:GODZ- ...
```

przy czym na miejscu kropek wypisywane są proponowane parametry.

Zaakceptowanie lub zmiana parametrów polega na napisaniu na monitorze jednego lub kilku z następujących tekstów (kolejność tekstów jest dowolna, z wyjątkiem tekstu **START**):

**DATA** - dzień - miesiąc - rok \*

jeśli należy zmienić datę; dzień - miesiąc - rok stanowią wprowadzaną datę;

**DRUK** - numer drukarki - typ drukarki - liczba wierszy na stronie \*

jeśli należy zmienić parametry drukarki; typ drukarki wynosi 1 dla DW-1, 2 dla DW-2 i 4 dla DW-2E;

PROB - numer problemu \*

jeśli operator chce, aby numeracja problemów rozpoczęła się od podanego numeru problemu + 1;

TMS - numer modułu - numer kanału \*

jeśli taśma systemu została przeniesiona na inną jednostkę lub

TMS-BRAK \*

jeśli taśma systemu ma być odłączona;

ZEST - liczba zestawów - sposób podziału \*

jeśli operator chce zmienić sposób podziału maszyny na zestawy; liczba zestawów i sposób podziału oznaczają:

- 1-C jeden zestaw 0 obejmujący całą maszynę;
- 2-A dwa zestawy 0 i 1, pierwszy zawiera połowę wszystkich urządzeń ewentualnie zaokrągloną w górę, drugi - pozostałe urządzenia;
- 2-B dwa zestawy 0 i 1 tak jak 2-A, z tą różnicą, że 4 jednostki taśmy magnetycznej należą do pierwszego zestawu, a pozostałe do drugiego;

GODZ - godzina - minuta - sekunda \*

jeśli operator chce zmienić aktualny czas;

ADAPTACJA \*

jeśli operator chce powtórzyć od początku czynności opisane w p. 5.1.2;

START \*

jeśli wszystkie potrzebne zmiany parametrów zostały już dokonane, bądź akceptuje się parametry proponowane przez maszynę.



Przykładami opisanych tekstów są:

DATA - 6 - 12 - 72 \*

PROB - 15 \*

DRUK - 1 - 4 - 66 \*

Po napisaniu przez operatora tekstu START maszyna wypisze na monitorze tekst

0•START SO 141

---

po czym na stoliku operatora zapalą się lampki nad kluczami 22 i 23. Oznacza to gotowość maszyny do wprowadzania programów.

## 5.2. UPROSZCZONE METODY STARTOWANIA SO

Oprócz opisanego w punkcie 5.1 sposobu startowania SO istnieją sposoby uproszczone, polegające na wykonaniu prostych operacji, w wyniku których można pominąć czynności opisane w punkcie 5.1.1 i 5.1.2. Metody te nie zawsze są skuteczne. Zostały one opisane w punktach 5.2.1, 5.2.2 i 5.2.3. W punkcie 5.2.4 opisany został sposób ominięcia czynności opisanych w punkcie 5.1.3.

### 5.2.1. Regeneracja SO

Regeneracja SO polega na wciśnięciu na stoliku operatora klucza 16 i naciśnięciu klucza ZGŁOSZENIE OPERATORA. Regeneracja jest możliwa tylko wówczas, gdy maszyna jest w stanie "praca" (zapalona lampka PRACA).

Metoda ta jest skuteczna tylko wówczas, gdy nie zostały zniszczone w PAO fragmenty superwizora dotyczące obsługi przerw oraz sprowadzania systemu operacyjnego z bębna.

### 5.2.2. Start z 15 słowa PAO

Start z 15 słowa pamięci polega na wykonaniu następujących czynności na stoliku operatora:

- naciśnięcie klucza STOP,
- wyciśnięcie kluczy 4-19 oraz A,M,B,L,R,
- wciśnięcie kluczy 20-23 (znaczenie kluczy 0-3 jest opisane w punkcie 5.2.4),
- naciśnięcie klucza ŁADUJ,
- wciśnięcie klucza L,
- naciśnięcie klucza ŁADUJ,
- wyciśnięcie klucza L,
- naciśnięcie klucza START.

Metoda ta jest skuteczna tylko wówczas, gdy nie został zniszczony w PAO fragment superwizora dotyczący sprowadzania systemu operacyjnego z bębna.

### 5.2.3. Startowanie SO za pomocą ściągaczki z bębna

Metoda ta wymaga posiadania tzw. "ściągaczki SO z bębna". Sposób przygotowania takiej ściągaczki został podany w punkcie 7.15.2.

W celu startowania SO za pomocą "ściągaczki SO z bębna" należy na stoliku operatora:

- nacisnąć klucz STOP,
- wycisnąć klucze 4-23 oraz A,M,B,L,R, (znaczenie kluczy 0-3 opisano w punkcie 5.2.4),
- podłożyć taśmę ze ściągaczką do czytnika, ewentualnie włączyć gotowość czytnika,
- nacisnąć klucz ŁADUJ,
- wcisnąć klucz L,
- nacisnąć klucz ŁADUJ,
- nacisnąć 5 razy klucz WPROWADZ,
- nacisnąć klucz START.

Metoda ta jest skuteczna wówczas, gdy na bębnie nie uległy zniszczeniu system operacyjny, zestaw programów obsługi SWW oraz elementy biblioteki systemu.

#### 5.2.4. Pominięcie określania parametrów SO

Startowanie SO (zarówno pełne jak uproszczone) może być realizowane w dwóch reżimach, w zależności od położenia kluczy 0-3 na stoliku operatora.

Jeśli klucze 0-3 są wciśnięte, to czynności opisane w punkcie 5.1.3 zostają pominięte. Parametrami SO są wówczas parametry ostatnio określone.

Jeśli choć jeden z kluczy 0-3 jest wyciśnięty, wówczas czynności opisane w punkcie 5.1.3 są wykonywane.

### 5.3. OBSŁUGA STOLIKA OPERATORA I URZĄDZEŃ POMOCNICZYCH

#### 5.3.1. Znaczenie kluczy i lampek na stoliku operatora

Przy operowaniu maszyną w SO-141 korzysta się z kluczy i lampek schematycznie pokazanych i opisanych na rys. 12.

ZESTAW 1

ZESTAW 0

LAMPKI SYGNALIZUJACE:

KONIEC PROBLEMU ALARM	CIAG DALSZY TASMU	WLAZENIE PERFORATORA	WLAZENIE KONIEC PROBLEMU
⊗	⊗	⊗	⊗

LAMPKI SYGNALIZUJA

21

18

KLUCZE POWODUJACE:

WYRZUCENIE PROBLEMU	CIAG DALSZY TASMU	WLAZENIE PERFORATORA	WYRZUCENIE PROBLEMU
◇	◇	◇	◇

KLUCZE POWODUJACE



ZŁOZENIE OPERATORA

Fig. 12. Niektóre klucze i lampki na stoliku operatora

Klucze 19-23 oraz odpowiadające im lampki dotyczą zestawu 0 we wszystkich reżimach pracy (1-C, 2-A i 2-B), natomiast klucze 17-18 i odpowiadające im lampki dotyczą zestawu 1, a nie mają znaczenia w reżimie 1-C.

Położenie kluczy jest sprawdzane w momencie naciśnięcia klucza ZGŁOSZENIE OPERATORA. Jeśli zatem operator chce wykonać jakąś czynność sterowaną ze stolika operatora, to powinien odpowiednio wcisnąć lub wycisnąć właściwy klucz i nacisnąć ZGŁOSZENIE OPERATORA, pamiętając jednocześnie o tym, że równocześnie zostanie odczytane położenie innych kluczy.

Znaczenie poszczególnych kluczy jest następujące (podana interpretacja dotyczy klucza wciśniętego):

w y r z u c e n i e   p r o g r a m u - żądanie zakończenia działania programu i przejście do następnego kroku w problemie;  
 w y r z u c e n i e   p r o b l e m u - żądanie zakończenia całego problemu;  
 w ł ą c z e n i e   p e r f o r a t o r a - perforator jest włączony do pracy w systemie;  
 w ł ą c z e n i e   c z y t n i k a - czytnik jest włączony do pracy w systemie;  
 c i ą g   d a l s z y   t a ś m y - dla danego problemu może być jeszcze wczytywana taśma papierowa.

Znaczenie poszczególnych lampek jest następujące (podana interpretacja dotyczy lampki zapalającej):

a l a r m - jedno lub kilka urządzeń zostało wyłączone z pracy w systemie, bądź też została zawieszona - programowo lub przez operatora - praca jednostki centralnej; przyczynę alarmu

można określić za pomocą monitora (patrz punkt 5.4.4);

k o n i e c   p r o b l e m u - problem został zakończony; jeśli nie ma alarmu, to należy zgłosić brak dalszej taśmy do czytania w zakończonym problemie;

w ł ą c z e n i e   p e r f o r a t o r a - perforator jest włączony do pracy w systemie.  
w ł ą c z e n i e   c z y t n i k a - czytnik jest włączony do pracy w systemie;

c i ą g   d a l s z y   t a ś m y - w aktualnie wykonywanym problemie może być jeszcze wczytywana taśma papierowa.

### 5.3.2. Znaczenie kluczy i lampek na modułach urządzeń zewnętrznych

W niniejszym punkcie opisane jest znaczenie kluczy NIEGOTÓW, GOTÓW, ZGŁOSZENIE OPERATORA oraz lampek NIEGOTÓW, GOTÓW, PRACA dla wszystkich modułów urządzeń zewnętrznych, za wyjątkiem jednostek taśmy magnetycznej (bęben nie jest w tym przypadku traktowany jako urządzenie zewnętrzne).

Omawiane urządzenia, jeśli zostało włączone ich zasilanie, mogą się znajdować w dwóch stanach: technicznej gotowości do pracy oraz braku tej gotowości. W celu włączenia technicznej gotowości urządzenia należy nacisnąć klucz GOTÓW (oznaczany również symbolem ⊙), natomiast w celu wyłączenia gotowości - klucz NIEGOTÓW (oznaczany symbolem ⊙). Istnienie lub brak technicznej gotowości jest odpowiednio sygnalizowane lampkami GOTÓW i NIEGOTÓW, które w niektórych urządzeniach są podświetlonymi kluczami.

Jeśli urządzenie jest w stanie GOTÓW, to może ono być podłączone do pracy w systemie lub odłączone. Podłączenie do pracy w systemie jest sygnalizowane zapaleniem lampki PRACA. Do zmiany tego stanu, tj. do podłączenia do systemu, jeśli urządzenie jest odłączone i odwrotnie - służy klucz ZGŁOSZENIE OPERATORA.

Naciśnięcie klucza ZGŁOSZENIE OPERATORA dla czytnika taśmy lub czytnika kart w stanie NIE-GOTÓW oznacza brak ciągu dalszego taśmy lub kart w tym problemie. Jest to więc odpowiednik wyłączenia klucza CIĄG DALSZY TAŚMY na stoliku operatora. Naciśnięcie klucza ZGŁOSZENIE OPERATORA dla innych urządzeń winno odbywać się w stanie GOTÓW.

### 5.3.3. Włączanie i wyłączanie urządzeń zewnętrznych

Włączanie i wyłączanie urządzeń zewnętrznych odbywa się w trzech etapach:

- a) włączenie lub wyłączenie zasilania; w zasadzie powinno być wykonywane przez konserwatora; wyjątek stanowią czytniki taśmy, które można włączyć lub wyłączyć za pomocą przyciśnięcia jednego klucza;
- b) włączenie lub wyłączenie gotowości technicznej; opisane ono jest w punkcie 5.3.2 (czynność ta nie dotyczy jednostek taśmy magnetycznej oraz czytnika i perforatora taśmy znajdujących się na stoliku operatora);
- c) włączenie do lub wyłączenie z pracy w systemie; opisane ono jest w punkcie 5.3.1 dla czytnika i perforatora taśmy znajdujących się na stoliku operatora oraz w punkcie 5.3.2 - dla pozostałych urządzeń zewnętrznych, oprócz jednostek taśmy magnetycznej.

Włączenie do pracy w systemie jednostki taśmy magnetycznej, po założeniu na nią szpuli z taśmą (co w zasadzie powinien robić konserwator), polega na wciśnięciu klucza M i odczekaniu aż zapali się lampka GOTÓW.

Dodatkowe uwagi dotyczące omawianych zagadnień są umieszczone w punkcie 5.4.

Włączanie zasilania i gotowości technicznej urządzeń oraz włączanie urządzeń wyjściowych (perforatory, drukarka) do pracy w systemie odbywa się dość rzadko, natomiast włączanie do pracy w systemie urządzeń wejścia (np. czytników taśmy) odbywa się często, czasami kilkakrotnie w problemie.

#### 5.3.4. Obsługa operatorska maszyny w czasie wykonywania problemów

Obsługa operatorska maszyny sprowadza się w zasadzie do wykonywania czynności wymienionych poniżej. Dodatkowe uwagi zawarte są w punkcie 5.4.

W poniższym opisie przez zestaw rozumie się zestaw 0, jeśli maszyna pracuje w reżymie 1-C, a gdy maszyna pracuje w reżymie 2-A lub 2-B - zestaw 0 lub 1, w zależności od tego, który z nich operator obsługuje.

- Przygotować zestaw do wprowadzenia problemu. Jeśli poprzedni problem został zakończony (lampka KONIEC PROBLEMU jest zapalona), wówczas należy doprowadzić do alarmu czytnika (tj. do zapalenia się lampki ALARM) przez zgłoszenie braku dalszego ciągu taśmy papierowej. W tym celu należy dla zestawu 0 wy-



cisnąć CIĄG DALSZY TAŚMY, nacisnąć ZGŁOSZENIE OPERATORA, po czym wcisnąć CIĄG DALSZY TAŚMY; dla zestawu 1 - nacisnąć NIEGOTÓW i ZGŁOSZENIE OPERATORA na stoliku czytelnika.

- Przygotować wprowadzanie danych. W tym celu należy włożyć karty do czytnika kart, taśmę do czytnika taśmy (oprócz czytnika, który będzie wczytywał problem), założyć szpule taśm magnetycznych na odpowiednie jednostki. Włączyć potrzebne urządzenia wejściowe do pracy w systemie (patrz punkty 5.3.2, 5.4.1 i 5.4.3).
- Przygotować urządzenia wyjściowe. Sprawdzić rodzaj (5 lub 8-kanalowa) oraz ilość taśmy papierowej w perforatorach, w razie potrzeby wymienić taśmę. Założyć szpule z roboczymi taśmami magnetycznymi na odpowiednie jednostki, ewentualnie założyć właściwy papier do drukarki. Włączyć urządzenia wyjściowe do pracy w systemie (o ile nie były włączone, patrz punkty 5.3.1, 5.3.2, 5.4.1 i 5.4.2).
- Wprowadzić problem. Założyć odpowiednią taśmę dziurkowaną do pierwszego czytnika w zestawie. Włączyć czytnik do pracy w systemie. W tym celu należy dla zestawu 0 wcisnąć klucz WŁĄCZENIE CZYTNIKA i nacisnąć ZGŁOSZENIE OPERATORA, a dla zestawu 1 - nacisnąć GOTÓW i ZGŁOSZENIE OPERATORA.
- Usuwać alarmy maszyny. W tym celu należy obserwować lampkę ALARM; po jej zapaleniu się - ustalić przyczynę alarmu (np. za pomocą monitora - patrz punkt 5.4.4). Jeśli alarm powstał z przyczyn technicznych (np. wyłączenie drukarki) - usunąć jego przy-

czynę. Jeśli alarm powstał na skutek przeczytania partii danych i żądania następnej partii - włożyć odpowiednie (zgodnie z instrukcją obsługi problemu) dane do czytnika i włączyć czytnik do pracy w systemie, bądź zasygnalizować brak dalszych danych (patrz punkt 5.3.2). Jeśli alarm powstał z przyczyn programowych - wykonać czynności opisane w instrukcji operatorskiej danego problemu.

