

ZAKŁAD DOŚWIADCZALNY MINIKOMPUTERÓW

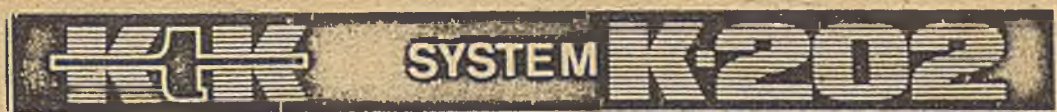
(Opis wyłącznie do celów szkoleniowych)

MASZYNA CYFROWA
K-202
ORGANIZACJA LOGICZNA

OŚRODEK DOSKONALENIA KADR TECHNICZNYCH OW/NOT

Warszawa

1973



ZAKŁAD DOŚWIADCZALNY MINIKOMPUTERÓW

(Opis wyłącznie do celów szkoleniowych)

MASZYNA CYFROWA
K-202
ORGANIZACJA LOGICZNA

OŚRODEK DOSKONALENIA KADR TECHNICZNYCH O W/NOT

Warszawa

1973

SPIS TREŚCI

	Str.
1. POSTAĆ INFORMACJI W MASZYNIE	3
2. SCHEMAT BLOKOWY MASZINY	5
3. REJESTRY I WSKAŹNIKI MASZINY	9
4. OGÓLNE ZASADY WYKONYWANIA ROZKAZÓW	13
4.1. Pobranie rozkazu	14
4.2. Określenie efektywności rozkazu	14
4.3. Wyznaczenie efektywnego argumentu	15
4.4. Wykonanie operacji określonej rozkazem	19
5. LISTA ROZKAZÓW	19
6. STANY MASZINY	28
7. PULPIT TECHNICZNY MASZINY	29

1. POSTAĆ INFORMACJI W MASZYNIE

Maszyna K-202 pracuje w systemie binarnym. Dla zapisu liczb i w arytmetyce stosowana jest notacja uzupełnieniowa do dwóch. Słowo maszyny zawiera 16 bitów numerowanych od 0 do 15 i służy do przedstawiania następujących informacji:

- L i c z b a s t a ł o p r z e c i n k o w a k r ó t k a -
k a zajmuje jedno słowo i traktowana jest jako liczba całkowita /Rys. 1a/.
- I n f o r m a c j a l o g i c z n a zajmuje jedno słowo i traktowana jest jako ciąg 16 niezależnych bitów /Rys. 1b/.
- Z n a k i a l f a n u m e r y c z n e. Ich długość, rozmieszczenie w słowie oraz kody określone są programem. W szczególności stosowane są znaki 7 lub 8-bitowe umieszczone po dwa w słowie /Rys. 1c/, zapisane w kodzie ISC-7.
- L i c z b a s t a ł o p r z e c i n k o w a d ł u g a określona jest programem. W szczególności stosowana jest postać, w której liczba długa zajmuje dwa słowa i traktowana jest jako liczba całkowita /Rys. 1d/.
- L i c z b a z m i e n n o p r z e c i n k o w a zajmuje trzy słowa /Rys. 1e/. Cecha zajmuje pierwsze słowo i jest liczbą stałoprzecinkową. Mantysa zajmuje dwa następne słowa i traktowana jest jako liczba zaprzecinkowa w notacji uzupełnieniowej do dwóch.
- A d r e s jest 16-bitową, całkowitą liczbą binarną z przedziału 0-65536.
- R o z k a z zajmuje od jednego do czterech słów i może posiadać kilka postaci /Rys. 2/.

a/ Rozkaz w podstawowej postaci zajmuje jedno słowo nazywane słowem podstawowym /Rys. 2a/, które zawiera następujące informacje:

- KOD OPERACJI /poz.0-4/ - pole 5-bitowe pozwalające na rozróżnienie 32 rozkazów podstawowych i grup rozkazowych. W wielu przypadkach część pozostałych pól rozkazu stanowi przedłużenie kodu operacji, co pozwala na faktyczne zwiększenie pełnej ilości rozkazów do ponad 90.
- D /poz.5/ - zawiera bit pośredniego argumentu lub znak argumentu operacji lub stanowi przedłużenie kodu operacji,
- W /poz.6/ - bit wskazujący, że rozkaz jest warunkowy,
- A /poz.7-9/ zawiera numer rejestru uniwersalnego lub stanowi przedłużenie kodu operacji,
- B /poz.10-12/ - zawiera numer rejestru indeksowego służącego do B-modyfikacji argumentu lub stanowi przedłużenie kodu operacji,
- C /poz.13-15/ - służy do wyznaczenia argumentu rozkazu lub stanowi przedłużenie kodu operacji.

W niektórych przypadkach pola D, B i C łącznie zawierają bezpośrednio 7-bitowy argument rozkazu.

b/ Rozkaz warunkowy - /W=1/ - /Rys. 2b, e, f, h/ posiada pole warunków WC, które jest informacją logiczną zajmującą następne słowo za słowem podstawowym rozkazu.

c/ Rozkaz z polem M - /C=0/ - /Rys. 2c, e, g, h/. Pole M jest liczbą stałoprzecinkową i zajmuje następne słowo

za polem WC, jeśli rozkaz posiada je, lub za słowem podstawowym rozkazu w przeciwnym przypadku. Pole M zawiera pierwotny argument rozkazu.

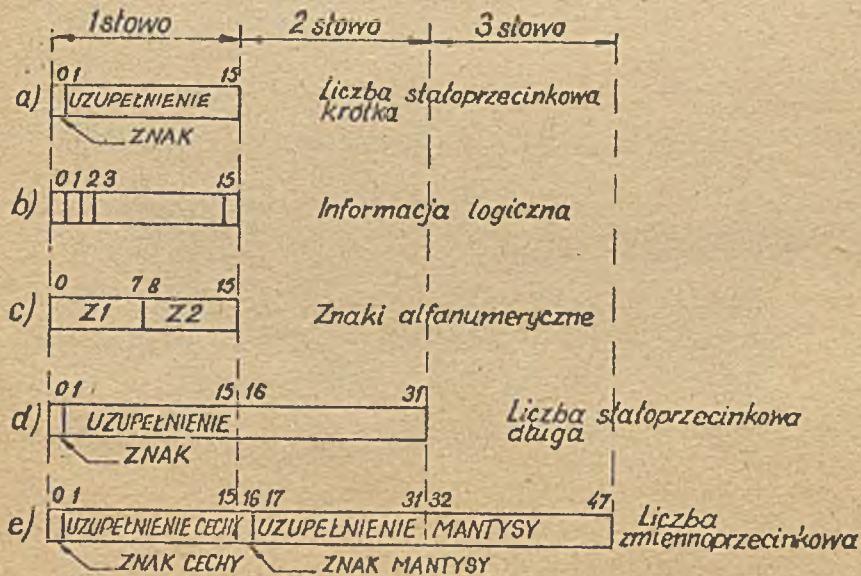
d/ Rozkaz ex - wywołanie ekstrakodu /Rys. 2d, f, g, h/ posiada pole NAZWA w następnym słowie za pozostałymi polami rozkazu. NAZWA zajmuje pozycje 9-15 tego słowa i jest całkowitą, nieujemną liczbą binarną. Pozostałe pozycje są ignorowane.

2. SCHEMAT BLOKOWY MASZINY

Ogólny schemat blokowy maszyny przedstawiony jest na Rys. 3.

Maszyna K-202 składa się z następujących modułów:

- Procesor, podstawowy moduł maszyny. W skład maszyny może wchodzić do 4 procesorów.