- Obserwować pracę monitora. W razie potrzeby wprowadzić z monitora odpowiedni tekst, zgodnie z instrukcją operatorską problemu (patrz punkt 5.4.4).
- Wykonywać czynności przewidziane instrukcją operatorską. Zwykle wykonywanie tych czynności, takich jak zmiana szpuli na jednostce taśmy magnetycznej, wprowadzanie danych, wypisywanie tekstów na monitorze, zmiana papieru w drukarce itp. powinno być związane z alarmem maszyny.
- Obserwować pracę urządzeń. W razie nieprawidłowej pracy urządzenia (np. zerwanie taśmy papierowej w perforatorze) wyłączyć urządzenie z pracy w systemie, usunąć awarię, włączyć urządzenie do pracy w systemie. W razie potrzeby powtórzyć wykonanie całego problemu od początku.
- W razie konieczności wyrzucić program lub problem. W uzasadnionych przypadkach, określonych w instrukcji operatorskiej lub wynikających z doświadczenia operatora, należy wyrzucić program (wcisnąć klucz WYRZUCENIE PROGRAMU i nacisnąć ZGŁOSZENIE OPERATORA)

lub - rzadziej - problem (wcisnąć klucz WYRZUCENIE PROBLEMU i nacisnąć ZGŁOSZENIE OPERATORA). Wyrzucenie problemu powoduje natychmiastowe zakończenie całego problemu, natomiast po wyrzuceniu programu maszyna wykona następne kroki problemu (np. POST MORTEM).

- Zakończyć problem. Zdjąć szpule taśm magnetycznych, zwinąć taśmy papierowe z danymi i wynikami, oderwać, złożyć i opisać wyniki z drukarki itp. Niektóre z tych czynności mogą być wykonane po przetworzeniu większej liczby problemów.

#### 5.4. DODATKOWE UWAGI O OBSŁUDZE NIEKTÓRYCH URZĄDZEŃ

##### 5.4.1 Jednostki taśmy magnetycznej

Przy obsłudze jednostek taśmy magnetycznej należy m.in. pamiętać, że:

- a) w pomieszczeniach z jednostkami taśmy magnetycznej, które są z reguły klimatyzowane należy zachować szczególną czystość; palenie w tych pomieszczeniach tytoniu jest niedopuszczalne;
- b) nie wolno otwierać drzwiczek jednostki jeśli nie jest włączony przycisk WS;
- c) w celu zdjęcia szpuli należy odwinąć taśmę (programowo lub jednym z kluczy oznaczonych symbolem  $\rightarrow$  lub  $\rightleftarrows$ ), po czym należy nacisnąć klucz WS i koniecznie odczekać kilkadziesiąt sekund, aż zostaną zwolnione hamulce w jednostce; zdejmowanie taśmy bez odczekania na zwolnienie hamulców powoduje rozciąganie taśmy, co początkowo może być niezauważalne,

ale po kilkakrotnym powtórzeniu się powoduje trwałe zmniejszenie pewności zapisu;

- d) na większości szpul znajduje się uchwyt, w który powinna być włożona metryczka szpuli; w przypadku braku uchwytu metryczka powinna być przyklejona lepcem nie powodującym zanieczyszczeń mogących się przedostać do taśmy; metryczka powinna być zawsze aktualnie wypełniona, treść jej stanowić mogą np. następujące informacje:

- nr szpuli,
- dysponent szpuli,
- nazwa zbioru (zbiorów) na szpuli,
- data zapisu,
- termin ważności,
- nr jednostki taśmy, na której dokonano zapisu.

Ostatnia z wymienionych informacji jest zalecana ze względu na występującą czasami niewymienność jednostek PT-2.

#### 5.4.2. Drukarka wierszowa

Obsługa drukarki wierszowej DW-1 nie wymaga komentarza, natomiast przy obsłudze drukarki wierszowej DW-2 i DW-2E, omówienia wymaga jedynie sposób ustawiania strony. Aby ustawić właściwie stronę i zapewnić prawidłowe jej prowadzenie należy:

- a) sprawdzić, czy założono właściwą taśmę sterującą wysuwaniem strony oraz czy taśma ta nie jest podarta;
- b) jeśli lampka PRACA pali się, wówczas należy nacisnąć ZGŁOSZENIE OPERATORA (na drukarce) gasząc tym samym lampkę PRACA;

- c) nacisnąć klucz NIEGOTÓW;
- d) ustawić stronę naciskając odpowiednio klucze WYSUW WIERSZA i WYSUW STRONY; na ogół na drukarkach zaznaczono kreską położenie linii między stronami w chwili ustawienia strony;
- e) nacisnąć klucz GOTÓW i zaczekać na właściwe ustawienie się taśmy sterującej wysuwaniem strony; zostanie to zasygnalizowane zapaleniem się lampki podświetlającej klucz GOTÓW;
- f) nacisnąć klucz ZGŁOSZENIE OPERATORA w celu włączenia drukarki do pracy w systemie.

Tak ustawiona strona będzie zachowana tak długo, jak długo drukarka jest w stanie GOTÓW. Dopiero zapalenie lampki NIEGOTÓW powoduje konieczność ponownego ustawienia strony. Należy zwrócić uwagę, że ani naciskanie kluczy WYSUW WIERSZA lub WYSUW STRONY, ani ponowne startowanie systemu nie wymagają powtórnego ustawienia strony. Naciskanie kluczy WYSUW WIERSZA lub WYSUW STRONY powinno się odbywać przy drukarce wyłączonej z pracy w systemie, tj. przy zgaszonej lampce PRACA; w przeciwnym przypadku, gdyby maszyna rozpoczęła w tym czasie drukowanie, mogłyby nastąpić zakłócenia. Wszystkie powyższe uwagi mają zastosowanie tylko wówczas, gdy program właściwie korzysta z rozkazu ustawienia strony. Programy, w czasie działania których operator musi specjalnie ustawiać stronę w sposób nietypowy powinny być uznane za niewłaściwie wykonane i odpowiednio poprawione; można to na ogół zrobić niewielkim nakładem pracy.

### 5.4.3. Czytnik kart

Włożenie do czytnika kart paczki kart jest związane z wykonaniem kolejno następujących czynności:

- a) sprawdzić czy czytnik jest wyłączony z pracy w systemie, tj. czy lampka PRACA jest zgaszona; jeśli nie - nacisnąć ZGŁOSZENIE OPERATORA,
- b) nacisnąć klucz NIEGOTÓW,
- c) włożyć karty do czytnika,
- d) wprowadzić jedną kartę naciskając klucze POJED START lub SINGLE SHOT,
- e) nacisnąć klucz GOTÓW oraz ZGŁOSZENIE OPERATORA.

W celu dołożenia kart wystarczy wyłączyć czytnik z pracy w systemie (naciskając ZGŁOSZENIE OPERATORA), dołożyć karty do kart już znajdujących się w czytniku i włączyć czytnik do pracy w systemie (przez ponowne naciśnięcie ZGŁOSZENIA OPERATORA).

Zgłoszenie braku dalszych kart do wczytywania powinno się odbywać zgodnie z opisem podanym w punkcie 5.3.2.

### 5.4.4. Monitor dalekopisowy

Monitor dalekopisowy może służyć jako urządzenie wejścia i wyjścia do:

- a) wypisania tekstu przez maszynę,
- b) wprowadzenia tekstu przez operatora na żądanie maszyny,
- c) wprowadzania przez operatora pytania lub żądania,

d) odpowiedzi maszyny na pytanie lub żądanie operatora.

ad a) Wypisanie tekstu przez maszynę. Wypisywany tekst jest zawsze rozpoczynany od znaków  $N;$ , gdzie  $N$  jest numerem zestawu (tzn.  $N=0$  lub  $1$ ). Dalej wypisany tekst stanowi zazwyczaj informacje dla operatora. W czasie wypisywania całego tekstu pali się lampka OPERACJA PISANIE.

ad b) Wprowadzenie tekstu przez operatora na żądanie maszyny. Maszyna żądając wprowadzenia przez operatora tekstu wypisuje znaki  $N;$ , gdzie  $N$  jest numerem zestawu (tzn.  $N=0$  lub  $1$ ). Po wypisaniu przez maszynę tego tekstu operator powinien szybko wprowadzić odpowiedni tekst; w przeciwnym przypadku maszyna uzna, że tekst nie będzie wprowadzany. Tekst pisze się zgodnie z punktem 5.1.2 lub 5.1.3 (jeśli odbywa się to w trakcie startowania  $S0$ ), lub zgodnie z instrukcją operatorską aktualnie przetwarzanego problemu. Tekst nie powinien przekraczać 93 znaków i musi być zakończony znakiem  $*$  w tym samym wierszu dalekopisowym.

Jeśli w trakcie wprowadzania tekstu operator pomyli się, to może skasować pomyłkę wprowadzając ponownie całą informację od nowego wiersza.

W czasie wypisywania przez maszynę znaków  $N;$  pali się lampka OPERACJA PISANIE, a następnie lampka OPERACJA CZYTANIE.

ad c) Wprowadzanie przez operatora pytania lub żądania. W celu wprowadzenia pytania lub żądania należy:

- nacisnąć klucz ZGŁ.WYŁĄCZ i poczekać aż zapali się lampka OPERACJA CZYTANIE;
- napisać na monitorze znaki N- ,gdzie N jest numerem zestawu (tzn. N=0 lub 1; jeśli żądanie nie dotyczy żadnego zestawu należy przyjąć N=0);
- napisać odpowiednie pytanie lub żądanie, kończąc tekst znakiem \* ; wykaz dopuszczalnych pytań i żądań oraz ich znaczenie podane jest w punkcie 5.5. Jeśli w trakcie pisania operator. pomyli się, może skasować pomyłkę wprowadzając ponownie cały tekst (ale bez znaków N- ) od nowego wiersza.

ad d) Odpowiedź maszyny na pytanie lub żądanie operatora. Maszyna odpowiadając na pytanie operatora lub potwierdzając jego żądanie wypisuje odpowiedni tekst, jak to podano w punkcie 5.5; jeśli pytanie lub żądanie dotyczyło określonego zestawu, to tekst jest poprzedzony numerem zestawu N oraz spacją.

W czasie wypisywania odpowiedzi pali się lampka OPERACJA PISANIE.

## 5.5. TEKSTY PRZY WSPÓŁPRACY Z MONITOREM DALEKOPISOWYM

### 5.5.1. Teksty dotyczące określonego zestawu

Wszystkie wprowadzane teksty powinny być poprzedzone znakami N- (N jest numerem zestawu, tzn. N=0 lub 1) i zakończone znakiem \* .Teksty odpowiedzi są poprzedzone numerem zestawu N i spacją.



Tekst wprowadzany	Objaśnienie	Tekst odpowiedzi
CO*	określ urządzenie, które stanowi przyczynę alarmu	ALARM ident. urzędzenia lub ALARM CENTRM lub NIC
W-k-b*	wpisz do k-tego bitu trzeciego słowa zestawu wartość b (b=0 lub b=1; $0 \leq k \leq 23$ )	W= wartość trzeciego słowa binarnie
Z*	zawieś (zatrzymaj) pracę zestawu	Z
S*	likwiduj zawieszenie pracy zestawu	S

### 5.5.2. Teksty nie dotyczące zestawu

Wszystkie wprowadzane teksty powinny być poprzedzane znakami 0- i zakończone znakiem \* .

Tekst wprowadzany	Objaśnienie	Tekst odpowiedzi
B-adres *	podaj oktalnie z bębna zawartość słowa o podanym dziesiętnie adresie	Badres-zawartość
P-adres *	podaj oktalnie z PAO zawartość słowa o podanym dziesiętnie adresie	Padres-zawartość
K-wartość *	wpisz wartość jako liczbę ósemkową do zmiennej K	K-wartość
	dany dziesiętnie adresie	
K-wartość *	wpisz wartość jako liczbę ósemkową do zmiennej K	K-wartość

Tekst wprowadzany	Objaśnienie	Tekst odpowiedzi
WB-adres*	do słowa na bębnie o podanym dziesiętnie adresie wpisz wartość zmiennej K	Badres=wartość
WP-adres*	do słowa w PAO o po- danym dziesiętnie adresie wpisz war- tość K	Padres=wartość
XD-oktalna*	przelicz podaną liczbę oktalną na dziesiętną	Xoktalna=Ddziesiętna
DX-dzie- siętna*	przelicz podaną liczbę dziesiętną na oktalną	Ddziesiętna=Xoktalna
siętna* dziesiętną na oktalną		

## 6. URZĄDZENIA ZEWNĘTRZNE I KODY

### 6.1. WYKAZ URZĄDZEŃ ZEWNĘTRZNYCH I ICH NUMERÓW

Poniższa tabelka podaje zestawienie urządzeń zewnętrznych maszyny oraz symbole tych urządzeń.

<u>Urządzenie</u>	<u>Symbol</u>
czytnik taśmy papierowej	CP
perforator taśmy papierowej	DP
drukarka wierszowa	DW
bęben systemu	BSS
bęben podstawowy (bęben użytkownika)	BSM
taśma magnetyczna	TM
czytnik kart	CK

Niezależnie od liczby bębnow oraz liczby słów w pamięci bębnowej, bęben systemu jest traktowany jako jedno urządzenie; podobnie bęben podstawowy jest też zawsze jednym urządzeniem. Oprócz BBS i BEM do maszyny można podłączyć po kilka urządzeń zewnętrznych jednego typu; symbole tych urządzeń są zawsze uzupełniane kolejnymi literami alfabetu, np.

czytnik taśmy papierowej 1 - CPA  
 czytnik taśmy papierowej 2 - CPB  
 itd.

Do każdego urządzenia przypisany jest numer (lub dwa numery, jeśli urządzenie służy do zapisu i odczytu informacji), wykorzystywany w rozkazach dotyczących współpracy z tym urządzeniem. Numery te są określone poniższą tabelką (w tabelce podano również uprzednio używane i nie zalecane do stosowania symbole urządzeń):

<u>symbol</u> <u>urządzenia</u>	<u>numer urządzenia (uz)</u> <u>pisanie</u>	<u>czytanie</u>	<u>stare</u> <u>symbole</u>
CPA		1	CTP1
CPB		3	CTP2
DPA	0		PTP1
DPB	2		PTP2
DWA	4		DW
BBS		5	BEBS
BBM	6	7	BEBP, BEBC
TMA	8	9	TP11, TC11
TMB	10	11	TP12, TC12
TMC	12	13	TP13, TC13
TMD	14	15	TP21, TC21
TME	16	17	TP22, TC22
TMF	18	19	TP23, TC23
CKA		21	CK

Numery innych urządzeń dołączanych do maszyny mogą być różne w różnych edycjach systemu operacyjnego.

Niżej wymienione urządzenia specjalne mają dodatkowy numer us, określony następująco:

Symbol <u>urządzenia</u>	numer urządzenia (us)		Uwagi
	<u>pisanie</u>	<u>czytanie</u>	
TMS	0	1	taśma ma- gnetyczna systemu
BBS	4	5	bęben magn. systemu
BBM	6	7	bęben magn. użytkownika

Sposób odwołania do urządzeń przez numer uz lub us podano w opisach odpowiednich rozkazów (por. punkty 3.4.5).

## 6.2. KOD KW-6

		00	01	10	11		
X <sub>1</sub> X <sub>2</sub> X <sub>3</sub> X <sub>4</sub> X <sub>5</sub> X <sub>6</sub>		0	16	32	48		
0000	0	NU NULL	IT <	XL L	Z0	0	
0001	1	SP SPace	GT >	XM M	Z1	1	
0010	2	FP .	QS ?	XN N	Z2	2	
0011	3	LP (	ES EScape	XO O	Z3	3	
0100	4	PL +	SO Shift Out	XP P	Z4	4	
0101	5	AS *	XA A	XQ Q	Z5	5	
0110	6	RP )	XB B	XR R	Z6	6	
0111	7	SE ;	XC C	XS S	Z7	7	
1000	8	MI -	XD D	XT T	Z8	8	
1001	9	SL /	XE E	XU U	Z9	9	
1010	10	CM ,	XF F	XV V	LF Line Feed		
1011	11	CL :	XG G	XW W	N6	◊	
1100	12	AP ' ,	XH H	XX X	FF Form Feed		
1101	13	EQ =	XI I	XY Y	CS	⋈	
1110	14	LB [	XJ J	XZ Z	CR Carrage Return		
1111	15	RB ]	XK K	SI Shift In	DE Delete		

6.3. KOD KW-6

		0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111
		0	16	32	48	64	80	96	112	128	144	160	176	192	208	224	240
	X <sub>1</sub> X <sub>2</sub> X <sub>3</sub> X <sub>4</sub> X <sub>5</sub> X <sub>6</sub> X <sub>7</sub> X <sub>8</sub>																
0000	0	NU Null	BS Back Space	DL Data Link escape	CW Cancel	SP Space	LP (	Z0 0	Z8 8	AT @	XH H	XP P	XX Y	N3	YH h	YP p	YX x
0010	2	SH Start of Heading	HT Hori- zontal Tabula- tion	D1 Device control 1	EM End of Medium	EX I	RP )	Z1 1	Z9 9	XA A	XI I	XC Q	XY Y	YA a	YI i	YQ q	YY y
0100	4	ST Start of Text	LF Line Feed	D2 Device control 2	SU Substi- tute	QT "	AS *	Z2 2	OL :	XB B	XJ J	XR R	XZ z	YB b	YJ j	YR r	YZ z
0110	6	ET End of Text	VT Verti- cal Ta- bula- tion	D3 Device control 3	ES EScape	NR #	PL +	Z3 3	SE ;	XC C	XK K	XS S	LB [	YC c	YK k	YS s	N4
1000	8	EN End of trans- mission	FP Form Feed	D4 Device control 4	FS File Sepa- rator	CS %	CW ,	Z4 4	LT <	XD D	XL L	XT T	N1 z	YD d	YL l	YT t	N5 z
1010	10	EY EnquirY	CR Carri- age Return	NA Negative Acknow- ledge	GS Group Sepa- rator	PC %	MI -	Z5 5	EQ =	XE E	XM M	XU U	RB ]	YE e	YM m	YU u	N6 ◊
1100	12	AC Acknow- ledge	SO Shift Out	SY Synchro- nous Idle	RS Record Sepa- rator	AN &	PP .	Z6 6	GT >	XF F	XN N	XV V	N2	YF f	YN n	YV v	N7
1110	14	BE BEll	SI Shift In	EB End of trans- mission Block	US Unit Separa- tor	AP '	SL /	Z7 7	QS ?	XG G	XO O	XW W	UH -	YG g	YO o	YW w	TE LE.t.

Uwaga: wartości nieparzyste odpowiadają pochodnym znaków o wartościach parzystych

6.4. WARTOŚCI WYBRANYCH ZNAKÓW W KW-6  
I KW-8

KW-6		ZNAK	KW-8	
DZIESIĘT- NIE	OKTAL- NIE		DZIESIĘT- NIE	OKTAL- NIE
1	1	SP	64	100
2	2	.	92	134
3	3	(	80	120
4	4	+	86	126
5	5	*	84	124
6	6	)	82	122
7	7	;	118	166
8	10	-	90	132
9	11	/	94	136
10	12	,	88	130
11	13	:	116	164
12	14	'	78	116
13	15	=	122	172
14	16	[	182	266
15	17	]	186	272
16	20	<	120	170
17	21	>	124	174
18	22	?	126	176
21	25	A	130	202
22	26	B	132	204
23	27	C	134	206
24	30	D	136	210
25	31	E	138	212
26	32	F	140	214
27	33	G	142	216
28	34	H	144	220
29	35	I	146	222
30	36	J	148	224
31	37	K	150	226



KW-6		ZNAK	KW-8	
DZIESIĘT- NIE	OKTAL- NIE		DZIESIĘT- NIE	OKTAL- NIE
32	40	L	152	230
33	41	M	154	232
34	42	N	156	234
35	43	O	158	236
36	44	P	160	240
37	45	Q	162	242
38	46	R	164	244
39	47	S	166	246
40	50	T	168	250
41	51	U	170	252
42	52	V	172	254
43	53	W	174	256
44	54	X	176	260
45	55	Y	178	262
46	56	Z	180	264
48	60	0	96	140
49	61	1	98	142
50	62	2	100	144
51	63	3	102	146
52	64	4	104	150
53	65	5	106	152
54	66	6	108	154
55	67	7	110	156
56	70	8	112	160
57	71	9	114	162
58	72	LF	20	24
59	73	◇	250	372
60	74	FF	24	30
62	76	CR	26	32

## 6.5. SPIS PODPROGRAMÓW WEJŚCIA I WYJŚCIA

## 6.5.1. Podprogramy wejścia

<u>Symbol</u>	<u>długość</u>	<u>nazwa podprogramu</u>
PPUS	2	pusty
PWE	11	wejściowy wewnętrzny
PCTP	128	czytania taśmy papierowej
PCKP	119	czytania kart Holleritha (prosty)
PCKA	104	czytania kart Aritma (prosty)
PCK	356	czytania kart perforowanych
PCM	83	czytania monitora
PCT8	233	czytania znaków 8-bitowych z taśmy magnetycznej
PCT6	282	czytania znaków 6-bitowych z taśmy magnetycznej (selek- tywny)
PCTM	287	czytania znaków 6-bitowych z taśmy magnetycznej

## 6.5.2. Podprogramy wyjścia

<u>Symbol</u>	<u>długość</u>	<u>nazwa podprogramu</u>
PPUS	2	pusty
PWY	11	wyjściowy wewnętrzny
PPTP	149	pisania taśmy papierowej
PDW	149	wyjścia na drukarkę DW-1
PDV	149	wyjścia na drukarkę DW-2
PPM	102	pisania monitora
PPT8	223	pisania znaków 8-bitowych na taśmę magnetyczną

## 6.6. OPIS PODPROGRAMÓW WEJŚCIA I WYJŚCIA<sup>1)</sup>

### 6.6.1. Podprogramy wejścia

Funkcją podprogramu wejścia jest dostarczenie programowi wywołującemu kolejnego kwantu informacji pobranego z jakiegoś źródła wejściowego. Informacje pobierane ze źródła mogą przy tym ulegać przekształceniu dokonywanemu przez dekodery związany z danym podprogramem w ramach symbolicznego wejścia. W takim przypadku dekodery są podprogramami podprogramu wejścia.

Opis każdego podprogramu wejścia składa się z następujących elementów:

- charakterystyka źródła, z którego podprogram pobiera informacje,
- lokalizacja i postać kwantu informacji dostarczanego przez podprogram,
- możliwości współpracy z dekodernami,
- organizacja pracy podprogramu, a w szczególności liczba wykorzystywanych numerów przesłań.

Wyróżnioną grupę podprogramów wejścia stanowią tzw. proste podprogramy wejścia; kwantem dostarczanej dla nich informacji jest jeden znak umieszczony w B-rejestrze.

Podprogramy, które mogą współpracować z dekodernami nazywają się dekodującymi.

Wyróżnionym podprogramem w grupie podprogramów wejścia jest podprogram pusty o nazwie PPUS, który nie wykonuje żadnej czynności.

<sup>1)</sup> Ten punkt w całości opracowali: Krzysztof Bytnerowicz, Zbigniew Kosowski, Marian Skupiński.

Poniżej podana jest charakterystyka poszczególnych podprogramów wejścia.

- a) Podprogram pusty - PPUS:  
 źródło informacji - brak;  
 kwant informacji - nie dostarcza żadnych informacji;  
 podprogram niedekodujący;  
 organizacja pracy: natychmiastowy powrót do programu wywołującego; podprogram nie wykorzystuje przesłań zewnętrznych.
- b) Podprogram wejściowy - PWE:  
 źródło informacji - zawartość B-rejestru;  
 kwant informacji - znak 6 lub 8-bitowy, niezmienną zawartość B-rejestru, lub 0 w B-rejestrze w zależności od dekodera, z którym podprogram współpracuje (podprogram prosty);  
 podprogram dekodujący;  
 organizacja pracy: działanie polega jedynie na wykorzystaniu dekodera do przekodowania zawartości B-rejestru. Podprogram nie definiuje przesłań zewnętrznych.
- c) Podprogram czytania taśmy papierowej - PCTP:  
 źródło informacji - perforowana taśma papierowa czytana przez czytnik taśmy papierowej CP;  
 kwant informacji - 1 znak w B-rejestrze (podprogram prosty); po sygnale końca danych z czytnika CP -0 w B-rejestrze;  
 podprogram dekodujący;  
 organizacja pracy: obsługa dwóch buforów po 32 słowa każdy; wykorzystuje 2 numery przesłań.

d) Podprogram czytania kart Holleritha (prosty) - PCKP:

źródło informacji - karty perforowane czytane przez czytnik kart CK;

kwant informacji - 1 znak w B-rejestrze (podprogram prosty); w przypadku gdy dekodery jest dekodery jednostkowy (DJED) - kwantem jest kolejna kolumna karty wpisana do B-rejestru na pozycje 0-11 tak, że perforacja wiersza 12 karty odpowiada pozycji 0; pozycje 12-23 B-rejestru zostają wyzerowane; po sygnale końca danych z CK kwantem podawanej do B-rejestru informacji jest stale -0;

podprogram dekodujący;

organizacja pracy - obsługa 1 buforu (81 słów) na jedną kartę; czytnik CK zatrzymuje się po przeczytaniu każdej karty do buforu; wykorzystuje 1 numer przesłania.

e) Podprogram czytania kart perforowanych Aritma (prosty) - PCKA:

źródło informacji - karty perforowane 90-kolumnowe czytane przez czytnik kart;

kwant informacji - 1 znak na najmłodszych pozycjach B-rejestru (podprogram prosty); w przypadku, gdy dekodery jest dekodery jednostkowy (DJED), kwantem jest kolejna kolumna karty umieszczona na pozycjach 0-5 B-rejestru; pozycje 6-23 B-rejestru zostają wyzerowane. Znaki z karty są podawane najpierw z górnej połowy, a następnie z dolnej; znak końca karty (po przekodowaniu)

zostaje podany po ostatnim znaku dolnej połowy karty; po sygnale końca danych z CK kwantem podawanej informacji jest wartość -O w B-rejestrze;

podprogram dekodujący;

organizacja pracy - obsługa 1 buforu (46 słów) na jedną kartę; CK zatrzymuje się po przeczytaniu każdej karty; wykorzystuje 1 numer przesłania.

f) Podprogram czytania kart perforowanych - PCK:

źródło informacji - karty perforowane czytane przez czytnik kart;

kwant informacji - pole w pamięci operacyjnej, w którego słowach na najmłodszych bitach znajdują się pojedyncze znaki z kolejnych kolumn karty; adres pola jest określany przez zawartość B-rejestru przy wejściu do podprogramu; po wczytaniu karty w rejestrach zostają umieszczone następujące informacje:

rejestr A - maksymalna wartość przekodowanego znaku (w sensie rozkazu POB),

rejestr M - minimalna wartość przekodowanego znaku (w sensie rozkazu POB),

rejestr B - liczba przeczytanych kolumn; po sygnale końca danych z CK kwantem informacji jest -Q podawane do B-rejestru.

podprogram dekodujący;

organizacja pracy - obsługa 3 buforów wykorzystywanych równocześnie, umożliwia ciągłą pracę CK; wykorzystuje 3 numery przesłań.

- g.) Podprogram czytania z monitora - PCM:  
 źródło informacji - klawiatura monitora dalekopisowego;  
 kwant informacji - 1 znak w B-rejestrze (podprogram prosty);  
 podprogram dekodujący;  
 organizacja pracy - obsługa 1 bufora (32 słowa) zapełnianego za pomocą rozkazu SDY - CZYTAJ MONITOR; po przeczytaniu jednego wiersza do bufora znaki zaczynają być kolejno przekazywane (po ewentualnym zdekodowaniu) do programu wywołującego; po przekazaniu ostatniego znaku wiersza (znaku LF) - przy kolejnym odwołaniu do podprogramu - ponownie zostaje wykonany rozkaz SDY - CZYTAJ MONITOR zapełniający bufor od początku; ograniczenie długości wiersza na monitorze wynika z wielkości buforu i właściwości rozkazu SDY - CZYTAJ MONITOR.  
 Podprogram nie definiuje przesłań zewnętrznych.
- h.) Podprogram czytania znaków 8-bitowych z TM - PCT8:  
 źródło informacji - ciąg znaków 8-bitowych zapisanych na TM wg standardu SMAO; taśma magnetyczna zapisana wg standardu SMAO stanowi plik bądź ciąg plików o strukturze zgodnej ze standardem ZAM 41 i o blokach informacyjnych długości 128 słów;  
 kwant informacji - 1 znak 8-bitowy w B-rejestrze; w przypadku odczytania MKZ -0 w B-rejestrze (podprogram prosty);  
 podprogram dekodujący;

organizacja pracy - obsługa jednego buforu (128 słów), do którego czytane są kolejne bloki z TM aż do napotkania MKZ; podprogram PCT8 pomija bloki organizacyjne z wyjątkiem MKZ, a kolejne znaki pobiera z bloków informacyjnych poczynając od 7 słowa bloku informacyjnego; każde słowo zawiera 3 kolejne znaki 8-bitowe umieszczone na pozycjach 0-7, 8-15 i 16-23. Podprogram sprawdza kolejność czytanych bloków oraz sumę kontrolną bloku i w przypadku wykrycia nieprawidłowości (zła suma lub zła kolejność) powoduje wypisanie na monitorze tekstu 'ZLA TM' oraz wykonanie rozkazu STO 502. W przypadku 8-krotnego wykrycia błędu parzystości sygnalizowanego przez jednostkę pamięci taśmowej podprogram wypisuje na monitorze tekst 'BL TM' oraz ponawia próby odczytania bloku. Podprogram definiuje jedno przesłanie zewnętrzne.

- 1) Podprogram czytania znaków 6-bitowych z TM (selektywny) - PCT6:

źródło informacji - ciąg znaków 6-bitowych zbitych po 4 w słowo i zapisanych na TM wg standardu SMAD; taśma magnetyczna zapisana wg standardu SMAD stanowi ciąg bloków o długości 203 słowa każdy, stanowiących dokument SMAD i zakończonych blokiem sygnalizującym koniec dokumentu; informacja zawarta jest w słowach 1-200; słowo 0 zawiera informację o rodzaju bloku (0 - kolejny blok strony, 1 początek nowej strony, 2 koniec dokumentu), a słowa



201-202 zawierają sumę słów 0-200; w bloku zawierającym 1 w słowie 0 słowa 1-10 traktowane są jako nagłówek strony SMAD; 4 kolejne znaki umieszczone są na pozycjach 0-5, 6-11, 12-17, 18-23;

kwant informacji - 1 znak 6-bitowy w B-rejestrze; w przypadku odczytania końca dokumentu -0 w B-rejestrze (podprogram prosty);

podprogram niedekodujący;

organizacja pracy - obsługa jednego buforu (203 słowa), do którego czytane są kolejne bloki z TM aż do napotkania końca dokumentu; podprogram podaje znaki 6-bitowe (od lewej) kolejnych słów bloku aż do ich wyczerpania, poczynając od 1 słowa w przypadku bloku zwykłego oraz poczynając od 11 słowa dla bloku rozpoczynającego nową stronę SMAD; w przypadku 7-krotnego kolejnego wykrycia błędu parzystości lub niezgodności sumy kontrolnej, podprogram wypisuje na monitorze tekst 'BL TM', a następnie ponawia próby odczytania bloku; podprogram definiuje jedno przesłanie zewnętrzne.

j) Podprogram czytania znaków 6-bitowych z TM - PCTM:

źródło informacji - ciąg znaków 6-bitowych zbitych po 4 w słowo i zapisanych na TM w standardzie SMAD (patrz punkt i);

kwant informacji - 1 znak 6-bitowy w B-rejestrze; w przypadku odczytania końca dokumentu -0 w B-rejestrze (podprogram prosty);

podprogram niedekodujący ;  
 organizacja pracy - jak PCT6 z tą różnicą,  
 że podprogram podaje nagłówek strony  
 SMAD.