Rys.1. Postać informacji w K-202

Każdy moduł Procesora zawiera:

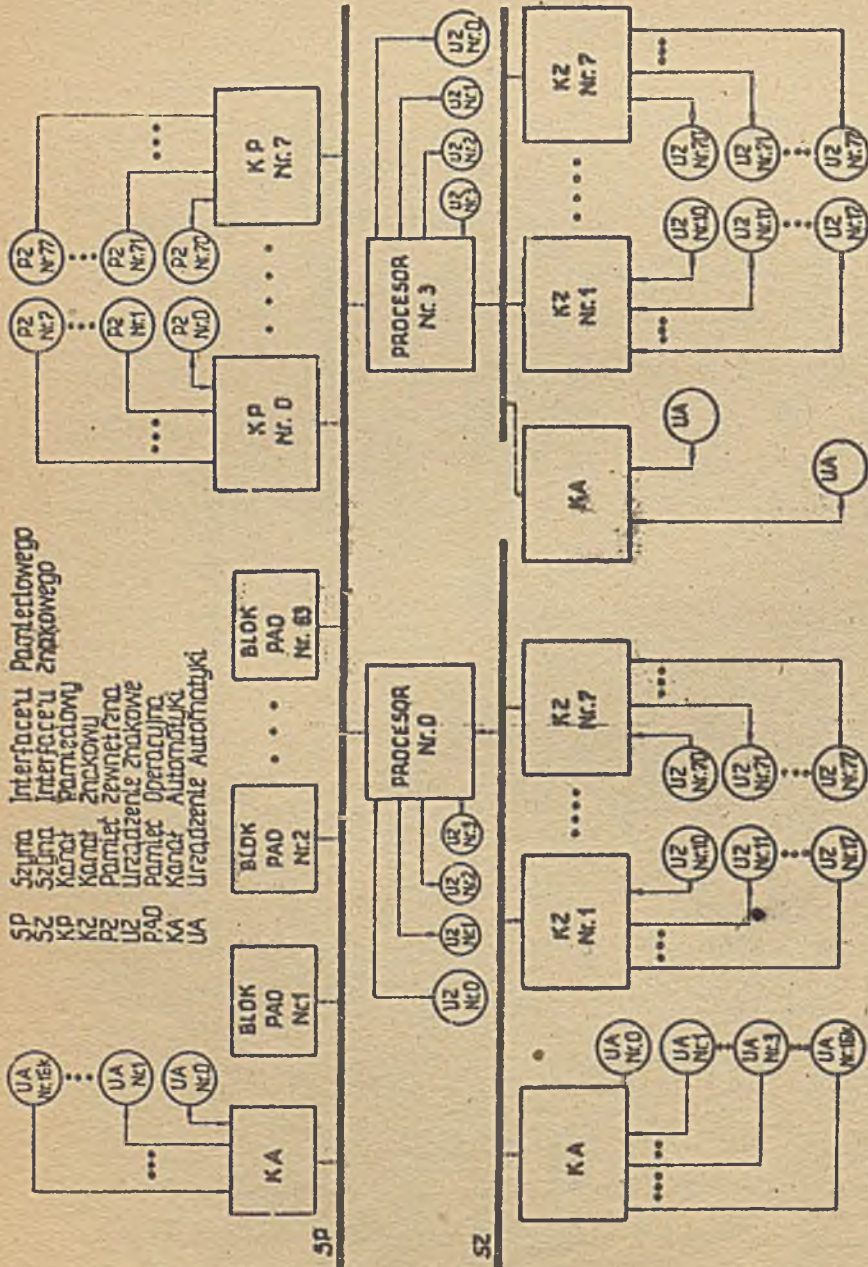
- Jednostkę Centralną - podstawową jednostkę przetwarzania informacji.
- Blok systemowy i blok użytkowy Pamięci Operacyjnej o łącznej pojemności do 16k słów.
- Kanał Znakowy urządzeń WE-Wy.
- Bloki Pamięci Operacyjnej /PAO/ w ilości do 64, numerowane od 0 do 63. Blok zerowy może mieć pojemność 4k, 8k, 12k lub 16k słów 16-bitowych i mieści się w Procesorze. Dalsze bloki mogą mieć pojemność 16k, 32k lub 64k słów każdy.
- Kanały Pamięciowe w ilości do 8, służące do połączenia modułów Pamięci Zewnętrznych z Pamięcią Operacyjną i Procesorami.
- Pamięci Zewnętrzne. Do każdego Kanału Pamięciowego można dołączyć do 8 modułów Pamięci Zewnętrznej; takich jak dyski, bębny, taśmy magnetyczne itp.
- Kanały Znakowe w ilości do 8, służące do połączenia Urządzeń Znakowych z Procesorem. Kanał Nr 0 mieści się w Procesorze.
- Urządzenia Znakowe - urządzenia WE-WY. Do każdego Kanału Znakowego można dołączyć do 8 urządzeń We-Wy, w sumie 64 urządzenia.