### 6.6.2. Podprogramy wyjścia

Funkcją podprogramu wyjścia jest pobranie od programu wywołującego, kolejnego kwantu informacji i dostarczenie go do odbiornika wyjściowego. Dekoder związany z danym podprogramem w ramach symbolicznego wyjścia może przekształcać informacje podawane do odbiornika. W takim przypadku dekodek jest podprogramem podprogramu wyjścia.

Opis każdego podprogramu wyjścia składa się z następujących elementów:

- charakterystyka odbiornika, do którego podprogram podaje informacje,
- lokalizacja i postać kwantu informacji pobieranego przez podprogram,
- możliwość współpracy z dekodekami,
- organizacja pracy podprogramu, a w szczególności - liczba wykorzystywanych numerów przesłań.

Wyróżnioną grupę podprogramów wyjścia stanowią tzw. proste podprogramy wyjścia, u których kwantem pobieranej informacji jest jeden znak umieszczony w B-rejestrze.

Wyróżnionym podprogramem w grupie podprogramów wyjścia jest podprogram pusty o nazwie PPUS, który nie wykonuje żadnej czynności.

Poniżej podana jest charakterystyka poszczególnych podprogramów wyjściowych.

- a) Podprogram pusty - PPUS;  
 odbiornik informacji - brak;  
 kwant informacji - nie pobiera żadnych informacji (nie zmienia wartości B-rejestru);  
 podprogram niedekodujący;  
 organizacja pracy - natychmiastowy powrót do programu wywołującego. Podprogram nie wykorzystuje przesłań zewnętrznych.
- b) Podprogram dekodujący wyjściowy - PWY;  
 odbiornik informacji - B-rejestr;  
 kwant informacji - znak umieszczany w B-rejestrze; w przypadku współpracy z dekodorem sklejającym, podprogram może żądać podania następnego znaku i wówczas w B-rejestrze zostaje umieszczona wartość 0; przy współpracy z dekodorem rozklejającym podprogram może w miejsce 1 znaku wydać ciąg znaków; w tym przypadku znaki będą podawane do B-rejestru pojedynczo z dodanym bitem na pozycji 0 B-rejestru (minus), z wyjątkiem ostatniego znaku ciągu;  
 podprogram dekodujący;  
 organizacja pracy - podprogram wykorzystuje dekodery do zmiany zawartości B-rejestru. Nie definiuje przesłań zewnętrznych.
- c) Podprogram pisania taśmy papierowej - PPTP;  
 odbiornik informacji - perforator taśmy papierowej;  
 kwant informacji - 1 znak w B-rejestrze (podprogram prosty);  
 podprogram dekodujący;  
 organizacja pracy - podprogram obsługuje 2 bufory po 32 słowa. Wykorzystuje 2 numery przesłań.

- d) Podprogram wyjścia na drukarkę DW-1 - PDW:  
 odbiornik informacji - drukarka wierszowa  
 typu DW-1;  
 kwant informacji - 1 znak w B-rejestrze  
 (podprogram prosty);  
 podprogram dekodujący;  
 organizacja pracy - obsługa 1 buforu (42  
 słowa) wypisywanego na drukarkę wierszową jako kolejny wiersz; znaki wpisywane są do buforu po 3 w słowo, poczynając od 3 słowa; dwa pierwsze słowa, ustawiane przez podprogram zawierają informacje sterujące pracą drukarki; podprogram definiuje jedno przesłanie zewnętrzne.
- e) Podprogram wyjścia na drukarkę DW-2 i DW-2E - PDV:  
 odbiornik informacji - drukarka wierszowa  
 typu DW-2 lub typu DW-2E;  
 kwant informacji - 1 znak w B-rejestrze  
 (podprogram prosty);  
 podprogram dekodujący;  
 organizacja pracy - identyczna jak dla podprogramu PDW (patrz punkt d).
- f) Podprogram pisania na monitor - PPM:  
 odbiornik informacji - monitor dalekopisowy;  
 kwant informacji - 1 znak w B-rejestrze  
 (podprogram prosty);  
 podprogram dekodujący ;  
 organizacja pracy - obsługa 1 buforu  
 (23 słowa) wypisywanego za pomocą rozkazu SDY - PISZ NA MONITOR - gdy bufor jest wypełniony (66 znaków) lub gdy znakiem do wypisania jest znak LF;  
 w przeciwnym przypadku następuje wpi-

sanie znaku (po ewentualnym zdekodowaniu) do buforu po 3 znaki w słowie i powrót do programu wywołującego. Powyższe ograniczenia wynikają z właściwości rozkazu SDY - PISZ NA MONITOR. Podprogram nie definiuje przesłań zewnętrznych.

g) Podprogram pisania znaków 8-bitowych na TM - PPT8:

odbiornik informacji - TM zapisywana ciągiem znaków 8-bitowych wg standardu SMAO (patrz punkt 6.6.1.h);

kwant informacji - 1 znak 8-bitowy w B-rejestrze (podprogram prosty);

podprogram dekodujący;

organizacja pracy - obsługa 1 buforu (128 słów) wypisywanego na TM jako kolejny blok. Znaki wpisywane są do buforu po 3 w słowie, poczynając od 7 słowa. Po wypełnieniu buforu podprogram ustawia kolejny numer bloku, liczy sumę kontrolną i pisze blok na TM. W przypadku błędnego przesłania podprogram cofa, kasuje i pisze jeszcze raz ten blok, a w przypadku napotkania końca TM - wykonuje rozkaz STO 501.

Podprogram definiuje jedno przesłanie zewnętrzne.

## 6.7. SPIS DEKODERÓW

Gwiazdką zaznaczono dekodery opisane w tablicach 6.8, 6.9, 6.10 i 6.11.

## 6.7.1. Dekodery wejściowe na KW6

<u>Symbol</u>	<u>długość</u>	<u>nośnik i kod</u>
*DM2C	54	taśma, M2
*DFEC	54	taśma, Ferranti (nietypowy)
*DFE6	54	taśma, Ferranti (standardowy)
*DOL6	59	taśma, Olivetti
*DBUL	63	karty, BULL
DBL6	152	karty, BULL2
DARN	58	karty, Aritma numeryczny
DARA	87	karty, Aritma alfanumeryczny
DAR6	151	karty, Aritma
*DIBM	177	karty, IBM
DSAM	113	karty, SAM
*DS46	170	karty, S4
*D86C	77	kod wewnętrzny KW8

## 6.7.2. Dekodery wyjściowe z KW6

<u>Symbol</u>	<u>długość</u>	<u>nośnik i kod</u>
*DM2P	95	taśma, M2
*DFEP	100	taśma, Ferranti (nietypowy)
*D6FE	100	taśma, Ferranti (standardowy)
*D6OL	65	taśma, Olivetti
DDW1	79	drukarka DW-1 (prosty)
DDW2	112	drukarka DW-1 (złożony)
*DDV1	79	drukarka DW-2 (prosty)
*DDV2	112	drukarka DW-2 (złożony)
*DDU1	79	drukarka DW-2E (prosty)
*DDU2	112	drukarka DW-2E (złożony)
*D68P	54	kod wewnętrzny KW8

## 6.7.3. Dekodery wejściowe na KW8

<u>Symbol</u>	<u>długość</u>	<u>nośnik i kod</u>
*DM28	51	taśma, M2
*DFE8	53	taśma, Ferranti
*DOP8	170	taśma, Optima
*DOL8	8	taśma, Olivetti
*DBU8	75	karty, BULL
*DBL8	135	karty, BULL2
DAN8	58	karty, Aritma numeryczny
*DIL8	132	karty, ICL
DIB8	132	karty, IBM
DSA8	115	karty, SAM
*D68C	54	kod wewnętrzny KW6
D8R8	268	kod wewnętrzny KW8R

## 6.7.4. Dekodery wyjściowe z KW8

<u>Symbol</u>	<u>długość</u>	<u>nośnik i kod</u>
*D8M2	112	taśma, M2
*D8OP	169	taśma, Optima
*D8OL	55	taśma, Olivetti
D8SE	169	taśma, Soemtron
DDW8	146	drukarka DW-1
*DDV8	146	drukarka DW-2
*DDU8	146	drukarka DW-2E
*D86P	77	kod wewnętrzny KW6
D88R	179	kod wewnętrzny KW8R

## 6.7.5. Inne dekodery

<u>Symbol</u>	<u>długość</u>	<u>opis</u>
DPUS	3	dekoder pusty
DJED	10	dekoder jednostkowy





	KW-8 D86C	BULL DBUL	IBM DIEM	S4 L846	OLIV. DOL6	FERRANTI DFB5	M-2 DM2C	KW-5 ZNAK WART.		
	?			?	?	?		?	18	
	◇		◇	◇	{125}	x,0	v, ]	◇. □	◇ 59	
	□			□	β, α	£	ZEST. 8	+	□ 61	
	ES				ES			ES	19	
					80			80	20	
								SI	47	
	ZEST. 2							NU	0	
	ZEST. 3				ZEST. 6			DE	63	
			*		ZEST. 7	DE	DE	NU	POMIJANE	
					EDNEC DANEC DANYCH	EDNEC DANEC DANYCH	EDNEC DANEC DANYCH	EDNEC DANEC DANYCH	EDNEC DANEC DANYCH	-0

## UWAGI DO TABLICY 6.8

- A-Z, a-z i 0-9 oznaczają wszystkie kolejne duże i małe litery alfabetu łacińskiego oraz cyfry. Natomiast Ą-Ż i ą-ż oznaczają odpowiednio ą, ć, ę, ł, ń, ó, ś, ź, ż oraz ą, ć, ę, ł, ń, ó, ś, ź, ż.
- Przez {n} oznaczano znak o wartości n, nie posiadający w danym kodzie reprezentacji graficznej.
- Podanie dwóch lub więcej znaków, np. x, ◇ oznacza, że albo te znaki mają jednakową reprezentację binarną (różne oznaczenia na różnych urządzeniach), albo mają różne reprezentacje, jednakowo interpretowane przez dekodery.
- Dekoder DOL6 nie kontroluje bitów parzystości.
- Dekoder D86C dekoduje wszystkie znaki pochodne na odpowiadające im znaki pierwotne; ponadto, jeśli S0 wystąpi przed S1, to wszystkie znaki między S0 i S1, ze znakami S0 i S1 włącznie, są dekodowane na NU; jeśli natomiast S1 wystąpi przed S0, to jest ono dekodowane na DE.
- W przypadku większej ilości jednakowo interpretowanych znaków, w tablicy odwołano się do jednego z następujących zestawów:
 

ZESTAW 1: SP, HT, !, #, %, &, @, \_\_, N2, N3, N4, N7;

ZESTAW 2: NU, SH, ST, ET, EN, EY, AC, BE, BS, SO, SI, D1, D2, D3, D4, NA, SY, EB, EM, FS, GS, RS, US (por. uwaga 5);

ZESTAW 3: SI, DL, CN, SU, DE (por.uwaga 5);

ZESTAW 4: SP, #, @, ", %, \_, !, &;

ZESTAW 5: HT, SP, !, # (lub £), %, &, @, ^,  
\_, ` , = (lub N7), {123};

ZESTAW 6: SI, DL, CN, SU;

ZESTAW 7: NU, SH, ST, ET, EN, EY, AC, BE,  
BS, D1, D2, D3, D4, NA, SY, EB,  
EM, FS, GS, RS, US, DE;

ZESTAW 8: ≠, =, [ , £.

## 6.9. WYBRANE DEKODERY TYPU KW6 - KOD ZEWNĘTRZNY

KW-6		M-2		FERRANTI		D-2		D-2E		OLIV	K-8
ZNAK	JART.	DL2P	DREP	DGFE	DDV1	DDV2	DDU1	DDU2	DGOL	DG8P	
A-Z	21-46	A-Z	A-Z	A-Z	A-Z	A-Z	A-Z	A-Z	A-Z	A-Z	A-Z
0-9	41-57	0-9	0-9	0-9	0-9	0-9	0-9	0-9	0-9	0-9	0-9
SP	1	SP	SP	SP	SP	SP	SP	SP	SP	SP	
LF	58	NL	NL	NL	NL	NL	NL	NL	NL	LF	
CR	62	CR	CR	CR	CR	CR	CR	CR	CR	CR	
FF	60	NL	NL	NL	FF	FF	FF	FF	FF	FF	
+	4	+	+	+	+	+	+	+	+	+	
-	8	-	-	-	-	-	-	-	-	-	
/	9	/	/	/	/	/	/	/	/	/	
*	5	* , 7	*	*	*	*	*	*	*	*	
(	3	(	(	(	(	(	(	(	(	(	
)	6	)	)	)	)	)	)	)	)	)	
[	14	[ , □	[ , □	[ , □	[ , □	[ , □	[ , □	[ , □	[ , □	[ , □	
]	15	] , □	] , □	] , □	] , □	] , □	] , □	] , □	] , □	] , □	
<	16	⊕	⊕	SP	<	<	<	<	<	<	
=	13	=	=	=	=	=	=	=	=	=	
>	17	⊗	⊗	>	>	>	>	>	>	>	
.	2	.	.	.	.	.	.	.	.	.	
:	11	:	:	:	:	:	:	:	:	:	
!	7	Ω , !	Ω , !	Ω , !	!	!	!	!	!	!	
;	10	;	;	;	;	;	;	;	;	;	
'	12	'	x, ⓧ	'	'	'	'	'	'	'	
?	18	⊗	⊕	?	>	SP	↑	↑	↑	↑	
◇	59	◇, □	v, ]	x, ⓧ	<	<	<	<	(125)	◇	
⊕	61	⊗	⊕	⊕	⊕	⊕	⊕	⊕	⊕	⊕	
ES	19	⊗	⊕	POWJANE	POWJANE	POWJANE	POWJANE	POWJANE	ES	ES	
SO	20	POWJANE	POWJANE	= , ⓧ	POWJANE	POWJANE	POWJANE	POWJANE	SO	SO	
SI	47	POWJANE	POWJANE	v, ]	POWJANE	POWJANE	POWJANE	POWJANE	SI	SI	
NU	0	NU	FS	POWJANE	POWJANE	POWJANE	POWJANE	POWJANE	NU	NU	
DE	63	POWJANE	POWJANE	POWJANE	POWJANE	POWJANE	POWJANE	POWJANE	DE	DE	

## UWAGI DO TABLICY 6.9

- A-Z oraz 0-9 oznaczają wszystkie kolejne litery alfabetu łacińskiego oraz cyfry.
- Przez  $\{n\}$  oznaczono znak o wartości  $n$ , nie posiadający w danym kodzie reprezentacji graficznej.
- Podanie dwóch lub więcej znaków, np.  $x, \diamond$  oznacza, że znaki te mają jednakową reprezentację binarną (różne oznaczenia na różnych egzemplarzach urządzeń).
- NL oznacza parę znaków: CR i LF.
- Dekoder DDV2 dekoduje pary znaków  $A \diamond, C \diamond, E \diamond, L \diamond, N \diamond, O \diamond, S \diamond, Z \diamond$  na  $\text{Ą}, \text{Ć}, \text{Ę}, \text{Ł}, \text{Ń}, \text{Ó}, \text{Ś}, \text{Ż}$ .
- Dekoder DDU2 dekoduje pary znaków  $A \diamond, C \diamond, E \diamond, L \diamond, O \diamond, S \diamond$  na  $\text{Ą}, \text{Ć}, \text{Ę}, \text{Ł}, \text{Ó}, \text{Ś}$ .

## 6.10. WYBRANE DEKODERY TYPU KOD ZEWNĘTRZNY - KWS

	KW-6 D68C	ICL DIL8	BULL DBU8	BULL2 DBL8	OLIV. DOL3	OPTIM DOP6	FERR DFE8	M-2 DM28	KW-8	
									ZNAK	WART.
	A-Z	A-Z	A-Z bez 011	A-Z	A-Z	A-Z	A-Z	A-Z a-z	A-Z	130-180
					{92}	UWAGA 4			z	184
					a-z {124}	a-z UWAGA 4			a-z	194-244 z 248
	0-9	0-9	0-9	0-9	0-9	0-9	0-9	0-9	0-9	96-114
	SP	SP	SP	SP	SP	SP	SP	SP	SP	64
	LF				LF	ZEST.4	LF	LF	LF	20
	CR			—	CR	—, CR	CR	CR	CR	26
	FF				FF				FF	24
					HT	←, HT			HT	18
					VT				VT	22
	+	+		+	+	+	+	+	+	86
	-	-/11		-	-	-	-	-	-	90
	/	/		/	/	/	/	/	/	94
	x	x		x	x	8, x	x	7, x	x	84
	!	!		!	!	!	!, !	!	!	116
	!	!		!	!	!	!, !	!, !	!	118
	.	.		.	.	.	.	.	.	92
	.	.		.	.	UWAGA 4	.	.	.	88
	(	(		(	(	%, (	(	(	(	80
	)	)		)	)	!, )	)	)	)	82
	[	[		[	[	!, [	[, =, [	[, □	[	182
	]	]		]	]	!, ]	], v	], □	]	186
	<	<		<	<	^, <	=	=	<	120
	=	=		=	=	=	=	=	=	122
	>	>		>	>	!! >	>	>	>	124
	?	?		{7/8}	?	?	?	?	?	126
	◇	z, ◇		{12/7/8}	{125}	7, ◇	x, ◇	◇ □	◇	250
				{0/7/8}						66
	□	8, □		{11/3/8}	8, □	z	z	8	□	72
	x	x		{0/4/8}	x	x	x	x	x	74

	KI-6 DGBC	ICL DILB	BULL DBUB	BULL DBLB	OLIV DOLB	OPTIM DOFB	FERR. DFEB	M-2 DM2B	KY-8	
									ZHAK	JART.
		" 4/10 ⊙ ⊕ —		" {12} {4/8} {3/8}	" 4 ⊙ 4, ⊕ —	" "	" "		" 4 ⊙ ⊕ —	68 78 76 128 70 190
					^ (123) —				N2 N3 N4 N7	188 192 246 252
	SO SI DE NU ES				SO SI EL DK NU ES				SO SI EL DE NU ES	28 30 50 254 0 54
					SH ST KT EN KY AC BE BS DL D1 D2 D3 D4 NA SY EB				SH ST KT EN KY AC BE BS DL D1 D2 D3 D4 NA SY EB	2 4 6 8 10 12 14 16 32 34 36 38 40 42 44 46

	KW-6 D68C	ICL DIL8	BULL DBU8	BULL2 DBL8	OLIV DOLA	OPTIM DOP8	FERR. DFB8	M-2 DM28	KW-8	
									ZNAK	WART.
					CN				CN	48
					SU				SU	52
					Y8				Y8	56
					Q8				Q8	58
					RS				RS	60
					U8				U8	62
						ZEST. 3	ERASE		POMIJANE	
					EDNEC DANTEH	EDNEC DANTEH	EDNEC DANTEH	EDNEC DANTEH	EDNEC DANTEH	-0



## UWAGI DO TABLICY 6.10

- A-Z, a-z i 0-9 oznaczają wszystkie kolejne duże i małe litery alfabetu łacińskiego oraz cyfry.
- Dla kombinacji dziurek na taśmie papierowej przez  $\{n\}$  oznaczono znak o wartości binarnej  $n$ , nie posiadający w danym kodzie reprezentacji graficznej; dla kart przez  $\{n\}$  lub  $\{n/m\}$  lub  $\{n/m/k\}$  oznaczono kombinacje składające się z dziurek w wierszach  $n$  lub  $n$  i  $m$  lub  $n$ ,  $m$  i  $k$ , które to kombinacje nie mają reprezentacji graficznej.
- Podanie dwóch lub więcej znaków, np.  $x, \diamond$  oznacza, że albo znaki te mają jednakową reprezentację binarną (różne oznaczenia na różnych egzemplarzach urządzeń), albo mają różne reprezentacje jednakowo interpretowane przez dekodery.
- Urządzenia typu OPTIMA są wykonane w dwóch wariantach, różniących się przestawieniem czcionki ze znakami  $\dot{Z}$  i  $\dot{z}$  z czcionką  $\acute{n}$  i  $\acute{e}$ . Dlatego, w zależności od urządzenia, na którym przygotowano taśmę, znaki  $\dot{Z}, \dot{z}, \acute{n}$  i  $\acute{e}$  będą zdekodowane poprawnie lub znaki  $\dot{Z}$  i  $\acute{n}$  oraz  $\dot{z}$  i  $\acute{e}$  zostaną wzajemnie zamienione.
- Dekoder DOPS wprowadza litery  $\acute{a}, \acute{c}, \acute{e}, \dots$  na pochodne znaków  $a, c, e, \dots$ , poza tym przypadkiem wymienione dekodery nie wprowadzają żadnych znaków na znaki pochodne.
- Dekoder DOLS nie kontroluje bitów parzystości.

W przypadku większej ilości jednakowo interpretowanych znaków, w tablicy odwołano się

do jednego z następujących zestawów:

ZESTAW 1: znaki z klawiatury LF, ↑, —  
oraz znak z przystawki ↑,  
oznaczany również ↑ ;

ZESTAW 2: wszystkie wartości nie wymienio-  
ne w tabeli, w uwagach 4 i 5 oraz  
w zestawach 1 i 3;

ZESTAW 3: znaki z przystawki SPE, NS, WS,  
LCA, SP, C1, ↓ (oznaczany czasem  
←--), oraz {255} .

## 6.11. WYBRANE DRODZKI TYPU KW6 - KOD ZEWNĘTRZNY

KW-8 ZNAK	M-2 DOM2	OPTIMA DBOP	DW-2 DDVB	DW-2E DDVB	OLIVET DSOL	KW6 D46P
A-Z	130-180	A-Z	A-Z	A-Z	A-Z	A-Z
z	184	Z	uwaga 6	z	z	{92}
a-z	194-244	A-Z	a-z	A-Z	A-Z	a-z
A	248	Z	uwaga 6	z	z	{124}
0-9	96-114	0-9	0-9	0-9	0-9	0-9
SP	64	SP	SP	SP	SP	SP
LF	20	NL	—, NL	NL	NL	LF
CR	26	CR	—, CR	CR	CR	CR
FF	24	NL	—, NL	FF	FF	FF
HT	18	SP	—, HT	SP	SP	HT
VT	22	NL	—, NL	FF	FF	VT
+	86	+	+	+	+	+
-	90	-	-	-	-	-
/	94	/	/	/	/	/
*	84	*, ?	*	*	*	*
:	116	:	:	:	:	:
;	118	;, □	;	;	;	;
.	92	.	.	.	.	.
,	88	,	uwaga 6	,	,	,
(	80	(	%	(	(	(
)	82	)	!	)	)	)
[	182	[, □	,	[	[	[
]	186	], □	.	]	]	]
<	120	[, □	^	<	<	<
=	122	=	=	=	=	=
>	124	], □	.	>	>	>
?	126	SP	SP	>	?	?
◇	250	◇, □	?	<	{125}	◇
!	66	SP	SP	SP	!	SP
□	72	⊕	±	⊕	⊕, □	□

ZNAK	KW-8 WART.	M-2 DGM2	OPTIMA DSOP	DW-2 DDV8	DW-2E DDU8	OLIVET D80L	KW6 D86P		
%	74	SP	SP	SP	%	%	SP		
"	68	"	"	SP	"	"	"		
'	78	'	"	'	'	'	'		
&	76	SP	SP	SP	SP	&	SP		
@	128	SP	SP	SP	SP	@	SP		
#	70	SP	SP	SP	#	#	SP		
-	190	SP	-	-	SP	-	SP		
N2	188	SP	SP	SP	SP	"	SP		
N3	192	SP	SP	SP	SP		SP		
N4	246	SP	SP	SP	SP	(123)	SP		
N7	252	SP	SP	SP	SP	—	SP		
SO	28	[ , □		[	[	SO	NU		
SI	30	] , ■		]	]	SI	NU		
EM	50	POMI JANE	POMI JANE	POMI JANE	POMI JANE	EM	NU		
DE	254	POMI JANE	POMI JANE	POMI JANE	POMI JANE	DE	DE		
NU	0	NU	NU	POMI JANE	POMI JANE	NU	NU		
ES	54	⊠	⊠	⊠	⊠	ES	ES		
SH	2	POMI JANE	POMI JANE	POMI JANE	POMI JANE	SH	NU		
ST	4	POMI JANE	POMI JANE	POMI JANE	POMI JANE	ST	NU		
ET	6	POMI JANE	POMI JANE	POMI JANE	POMI JANE	ET	NU		
EN	8	POMI JANE	POMI JANE	POMI JANE	POMI JANE	EN	NU		
EY	10	POMI JANE	POMI JANE	POMI JANE	POMI JANE	EY	NU		
AC	12	POMI JANE	POMI JANE	POMI JANE	POMI JANE	AC	NU		
BE	14	POMI JANE	POMI JANE	POMI JANE	POMI JANE	BE	NU		
BS	16	POMI JANE	POMI JANE	POMI JANE	POMI JANE	BS	NU		
DL	32	⊠	⊠	⊠	⊠	DL	DE		
D1	34	POMI JANE	LO1	POMI JANE	POMI JANE	D1	NU		
D2	36	POMI JANE	LO2	POMI JANE	POMI JANE	D2	NU		
D3	38	POMI JANE	LEU	POMI JANE	POMI JANE	D3	NU		
D4	40	POMI JANE	⊙	POMI JANE	POMI JANE	D4	NU		
NA	42	POMI JANE	POMI JANE	POMI JANE	POMI JANE	NA	NU		
BY	44	POMI JANE	POMI JANE	POMI JANE	POMI JANE	BY	NU		
EB	46	POMI JANE	POMI JANE	POMI JANE	POMI JANE	EB	NU		

K-8		M-2	OPTIMA	DF-2	DF-2E	OLIVET.	KW6		
ZNAK	ART.	DSM2	DSOP	DDVB	DDUE	DSOL	DS6P		
CN	48	*	2	*	*	CN	DE		
SU	52	*	2	*	*	SU	DE		
FS	56	POMJANE	POMJANE	POMJANE	POMJANE	FS	NU		
GS	58	POMJANE	POMJANE	POMJANE	POMJANE	GS	IU		
RB	60	POMJANE	POMJANE	POMJANE	POMJANE	RB	NU		
UB	62	POMJANE	POMJANE	POMJANE	POMJANE	UB	NU		

## UWAGI DO TABLICY 6.11

- A-Z, a-z i 0-9 oznaczają wszystkie kolejne duże i małe litery alfabetu łacińskiego oraz cyfry.
- Przez  $\{n\}$  oznaczono znak o wartości  $n$ , nie posiadający w danym kodzie reprezentacji graficznej.
- Podanie dwóch lub więcej znaków, np.  $\diamond, \square$  oznacza, że znaki te mają jednakową reprezentację binarną (różne oznaczenie na różnych egzemplarzach urządzeń).
- NL oznacza parę znaków: CR, LF.
- W tablicy podano wyłącznie sposób przekodowania znaków podstawowych; wszystkie znaki pochodne są przekodowywane tak, jak odpowiadające im znaki podstawowe z zachowaniem następujących wyjątków:
  - a) D80P przekodowuje znaki Ł,ą,ć,ę,ł,ń,ó,ś,ź,ż na takie same znaki (por. też uwagę 6);
  - b) DDV8 przekodowuje znaki Ȧ,Ĉ,Ĕ,Ł,Ń,Ô,Ŝ,Ż oraz ą,ć,ę,ł,ń,ó,ś,ź odpowiednio na Ȧ,Ĉ,Ĕ,Ł,Ń,Ô,Ŝ,Ż;
  - c) DDÜ8 przekodowuje znaki Ȧ,Ĉ,Ĕ,Ł,Ô,Ŝ oraz ą,ć,ę,ł,ó,ś odpowiednio na Ȧ,Ĉ,Ĕ,Ł,Ô,Ŝ.
- Urządzenia typu OPTIMA są wykonane w dwóch wariantach, różniących się przedstawieniem czcionki ze znakami Ż i ż z czcionką ń i ,. Dlatego w zależności od urządzenia, na którym taśma będzie wypisywana, znaki Ż, ż, ń i , będą zdekodowane poprawnie, lub znaki Ż i ń oraz ż i , zostaną wzajemnie zamienione.

- Jeśli znak SO wystąpi przed SI, to dekodery D86P dekoduje wszystkie znaki między SO i SI, ze znakami SO i SI włącznie na znaki NU; jeśli natomiast SI wystąpi przed SO, to SI jest dekodowane przez D86P na znak DE.

## 6.12. KOD M-2

<u>DZIES.</u>	<u>OKTAL.</u>	<u>LITERY</u>	<u>CYFRY</u>	<u>OBRAZ TAŚMY</u>
1	01	T	5	00.001
2	02	CR	CR	00.010
3	03	O	9	00.011
4	04	SP	SP	00.100
5	05	H	◇, □	00.101
6	06	N	,	00.110
7	07	M	.	00.111
8	10	LF	LF	01.000
9	11	L	)	01.001
10	12	R	4	01.010
11	13	G	], ⊠	01.011
12	14	I	8	01.100
13	15	P	0	01.101
14	16	C	:	01.110
15	17	V	=	01.111
16	20	E	3	10.000
17	21	Z	+	10.001
18	22	D	⊕	10.010
19	23	B	*, ?	10.011
20	24	S	'	10.100
21	25	Y	6	10.101
22	26	F	[, □	10.110
23	27	X	/	10.111
24	30	A	-	11.000
25	31	W	2	11.001
26	32	J	;, Ω	11.010
27	33	FS	FS	11.011
28	34	U	7	11.100
29	35	Q	1	11.101
30	36	K	(	11.110
31	37	LS	LS	11.111



## 6.13. KOD FERRANTI

DZIES.	OKTAL.	LITERY	CYFRY	OBRAZ TAŚMY
0	00	FS	FS	00.000
1	01	P	0	00.001
2	02	H	8	00.010
3	03	X	x,◇	00.011
4	04	D	4	00.100
5	05	T	;,→	00.101
6	06	L	v,]	00.110
7	07	.	.	00.111
8	10	B	2	01.000
9	11	R	≥,:	01.001
10	12	J	=	01.010
11	13	Z	+	01.011
12	14	F	)	01.100
13	15	V	6	01.101
14	16	N	SP	01.110
15	17	£	CR	01.111
16	20	A	1	10.000
17	21	Q	>	10.001
18	22	I	=,[,≠	10.010
19	23	Y	9	10.011
20	24	E	(	10.100
21	25	U	5	10.101
22	26	M	LF	10.110
23	27	?	η,'	10.111
24	30	C	*	11.000
25	31	S	3	11.001
26	32	K	-	11.010
27	33	LS	LS	11.011
28	34	G	7	11.100
29	35	X	/	11.101
30	36	O	,	11.110
31	37	ER	ER	11.111

## 6.14. KOD OPTIMA 528

DZIES.	OKTAL.	DUŻE LITERY	MAŁE LITERY	OBRAZ TAŚMY	UWAGI
0	0	NU	NU	00000.000	przewodzenie taśmy
1	1	§, *	1	00000.001	
2	2	%, (	2	00000.010	
4	4	?, 0	4	00000.100	
7	7	:	7	00000.111	
8	10	-	8	00001.000	
11	13	⊙		00001.011	stop
13	15	SPE	SPE	00001.101	koniec skoku taśmy
14	16	LEU	LEU	00001.110	przełączenie czytnika
16	20	SPACJA	SPACJA	00010.000	
19	23	l, )	3	00010.011	
21	25	+	5	00010.101	
22	26	=	6	00010.110	
25	31	/	9	00011.001	
26	32	`, [	`, ]	00011.010	
28	34	NS	NS	00011.100	wyłączył pisanie
31	37	↑,	↑,	00011.111	1. program
32	40	" ,	0	00100.000	
35	43	T	t	00100.011	
37	45	V	v	00100.101	
38	46	W	w	00100.110	
41	51	Z	z	00101.001	
42	52	l	zl	00101.010	
44	54	WS	WS	00101.100	włączył pisanie
47	57	LOA	LOA	00101.111	wyłączył dziurkowanie
49	61	ś	ó	00110.001	
50	62	S	s	00110.010	
52	64	U	u	00110.100	
55	67	X	x	00110.111	
56	70	Y	y	00111.000	
59	73	ś	.	00111.011	
61	75	§, >	^, <	00111.101	
62	76	HT, ---	HT, ---	00111.110	tabulator

64	100	ć	-	01000.000	
67	103	L	l	01000.011	
69	105	N	n	01000.101	
70	106	O	o	01000.110	
73	111	R	r	01001.001	
74	112	CR, —	CR, —	01001.010	
76	114	LO1	LO1	01001.100	włączenie 1.dziurkarki
79	117	LF, †	LF, †	01001.111	
81	121	J	j	01010.001	
82	122	K	k	01010.010	
84	124	M	m	01010.100	
87	127	P	p	01010.111	
88	130	Q	q	01011.000	
91	133	ę	ą	01011.011	
93	135	ż	ź	01011.101	patrz uwaga
94	136	↓, ---	↓, ---	01011.110	2.program
97	141	A	a	01100.001	
98	142	B	b	01100.010	
100	144	D	d	01100.100	
103	147	G	g	01100.111	
104	150	H	h	01101.000	
107	153	Ł	ł	01101.011	
109	155	SP	SP	01101.101	przeskok
110	156	LO2	LO2	01101.110	włączenie 2.dziurkarki
112	160	ń	,	01110.000	patrz uwaga
115	163	C	c	01110.011	
117	165	E	e	01110.101	
118	166	F	f	01110.110	
121	171	I	i	01111.001	
122	172	MAŁE LIT	MAŁE LIT	01111.010	
124	174	DUŻE LIT	DUŻE LIT	01111.100	
127	177	MASUJ. C	MASUJ. C	01111.111	
128	200	LF, ↗	LF, ↗	10000.000	

Uwaga: w niektórych urządzeniach wartości znaków Ź i ź oraz ń i , są zamienione ze sobą.



7. T A B L I C E

7.1. WARTOŚCI DZIESIĘTNE  $2^N$ ,  $2^{-N}$  oraz  $2^x$

$2^N$	$N$	$2^{-N}$	$x$	$2^x$
1	0	1.		
2	1	0.5		
4	2	0.25	0.1	1.071773462536293
8	3	0.125	0.2	1.148698354997035
16	4	0.0625	0.3	1.231144413344916
32	5	0.03125	0.4	1.319507910772894
64	6	0.015625	0.5	1.414213562373095
128	7	0.0078125	0.6	1.515716566510398
256	8	0.00390625	0.7	1.624504792712471
512	9	0.001953125	0.8	1.741101126592248
1024	10	0.0009765625	0.9	1.866065983073615
2048	11	0.00048828125	0.01	1.006955550056719
4096	12	0.000244140625	0.02	1.013959479790029
8192	13	0.0001220703125	0.03	1.021012125707193
16384	14	0.00006103515625	0.04	1.028113826656067
32768	15	0.000030517578125	0.05	1.035264923841377
65536	16	0.0000152587890625	0.06	1.042465760841121
131072	17	0.00000762939453125	0.07	1.049716683623067
262144	18	0.000003814697265625	0.08	1.057018040561380
524288	19	0.0000019073486328125	0.09	1.064370182453360
1048576	20	0.00000095367431640625	0.001	1.000693387462581
2097152	21	0.000000476837158203125	0.002	1.001387255711335
4194304	22	0.0000002384185791015625	0.003	1.002081605079633
8388608	23	0.00000011920928955078125	0.004	1.002776435901078
16777216	24	0.00000005960464477530625	0.005	1.003471748509503
33554432	25	0.0000000298023223876953125	0.006	1.004167543238973
67108864	26	0.00000001490116119384765625	0.007	1.004863820423785
134217728	27	0.000000007450580596923828125	0.008	1.005560580398468
268435456	28	0.0000000037242902984619140625	0.009	1.006257823497782
536870912	29	0.0000000018626451492309703125		
1073741824	30	0.000000000931322574615478515625		
2147483648	31	0.0000000004656612873077392578125		
4294967296	32	0.00000000023283064365386962890625		
8589934592	33	0.000000000116415321826934814453125		
17179869184	34	0.0000000000582076609134674072265625		
34359738368	35	0.00000000002910383045673370361328125		
68719476736	36	0.000000000014551915228366851806640625		
137438953472	37	0.0000000000072759576141834259033203125		
274877906944	38	0.00000000000363797880709171295166015625		
549755813888	39	0.000000000001818989403545856475830078125		
1099511627776	40	0.0000000000009094947017729282379150390625		
2199023255552	41	0.00000000000045474735088646411895751953125		
4398046511104	42	0.000000000000227373675443232059478759765625		
8796093022208	43	0.0000000000001136868377216160297393798828125		
17592186044416	44	0.00000000000005684341886080801486968994140625		
35184372088832	45	0.000000000000028421709430404007434844970703125		
70368744177664	46	0.0000000000000142108547152020037174224853515625		
140737488355328	47	0.00000000000000710542735760100185871124267578125		
281474974710656	48	0.000000000000003552713678800500929355621337890625		
562949953421312	49	0.0000000000000017763568394002504646778106689453125		
1125899906842624	50	0.00000000000000088817841970012523233890533447265625		

7.2. WARTOŚCI DZIESIĘTNE  $8^N$  I  $8^{-N}$

$N$	$8^N$	$8^{-N}$
0	1	1
1	8	0,125
2	64	0,015625
3	512	0,001953125
4	4096	0,000244140625
5	32768	0,000030517578125
6	262144	0,000003814697265625
7	2097152	0,000000476837158203125
8	16777216	0,000000059404644775390625
9	134217728	0,00000000745080596923828125
10	1073741824	0,00000000031322574615478515625
11	8589934592	0,00000000011641321826934814453125
12	68719476736	0,000000000014551915228366851806640625
13	549755813888	0,000000000001818989403545856475830078125
14	4398046511104	0,000000000000227373675443232059478759765625
15	35184372088832	0,00000000000002821709430400743844970703125
16	281474976710656	0,00000000000000352713678800500929355621337890625
17	22517998133685248	0,00000000000000044089209850062616169452667236328125
18	18014398509481984	0,0000000000000000055511151231257827021181583404541015625
19	14415188075855872	0,0000000000000000006938893907228377647697925567626953125
20	115292150460846976	0,000000000000000000086735173798840354720596224069595369140625
21	9223372036854775808	0,0000000000000000000108420217248355044340074528008699471142578125
22	73786976294838206464	0,0000000000000000000013552527156068805425093160010874271392822265625
23	590295810358703651712	0,00000000000000000000001694065894508600678136645001359283924102783203125
24	4722366482869645213696	0,00000000000000000000000021758236813575084767080625169910490512847900390625
25	3778931862957161709568	0,0000000000000000000000000026469779601696885595885078146234811314105987548828125

7.3. WARTOŚCI DZIESIĘTNE 16<sup>N</sup> 1 16<sup>-N</sup>

16 <sup>N</sup>	N	16 <sup>-N</sup>
1	0	1.0625
16	1	0.00390625
256	2	0.000244140625
4096	3	0.0000152587890625
65536	4	0.00000095367431640625
1048576	5	0.00000059404644775390625
16777216	6	0.00000037242902984619140625
268435456	7	0.00000002383064365386962890625
4294967296	8	0.0000000014551915228366851806640625
68719476736	9	0.0000000000994947017729282379150390625
1099511627776	10	0.000000000005684341886080801686968994140625
17592186044416	11	0.000000000000355271367880050929353621337890625
281474976710656	12	0.00000000000002204664925031308084726336181640625
4503599627370496	13	0.00000000000000138778780781445675529539585113525390625
72057594037927936	14	0.0000000000000000867361737983403547205962240693953369140625
1152921504606846976	15	0.00000000000000000542101086242752217003726400434970855712890625
18446744073709551916	16	0.000000000000000000338613178901720135627325000271856784820556640625
295147905179352825856	17	0.00000000000000000000211758236813575084767080625169910490512847900390625
4722366482869645213696	18	

7.4. WARTOŚCI OKRĄGŁE  $10^M$  I  $10^{-M}$ 

$10^M$	$M$	$10^{-M}$
1	0	1.000 000 000 000 000 000 00
12	1	0.063 146 314 631 463 146 31
144	2	0.005 075 341 217 270 243 66
1 750	3	0.000 406 111 564 570 651 77
23 420	4	0.000 032 155 613 530 704 15
303 240	5	0.000 002 476 132 610 706 54
3 641 100	6	0.000 000 206 157 364 055 37
46 113 200	7	0.000 000 015 327 745 152 75
575 360 400	8	0.000 000 001 257 143 561 06
7 346 545 000	9	0.000 000 000 104 560 276 41
112 402 762 000	10	0.000 000 000 006 676 337 66
1 351 035 564 000	11	0.000 000 000 000 537 657 77
16 432 451 210 000	12	0.000 000 000 000 043 136 32
221 411 634 520 000	13	0.000 000 000 000 003 411 35
2 657 142 036 440 000	14	0.000 000 000 000 000 264 11
34 327 724 461 500 000	15	0.000 000 000 000 000 022 01
434 157 115 760 200 000	16	0.000 000 000 000 000 001 63
5 432 127 413 542 400 000	17	0.000 000 000 000 000 000 14
67 405 553 164 751 000 000	18	0.000 000 000 000 000 000 01



## 7.5. NIEKTÓRE STAŁE MATEMATYCZNE

	dziesiętnie	oktalnie
$\pi$	3.141 592 654	3.110 375 524
$\pi^{-1}$	0.318 309 886	0.242 763 016
$\sqrt{\pi}$	1.772 453 851	1.613 376 111
$\ln \pi$	1.144 729 886	1.112 064 044
$\log_{10} \pi$	0.497 149 873	0.376 424 666
$\log_2 \pi$	1.651 496 129	1.515 441 632
$e$	2.718 281 829	2.557 605 213
$e^{-1}$	0.367 879 441	0.274 265 307
$\sqrt{e}$	1.648 721 271	1.514 112 307
$\log_{10} e$	0.434 294 482	0.336 267 543
$\log_2 e$	1.442 695 041	1.342 521 663
$\sqrt{2}$	1.414 213 562	1.324 047 463
$\ln 2$	0.693 147 181	0.542 710 300
$\log_{10} 2$	0.301 029 996	0.232 101 152
$\sqrt{10}$	3.162 277 660	3.123 054 073
$\ln 10$	2.302 585 093	2.232 730 674
$\log_2 10$	3.321 928 095	3.244 647 411

### 7.6. TABLICZKA DODAWANIA I MNOŻENIA LICZB OKTALNYCH

#### Dodawanie

0	01	02	03	04	05	06	07
1	02	03	04	05	06	07	10
2	03	04	05	06	07	10	11
3	04	05	06	07	10	11	12
4	05	06	07	10	11	12	13
5	06	07	10	11	12	13	14
6	07	10	11	12	13	14	15
7	10	11	12	13	14	15	16

#### Mnożenie

1	02	03	04	05	06	07
2	04	06	10	12	14	16
3	06	11	14	17	22	25
4	10	14	20	24	30	34
5	12	17	24	31	36	43
6	14	22	30	36	44	52
7	16	25	34	43	52	61

7.7. TABLICA ZMIANY LICZB OKTALNYCH NA DZIESIĘTNE  
I ODWROTNIE

cyfra oktalna	$X_1$	$X_2$	$X_3$	$X_4$	$X_5$	$X_6$	$X_7$	$X_8$
0	0	0	0	0	0	0	0	0
1	2097152	262144	32768	4096	512	64	8	1
2	4194304	524288	65536	8192	1024	128	16	2
3	6291456	786432	98304	12288	1536	192	24	3
4	*0	1048576	131072	16384	2048	256	32	4
5	*2097152	1310720	163840	20480	2560	320	40	5
6	*4194304	1572864	196608	24576	3072	384	48	6
7	*6291456	1835008	229376	28672	3584	448	56	7

Sposób korzystania z tablicy

7.7.1. Zamiana liczby oktalnej na dziesiętną:

dla liczby oktalnej postaci  $X_1X_2X_3X_4X_5X_6X_7X_8$  ( $X_i$  są cyframi 0,1,...,7) znaleźć odpowiednie wartości w tablicy, po czym wartości te dodać. Jeśli jedna z wartości jest oznaczona gwiazdką, wówczas otrzymaną sumę należy uzupełnić znakiem minus.

Przykład 1 - liczba oktalna 35740512:

$$6291456 + 1310720 + 229376 + 16384 + 0 + \\ + 320 + 8 + 2 = 7848266$$

Przykład 2 - liczba oktalna 50010033:

$$2097152 + 0 + 0 + 4096 + 0 + 0 + 2^4 + 3 = 2101275$$

Ponieważ przy wartości 2097152 wystąpiła \*, więc wartością liczby oktalnej 50010033 jest -2101275.

7.7.2. Zamiana liczby dziesiętnej na oktalną:

dla liczby dziesiętnej  $X$  znaleźć w tablicy największą cyfrę oktalną  $C_1$ , dla

1275.

7. Zamiana liczby dziesiętnej na oktalną:

dla liczby dziesiętnej  $X$  znaleźć w tablicy największą cyfrę oktalną  $C_1$ , dla

której  $X_1 \leq |X|$  oraz  $X_1$  jest bez gwiazdki, jeśli  $X$  ma znak plus lub  $X_1$  jest z gwiazdką, jeśli  $X$  ma znak minus. Obliczyć  $Y_1 = |X| - X_1$  i dla liczby  $Y_1$  znaleźć największą cyfrę oktalaną  $C_2$ , dla której  $X_2 \leq Y_1$ ; obliczyć  $Y_2 = Y_1 - X_2$  i dla liczby  $Y_2$  znaleźć największą cyfrę oktalaną  $C_3$ , dla której  $X_3 \leq Y_2$ ; postępować w ten sposób aż do otrzymania cyfry  $C_8$ . Cyfry  $C_1, C_2, \dots, C_8$  są kolejnymi cyframi liczby  $X$  w układzie oktalanym.

Przykład 1 - liczba dziesiętna  $X = 7848266$ :

$$\begin{aligned}
 C_1 &= 3; \text{ wtedy } X_1 = 6291456 \\
 Y_1 &= |X| - X_1 = 7848266 - 6291456 = 1556810 \\
 C_2 &= 5; \text{ wtedy } X_2 = 1310720 \\
 Y_2 &= Y_1 - X_2 = 1556810 - 1310720 = 246090 \\
 C_3 &= 7; \text{ wtedy } X_3 = 229376 \\
 Y_3 &= Y_2 - X_3 = 246090 - 229376 = 16714 \\
 C_4 &= 4; \text{ wtedy } X_4 = 16384 \\
 Y_4 &= Y_3 - X_4 = 16714 - 16384 = 330 \\
 C_5 &= 0; \text{ wtedy } X_5 = 0 \\
 Y_5 &= Y_4 - X_5 = 330 \\
 C_6 &= 5; \text{ wtedy } X_6 = 320 \\
 Y_6 &= Y_5 - X_6 = 330 - 320 = 10 \\
 C_7 &= 1; \text{ wtedy } X_7 = 8 \\
 Y_7 &= Y_6 - X_7 = 10 - 8 = 2 \\
 C_8 &= 2
 \end{aligned}$$

Liczba dziesiętna  $X = 7848266$  przyjmuje postać oktalaną 35740512, co jest zgodne z przykładem 1 z punktu 7.7.1.

Przykład 2 - liczba  $-2101275$ :

$$\begin{aligned}
 C_1 &= 5; X_1 = 2097152 \\
 Y_1 &= |X| - X_1 = 2101275 - 2097152 = 4123
 \end{aligned}$$

$$\begin{aligned}
 C_2 &= 0; X_2 = 0 \\
 Y_2 &= Y_1 - X_2 = 4123 \\
 C_3 &= 0; X_3 = 0 \\
 Y_3 &= Y_2 - X_3 = 4123 \\
 C_4 &= 1; X_4 = 4096 \\
 Y_4 &= Y_3 - X_4 = 27 \\
 C_5 &= 0; X_5 = 0 \\
 Y_5 &= Y_4 - X_5 = 27 \\
 C_6 &= 0; X_6 = 0 \\
 Y_6 &= Y_5 - X_6 = 27 \\
 C_7 &= 3; X_7 = 24 \\
 Y_7 &= Y_6 - X_7 = 3 \\
 C_8 &= 3
 \end{aligned}$$

Liczba dziesiętna  $X = -2101275$  przyjmuje postać oktalną 50010033, co jest zgodne z przykładem 2 z punktu 7.7.1.

#### 7.8. TABLICA ZMIANY LICZB SZESNASTKOWYCH NA DZIESIĘTNE I ODWROTNIE

cyfra szesnastkowa	$X_1$	$X_2$	$X_3$	$X_4$	$X_5$	$X_6$
0	0	0	0	0	0	0
1	1048576	65536	4096	256	16	1
2	2097152	131072	8192	512	32	2
3	3145728	196608	12288	768	48	3
4	4194304	262144	16384	1024	64	4
5	5242880	327680	20480	1280	80	5
6	6291456	393216	24576	1536	96	6
7	7340032	458752	28672	1792	112	7
8	*0	524288	32768	2048	128	8
9	*1048576	589824	36864	2304	144	9
A	*2097152	655360	40960	2560	160	10
B	*3145728	720896	45056	2816	176	11
C	*4194304	786432	49152	3072	192	12
D	*5242880	851968	53248	3328	208	13
E	*6291456	917504	57344	3584	224	14
F	*7340032	983040	61440	3840	240	15

Sposób korzystania z tablicy

7.8.1. Zamiana liczby szesnastkowej na dziesiętną: dla liczby szesnastkowej postaci  $X_1X_2X_3X_4X_5X_6$  ( $X_i$  są cyframi  $0, 1, \dots, F$ ) znaleźć odpowiednie wartości w tablicy, po czym wartości te dodać. Jeśli jedna z wartości jest oznaczona gwiazdką, wówczas otrzymaną sumę należy uzupełnić znakiem minus.

Przykład 1 - liczba szesnastkowa 1A5801:

$$1048576 + 655360 + 20480 + 2048 + 0 + 1 = 172646$$

Przykład 2 - liczba szesnastkowa F00001:

$$7340032 + 0 + 0 + 0 + 0 + 1 = 7340033$$

Ponieważ przy wartości 7340032 wystąpiła \*, więc wartością liczby szesnastkowej F00001 jest -7340033.

7.8.2. Zamiana liczby dziesiętnej na szesnastkową: dla liczby  $X$  znaleźć w tablicy największą cyfrę szesnastkową  $C_1$ , dla której  $X_1 \leq |X|$  oraz  $X_1$  jest bez gwiazdki, jeśli  $X$  ma znak plus lub  $X_1$  ma gwiazdkę, jeśli  $X$  ma znak minus. Obliczyć  $Y_1 = |X| - X_1$  i dla liczby  $Y_1$  znaleźć największą cyfrę szesnastkową  $C_2$ , dla której  $X_2 \leq Y_1$ ; obliczyć  $Y_2 = Y_1 - X_2$  i dla liczby  $Y_2$  znaleźć największą cyfrę szesnastkową  $C_3$ , dla której  $X_3 \leq Y_2$ ; postępować w ten sposób aż do otrzymania cyfry  $C_6$ . Cyfry  $C_1, C_2, \dots, C_6$  są kolejnymi cyframi liczby  $X$  w układzie szesnastkowym.

Przykład 1 - liczba dziesiętna  $X = 1726465$

$$C_1 = 1; X_1 = 1048576$$

$$Y_1 = |X| - X_1 = 1726465 - 1048576 = 677889$$

$$C_2 = A; X_2 = 655360$$

$$Y_2 = Y_1 - X_2 = 677889 - 655360 = 22529$$

$$C_3 = 5; X_3 = 20480$$

$$Y_3 = Y_2 - X_3 = 22529 - 20480 = 2049$$

$$C_4 = 8; X_4 = 2048$$

$$Y_4 = Y_3 - X_4 = 2049 - 2048 = 1$$

$$C_5 = 0; X_5 = 0$$

$$Y_6 = Y_4 - X_5 = 1$$

$$C_6 = 1$$

Liczba dziesiętna  $X = 1726465$  przyjmuje w układzie szesnastkowym postać 1A5801, co jest zgodne z przykładem 1 z punktu 7.8.1.

Przykład 2 - liczba dziesiętna  $X = -7340033$

$$C_1 = F; X_1 = 7340032$$

$$Y_1 = |X| - X_1 = 1$$

$$C_2 = 0; X_2 = 0$$

$$Y_2 = 1$$

$$C_3 = 0; X_3 = 0$$

$$Y_3 = 1$$

$$C_4 = 0; X_4 = 0$$

$$Y_4 = 1$$

$$C_5 = 0; X_5 = 0$$

$$Y_5 = 1$$

$$C_6 = 1$$

Liczba dziesiętna  $X = -7340033$  przyjmuje w układzie szesnastkowym postać F00001, co jest zgodne z przykładem 2 z punktu 7.8.1.

### 7.9. TABLICA ZMIANY LICZB OKTALNYCH NA SZESNASTKOWE

Cyfra oktalna	$X_1$	$X_2$	$X_3$	$X_4$
0	000	00	0	0
1	200	40	8	1
2	400	80	10	2
3	600	C0	18	3
4	800	100	20	4
5	A00	140	28	5
6	C00	180	30	6
7	E00	1C0	38	7

### TABLICA POMOCNICZA

+	1	2	3	4	5	6	7
8	.	A	B	C	D	E	F
A	B	.	.	.	.	.	.
C	D	E	F	.	.	.	.
E	F	.	.	.	.	.	.

Sposób korzystania z tablicy: liczbę oktalną podzielić na dwie części po 4 cyfry. Każdą część zamienia się na postać szesnastkową oddzielnie. Dla każdej części liczby oktalnej postaci  $X_1X_2X_3X_4$  ( $X_i$  są cyframi 0,1,...,7) znaleźć odpowiednie wartości w tablicy, po czym wartości te dodać w układzie szesnastkowym. Przy dodawaniu cyfr szesnastkowych, których suma przewyższa 9 można posłużyć się tablicą pomocniczą.

Przykład - liczbę oktalną 76540123 zamienić na szesnastkową.



Część I liczby: 7654

Część II liczby: 0123

Część I

$X_1=7$       E00

$X_2=6$       180

$X_3=5$       28

$X_4=4$     +    4

---

FAC

Część II

$X_1=0$     000

$X_2=1$     40

$X_3=2$     10

$X_4=3$     +    3

---

053

Liczba oktalna 76540123 ma w układzie szesnastkowym postać FAC053.

## 7.10. WARTOŚCI DZIESIĘTNE, OKTALNE I NAZWY CZĘŚCI OPERACYJNYCH ROZKAZÓW

### 7.10.1. Rozkazy typu a (15-bitowy argument operacji)

WARTOŚĆ DZIES. OKT.		NAZWA OPERACJI	WARTOŚĆ DZIES. OKT.		NAZWA OPERACJI
0	00	SKO	31	37	P31
1	01	SDY	32	40	PZA
2	02	SSL	33	41	PAM
3	03	WRO	34	42	PAD
4	04	DOP	35	43	UAD
5	05	ODP	36	44	UAM
6	06	P06	37	45	UMA
7	07	P07	38	46	DOA
8	10	P08	39	47	ODA
9	11	P09	40	50	DOM
10	12	P10	41	51	ODM
11	13	P11	42	52	MNM
12	14	DOZ	43	53	DZD
13	15	ODZ	44	54	DOL
14	16	MNZ	45	55	OSL
15	17	DZZ	46	56	MNL
16	20	P16	47	57	POL
17	21	P17	49	61	FMM
18	22	P18	50	62	UMM
19	23	P19	51	63	PMB
20	24	P20	52	64	UMB
21	25	P21	53	65	POB
22	26	P22	54	66	DOB
23	27	P23	55	67	ODB
24	30	P24	56	70	SUB
25	31	P25	57	71	SOB
26	32	P26	58	72	SLR
27	33	P27	59	73	SSE
28	34	P28	60	74	SMA
29	35	P29	61	75	SZA
30	36	P30	62	76	UEB

7.10.2. Rozkazy ZZR (KOD=48, 8-bitowy parametr operacji)

WARTOŚĆ		NAZWA	WARTOŚĆ		NAZWA
DZIES.	OKT.	OPERACJI	DZIES.	OKT.	OPERACJI
1	01	PEB	33	41	STO
3	03	PPB	35	43	PPM
5	05	PZN	37	45	PRM
7	07	LMB	39	47	WBM
9	11	PDG	41	51	WMB
11	13	NZB	43	53	ZMB
13	15	PLA	45	55	PRB
15	17	CKA	47	57	NNR
17	21	LCD	49	61	NOR
19	23	PCD	51	63	OKZ
21	25	LAR	53	65	OAM
23	27	PAR	55	67	ZZN
25	31	LCA	57	71	ZBA
27	33	PCA	59	73	ZAM
29	35	LCM	61	75	NNR
31	37	PCM	63	77	NNR

Nie wymienione w tabelce wartości odpowiadają rozkazom NNR (Nic Nie Rób). Niektóre z wymienionych rozkazów są nielegalne.

7.10.3. Rozkazy OWW (KOD=63, 8-bitowy parametr operacji)

WARTOŚĆ		NAZWA	WARTOŚĆ		NAZWA
DZIES.	OKT.	OPERACJI	DZIES.	OKT.	OPERACJI
0	00	NAS	0	00	NAS
1	01	PZS	1	01	PZS
2	02	PRP	2	02	PRA
3	03	CRP	3	03	CRA
4	04	PTB	4	04	PTA
5	05	CTB	5	05	CTA
6	06	PBP	6	06	NNR
7	07	CBP	7	07	NNR

doty-  
czy  
kana-  
łu

doty-  
czy  
aryt-  
mome-  
tru

Wszystkie wymienione w tabeli rozkazy są nielegalne.

## 7.11. SPIS PRZYCZYŃ ZAKOŃCZENIA PROGRAMÓW UŻYTKOWNIKA

- 0-399 stop w programie
- 400 błędny parametr operacji WEOT
- 401 brak PAO dla programu
- 402 brak bębna dla programu
- 403 nie istnieje wskazany translator
- 404 próba wykonania TRAN przy wciśniętym kluczu 1
- 501 zbliżający się koniec taśmy magnetycznej podczas pracy PPT8
- 502 błędne sumy kontrolne bloków na taśmie magnetycznej wykryte przez PCT8
- 503 błędne parametry sekcji startowej
- 504 skok do 4 słowa PAO przy nieustawieniu tego słowa
- 505 skok do 5 słowa PAO przy nieustawieniu tego słowa
- 506 wykonanie jednego z rozkazów P06, P07, ..., P11 przy nieustawionej zawartości odpowiednio 6, 7 ... lub 11 słowa PAO
- 507 wykonanie rozkazu P31 dla wejścia lub wyjścia o niezdefiniowanym w czołówce dekodерze
- 508 wykonanie rozkazu P31 dla niedołączonego wejścia lub wyjścia
- 509 próba wyczyszczenia wejścia lub niedołączonego wyjścia
- 510 wykonanie rozkazu P31 dla wejścia lub wyjścia o niewłaściwie określonym dekodерze
- 511 wykonanie jednego z rozkazów P16, P17, ..., P30 przy nieustawionej zawartości odpowiednio 16, 17, ... lub 30 słowa PAO

- 512 nielegalny rozkaz lub SDY nietypowe z błędnym parametrem
- 513 wyczerpano czas zadeklarowany dla programu
- 514 program wyrzucony przez operatora
- 515 błędne parametry rozkazu SDY definiuj lub SDY wymień
- 516 próba wykonania rozkazu SDY inicjuj, dotyczącego niezdefiniowanego uprzednio przesłania
- 517 próba dołączenia niezdefiniowanego w czołówce wejścia lub wyjścia
- 518 próba dołączenia wejścia lub wyjścia po wyczerpaniu wolnych numerów przesłań
- 519 próba dołączenia wejścia lub wyjścia z błędnie określonym podprogramem lub dekoderm; również dołączenie źle wygenerowanego w systemie wejścia lub wyjścia
- 520 próba dołączenia wejścia lub wyjścia, gdy podprogram lub dekodery przekracza górną granicę PAO; również próba dołączenia wejścia lub wyjścia zerowego, gdy łączna długość podprogramu i dekodera przekracza 300
- 521 błędne parametry rozkazu SDY otwórz urządzenie
- 522 próba rozdefiniowania niezakończonego przesłania
- 523 błędne odwołanie do biblioteki systemu
- 524 odwołanie do taśmy magnetycznej nie otwartej lub otwartej do czytania
- 525 zapętlenie "adres do adresu" w rozkazie SDY
- 526 odwołanie do odłączonej taśmy magnetycznej systemu
- 527 w rozkazie SDY - czytaj monitor adres buforu plus długość buforu (32) przekraczają górną granicę PAO

- 529 błędne parametry rozkazu SDY - piaz  
masyw na taśmie magnetycznej lub SDY  
- czytaj masyw z taśmy magnetycznej
- 530 odwołanie do urządzenia, które nie  
jest taśmą magnetyczną, tak jak do  
taśmy magnetycznej
- 531 błędne parametry rozkazu SDY - podaj  
definicję przesłania
- 532 przekroczono listę T-etykiet podczas  
nadawania wartości T-etykiecie
- 540 próba pisania na zablokowanej taśmie  
magnetycznej lub inne błędy operowa-  
nia taśmą; również awarie taśmy magne-  
tycznej
- 541 błędne parametry rozkazu SDY podaj  
opis urządzenia

## 7.12. ZAWARTOŚĆ WYBRANYCH SŁÓW W PAO

Miejsca w PAO o adresach 0-855 są przewidziane dla pracy systemu operacyjnego. Miejsca te mogą być wykorzystywane przez program użytkowy inaczej niż niżej podano, jednakże wówczas działanie niektórych rozkazów może odbiegać od podanego opisu.

## 7.12.1. Przeznaczenie miejsc PAO 0-127 określone jest poniższą tabelką:

adres	przeznaczenie	stan początkowy
1	ślad rozkazu SLR; zawartość tego miejsca może być niszczone przez system operacyjny	nieokreślony
2	wykorzystuje rozkaz SUB	nieokreślony
3	wskaźniki; słowo to może być ładowane z monitora	nieokreślony
4	rozkaz, do którego następuje skok w przypadku przekroczenia zakresu normalizacji w działaniach zmiennoprzecinkowych	STO 504
5	rozkaz, do którego następuje skok w przypadku dzielenia zmiennoprzecinkowego przez zero	STO 505
16-30	wejście rozkazów programowanych P16, P17, ..., P30	STO 511
31	wejście rozkazu programowego P31, obsługującego symboliczne wejście i wyjście	SSL 128

38	adres w PAO ostatniego słowa dołączonego symbolicznego wejścia lub wyjścia zwiększony o 1	nieokreślony
57	adres w PAO, określający górną granicę dla dołączanych symbolicznych wejść i wyjść	pojemność PAO zestawu
123	początek obszaru w PAO dla podprogramów działających w czasie działania programu, a sterowanych przez system operacyjny	pojemność PAO
124	początek programu w PAO	standardowy początek programu w PAO
125	początek programu w PB	standardowy początek programu w PB
126	pojemność PAO zestawu	pojemność PAO zestawu
127	pojemność PB zestawu	pojemność PB zestawu
0 6-15 32-37 39-56 58-122	miejsca wykorzystywane przez system operacyjny; z punktu widzenia programu zmieniają się w sposób nieokreślony	nieokreślony



- 7.12.2. Miejsca 128-191 zawierają rozdzielacz i skorowidz symbolicznych wejść-wyjść. Zaleca się nie zmieniać zawartości tych miejsc, nawet jeśli program nie korzysta z symbolicznych wejść i wyjść.
- 7.12.3. Miejsca 192-255 zawierają listę dołączeń wejść i wyjść. Po dołączeniu wszystkich potrzebnych wejść i wyjść, program może wykorzystać ten obszar do własnych potrzeb (o ile nie korzysta z rozkazu SDY odłącz).
- 7.12.4. Miejsca 256-555 zawierają dołączony podprogram i dekodery dla symbolicznego wejścia 0. Program korzystający z symbolicznego wejścia 0, nie powinien zmieniać zawartości tych miejsc.
- 7.12.5. Miejsca 556-855 zawierają dołączony podprogram i dekodery dla symbolicznego wyjścia 0. Program korzystający z symbolicznego wyjścia 0, nie powinien zmieniać zawartości tych miejsc.



## 7.13.4. Taśma magnetyczna

Stosowana do maszyny ZAM 41 taśma magnetyczna jest nawinięta na dwa rodzaje szpul: szpule o średnicy ok. 26,5 cm (tzw. szpule duże) oraz szpule o średnicy ok. 17,5 cm (tzw. szpule małe). Ponieważ szpule małe są mniej stosowane, podano dla nich tylko niektóre dane.

	Szpule	
	duże	małe
Długość taśmy na szpuli ok.	750 m	200 m
ciężar szpuli z taśmą w pudełku	1,6 kg	0,5 kg
zewnętrzna średnica pudełka	30 cm	20 cm
wysokość pudełka	3 cm	3 cm
objętość pudełka	2150 cm <sup>3</sup>	950 cm <sup>3</sup>
objętość prostopadłościanu opisanego na pudełku	2700 cm <sup>3</sup>	1200 cm <sup>3</sup>
czas odczytu/zapisu	7 min	1 min 30 s
czas odwijania	3 min	45 s
pojemność w słowach zapisywanych blokami po 800 słów	3,1 M	0,8 M

Niektóre dane podają tablice porównawcze 5 i 6, punkt 7.13.5, str. 238 i 239.

## 7.13.5. Tablice porównawcze

## Tablica porównawcza

Parametry taśmy papierowej  
(wg średnicy zewnętrznej krawężka)

średnica zewnętrzna	10 cm	12,5 cm	15 cm	17,5 cm	20 cm (cały krawężek)
długość taśmy	55 m	100 m	150 m	220 m	300 m
ilość znaków	22 k	40 k	60 k	88 k	120 k
czas czytania	1 min 15 s	2 min 15 s	3 min 20 s	5 min	7 min
czas perforowania	22 s	40 s	1 min	1 min 30 s	2 min
ciężar	0,10kg	0,17 kg	0,25kg	0,35 kg	0,50 kg
	5-kanal.	0,16kg	0,27 kg	0,40kg	0,80 kg
	8-kanal.				

Parametry taśmy papierowej  
(wg ilości znaków)

Ilość znaków	10 k	50 k	100 k	200 k	500 k	1 M
długość taśmy	25 m	125 m	250 m	500 m	1250 m	2500 m
liczba krążków o średnicy zewnętrznej 15 cm		1	2	3	8	16
czas czytania	300 zn/s	3 min 20 s	4 min 30 s	10 min	25 min	50 min
1000 zn/s	35 s	1 min	2 min	3 min	8 min	16 min
czas perforacji brutto	150 zn/s	7 min	9 min	22 min	56 min	1 godz. 55 min
zmiana krążka						
ciężar	5-kanal.	0,20 kg	0,40 kg	0,80 kg	2,00 kg	4,00 kg
	8-kanal.	0,32 kg	0,64 kg	1,30 kg	3,15 kg	6,30 kg

1) razem z zakładaniem taśmy do perforatora

## Tablica porównawcza 3

## Parametry kart

(wg ilości znaków, przy założeniu 80 zn/karcie)

ilość znaków	10 k	50 k	100 k	200 k	500 k	1 M
liczba kart	125	625	1250	2500	6250	12500
liczba pudełek				1,5	3	6
czas czytania	20 s	1 min 30 s	3 min 10 s	6 min 20 s	16 min	31 min
objętość	350 cm <sup>3</sup>	1750cm <sup>3</sup>	3,5dcm <sup>3</sup>	7 dcm <sup>3</sup>	17,5dcm <sup>3</sup>	35 dcm <sup>3</sup>
ciężar	0,3 kg	1,6 kg	3,1 kg	6,2 kg	15,6 kg	31,2 kg

Tablica porównawcza 4

Parametry drukarki wierszowej

(orientacyjny czas drukowania zestawienia w zależności od liczby stron i liczby wierszy na stronie)

Liczba stron / liczba wierszy na stronie	40	50	60
100	7 min	9 min	10 min
150	11 min	13 min	15 min
200	14 min	17 min	20 min
250	18 min	22 min	25 min
500	35 min	43 min	50 min

## Tablica porównawcza 5

## Parametry taśmy magnetycznej

(pojemność dużej szpuli w słowach w zależności od długości bloku i krotności zapisu)

liczba słów w bloku	krotność zapisu		
	1	2	3
200	2,0 M	1 M	670 k
400	2,6 M	1,3 M	875 k
600	2,9 M	1,5 M	980 k
800	3,1 M	1,5 M	1,05 M
1000	3,2 M	1,6 M	1,09 M
1200	3,3 M	1,7 M	1,11 M
1400	3,4 M	1,7 M	1,13 M
1600	3,4 M	1,7 M	1,15 M
1800	3,4 M	1,7 M	1,16 M
2000	3,5 M	1,8 M	1,17 M



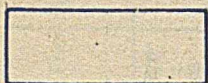
## Tablica porównawcza 6

## Parametry taśmy magnetycznej

(czas odczytu/zapisu różnej ilości słów w zależności od krotności zapisu)

ilość słów	krotność zapisu		
	1	2	3
100 k	15 s	30 s	45 s
250 k	35 s	1 min	1 min
500 k	10 s	10 s	45 s
	1 min	2 min	3 min
750 k	10 s	20 s	30 s
	1 min	3 min	5 min
1 M	45 s	30 s	15 s
1,5 M	2 min	4 min	7 min
	20 s	40 s	
2 M	3 min	7 min	
	30 s		
2,5 M	4 min		
	40 s		
3 M	5 min		
	50 s		
	7 min		

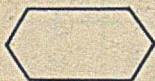
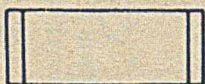
## 7.14. ZALECANE SYMBOLE GRAFICZNE DLA SCHEMATÓW BLOKOWYCH I SIECI DZIAŁAŃ



przetwarzanie



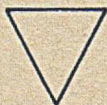
decyzja

przygotowanie  
(np. modyfikacja)  
rozkazuproces uprzednio  
określony

operacja ręczna



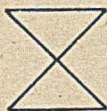
operacja pomocnicza



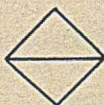
łączenie



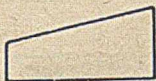
wybieranie



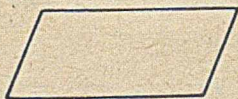
dobieranie



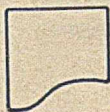
sortowanie



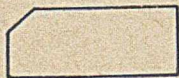
klawiaturowe wprowadzanie danych



wejście-wyjście



dokument, operacja  
na dokumencie czytel-  
nym dla człowieka



karta dziurkowana,  
operacje na kartach  
dziurkowanych



paczka kart dziur-  
kowanych, operacje  
na paczce kart  
dziurkowanych



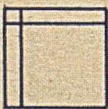
taśma dziurkowana,  
operacje na taśmie  
dziurkowanej



taśma magnetyczna  
(TM), operacje na  
taśmie magnetycznej



bęben magnetyczny  
(PB), operacje na  
bębnie magnetycznym



pamięć ferrytowa  
(PAO), operacje w  
pamięci ferrytowej



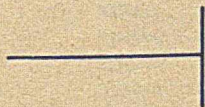
monitor dalekopisowy (MD), wprowadzanie/wyprowadzanie danych na MD



droga przepływu danych



skrzyżowanie dróg przepływu danych



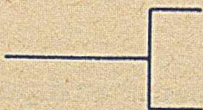
łączenie dróg przepływu danych



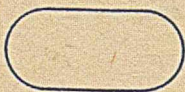
łącznik stronicowy



łącznik międzystronicowy



komentarz



początek, koniec  
lub przerwa

## 7.15. ŚCIĄGACZKI SO

Ściągaczki SO można przygotować perforując na taśmie papierowej w kodzie M-2 podane w punkcie 7.15.1 i 7.15.2 ciągi znaków; parami małych liter nu, fs i ls oznaczono odpowiednio znaki: pusty, cyfr i liter.

## 7.15.1. Ściągaczka SO z taśmy systemu

...nu	ZlsEEEX	ZAZFEE	ZlsQEBJ	ZlsAEEF	YEEKEU
	ZUAEEED	ZlsEESQ	ZAZFEE	ZlsQEBJ	ZlsAEEF
	YEDEYJ	ZUAEEA	EEDEEQ	ZASDEE	ZAZKEE
	ZlsEEEE	ZABDEB	ZKDEZY	Zfs 36;	+3 36
	+ 3/	+ 3+	33 3++	+2;2 [ ;	6 36
	+3;2[fs	+ / 36+	+ / [ 33	+;2[2	+ * (3[
	+;2[-	+ 2[fs	6lsAEEED	YlsWEES	EEDEDS
	EEDEDZ	ZlsEEEJ	ZWJEYE	ZJDWFA	ZDJEYD
	YlsJEEZ	ZDJEYB	YlsJEED	ZJDWFFs	6lsQEEB
	ZJDWFW	YlsWEEZ	EEDEBZ	EEDEDK	ZJDWFFs
	6lsWEEB	ZAXKEE	YlsWEED	EEDESA	ZJDWFW
	YlsWEES	EEDESA	ZlsEEEE	ZAXDEE	ZADKZY
	ZlsEZlsS	YSDJlsA	ZUJEBQ	ZAXKEE	ZXJUKQ
	ZAXKEE	EEDESA	ZAXFEE	ZXJUKK	ZAXKEE
	EEDESA	EEDJlsA	ZJDWFFs	6lsAEEEE	YlsWEES
	EEDESQ	EEDESJ	EDJEYE	ZASDBls	EEDEDF
	EEEEEE	EEEFEE	EEDJlsA	EKEZlsX	EEDAE
	EEEDAA	EEEDWE	EESAE	EESAA	EESWE
	EEEEEZ	nu...			

## 7.15.2. Ściągaczka SO z bębna

...nu	ZlsEEZZ	ZAZFEE	ZlsQEBJ	ZlsAEEF	YEEEFY
	ZUAEEED	ZEEEEEE	ZlsJEEW	ZlsJDSE	ZlsJEEW
	ZlsADEB	ZlsZXXX	ZAXDEE	ZlsJEEJ	ZDEEZB
	ZEWAE	ZDEEZY	ZEWAE	EEZAE	ZlsQDSE
	ZlsWDRE	EEEEEE	EEZARZ	nu...	

## 7.16. NAZWY UŻYWANE W ZDANIACH SUP, ZMI, UZU I DEK

Użytkownik korzystający z taśmy systemu dostarczonej przez IMM może stosować w zdaniach SUP, ZMI, UZU i DEK następujące nazwy:

w zdaniu SUP: SV, SVA, SVB, SVC

w zdaniu ZMI i UZU: ALGO, ASTE, AST1, AST2,

BCOB, BIB1, BIB2, CBT1, CBT2, CBT3,

CBT4, CBT5, CBT6, CBT7, CBT8, CBT9,

CB10, CB11, CB14, CEMA, COBO, COP1,

DRUP, DWSE, EDAL, EDCO, EOL, FUNA,

FUNB, FUN1, FUN2, FUN3, FUN4, FUN5,

FUN6, FUN7, FUN9, FUVA, FUVB, FUVB,

GPSS, GPS1, GPS2, GPS3, GPT1, GPWY,

GRAF, JOM, MAKR, MAN, MAN1, MSAS,

PJEG, PLEX, PPO1, PPO3, PPO9, PP10,

PP11, PP12, PP13, PP14, PP15, PP16,

PTSE, POPR, REST, SAKO, SMAO, SPSE,

TEST, VCTA, VKZP, VMEP, VMKD, VMKZ,

VPPO, VPTA, VSCP, VWTS, VWYD, VWYP,

VZBT, WEBI, WYSE, ZAKO, ZCOB

w zdaniu DEK: SWW, SWW1, SWW2



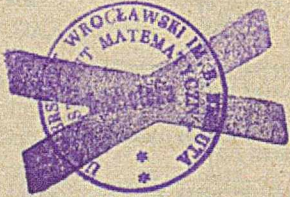
7.17. WYKAZ NIEKTÓRYCH OŚRODKÓW WYPOSAŻONYCH  
W MASZYNĘ ZAM 41

Nazwa i adres ośrodka	ILOŚĆ SZŁÓW		LICZBA JEDN. TM	CK	DW
	PAO	PB			
<u>Województwo katowickie</u>					
Biuro Projektów Przemysłu Syntezy Chemicznej "Prosynchem", Pracownia Techniki Obliczeniowej					
ul. Marcina Strzody 11, 44-100 Gliwice, tel. 91-10-81	20k	64k	7	CK3	DW1 DW2B
<u>Województwo krakowskie</u>					
Zakład Elektronicznej Techniki Obliczeniowej Instytutu Zootechniki					
32-083 Balice k. Krakowa, tel. 32-397	20k	64k	7	CK3 BETA	DW2
<u>Województwo łódzkie</u>					
Miejskie Przedsiębiorstwo Komunikacyjne, Ośrodek Elektronicznej Techniki Obliczeniowej					
ul. Tuwima 22/26 90-002 Łódź, tel. 84-861	20k	32k	7	CK3 BETA	DW2B
Zakład Elektronicznej Techniki Obliczeniowej					
ul. Hutora 69 90-558 Łódź, tel. 32-698	20k	64k	7	CK3 BETA	DW2 2 szt.
<u>Województwo warszawskie</u>					
Branżowy Ośrodek Zastosowań Elektronicznej Techniki Obliczeniowej Zjednoczenia Przemysłu Farmaceutycznego POLFA BOZETO POLFA					
ul. Al. Fleminga 2, 03-150 Warszawa 91, tel. 11-18-27	20k	64k	7	CK3 BETA	DW2

202/2

45-

Dział Przetwarzania Infor- macji Huty Warszawa	20k	64k	7	CK3	DW2B
ul. Kasprowicza 132, 01-949 Warszawa, tel. 35-96-70					
Ośrodek Obliczeniowy Insty- tutu Technicznego Wojsk Lotniczych,	20k	64k	7	CK2	DW2
00-911 Warszawa 62					
Ośrodek Obliczeniowy Wojs- kowej Akademii Technicznej,	20k	128k	7	CK3 CK3 BETA	DW1 DW2E
00-908 Warszawa 49					
Stołeczny Ośrodek Elektro- nicznej Techniki Obliczenio- wej SOETO					
ul. Królewska 27 00-060 Warszawa, tel. 26-86-33					
	1/	20k	64k	7	- DW2
	2/	20k	64k	7	CK3 BETA 2 szt.
Zakład Elektronicznej Tech- niki Obliczeniowej Instytu- tu Ekonomiki Rolnej					
ul. Koszykowa 6 00-564 Warszawa, tel. 29-97-82	20k	96k	5	CK3 BETA	DW2



S/202