Pomiędzy modułami istnieją następujące standardowe połączenia:

- Interfejs Pamięciowy jest łączem standardowym pomiędzy Kanałami Pamięciowymi, blokami PAO i Procesorami. Do Interfejsu Pamięciowego można dołączyć do 4 Procesorów, co umożli-

	1 słowo	2 słowo	3 słowo	4 słowo
a)	0 45 67 9 10 2 13 5 KOD OP DM A B C			
b)	0 45 67 9 10 2 13 5 KOD OP DM A B C	WC		
c)	0 45 67 9 10 2 13 5 KOD OP DM A B C	M		
d)	0 45 67 9 10 2 13 5 KOD OP DM A B C	89 NAZWA		
e)	0 45 67 9 10 2 13 5 KOD OP DM A B C	WC		
f)	0 45 67 9 10 2 13 5 KOD OP DM A B C	WC	89 NAZWA	
g)	0 45 67 9 10 2 13 5 KOD OP DM A B C	M	89 NAZWA	
h)	0 45 67 9 10 2 13 5 KOD OP DM A B C	WC	M	89 NAZWA

Rys. 2 - Postać rozkazu K-202



Rys. 3 - Schemat blokowy maszyny K-202

liwia pracę wieloprocesorową, 64 bloki PAO, 8 kanałów Pa -
mięciowych oraz kanały automatyki i pracy na bieżąco - do
16k sygnałów /punktów pomiarowo-kontrolnych/.

- Interface Znakowy jest łączem standardowym pomiędzy Proce-
sorem a Kanałami Znakowymi urządzeń wejścia-wyjścia. Do
Interface'u Znakowego można dołączyć do 8 Kanałów Znako -
wych oraz kanały automatyki i pracy w czasie rzeczywistym
- do 16k sygnałów.

Każdy Interface posiada ustaloną ilość przewodów, słu-
żących do przesyłania informacji, adresów i sygnałów sterują
cych.

3. REJESTRY I WSKAŹNIKI MASZYNY

W maszynie K-202 dostępne są dla programisty lub wpły-
wają na przebieg pracy następujące rejestry i wskaźniki:

- Rejestry i wskaźniki Jednostki Centralnej,
- Rejestry i wskaźniki Urządzeń Wejścia - Wyjścia,
- Pamięć Operacyjna maszyny.

a/ Rejestry i wskaźniki Jednostki Centralnej /Rys. 4a/

Symbol Nazwa i funkcja

RO Rejestr stanu maszyny 16-bitowy. Stanowi zbiór wskaź-
ników decydujących o sposobie wykonywania programu.
Pozycje rejestru RO oznaczają:
Q /poż./ - wskaźnik systemu. Stan Q=1 odpowiada pra-
cy programu normalnego, stan Q=0 - pracy Systemu

Operacyjnego. Stan Q wpływa na sposób adresowania PAO, sposób wykonania rozkazów nielegalnych oraz sposób zapisu do rejestru RO.

L,E,G /poz.1-3/ wskaźniki ustawiane w wyniku operacji porównań arytmetycznych.

V /poz.4/ - wskaźnik nadmiaru, do którego wpisywane jest "1" w przypadku przekroczenia zakresu liczb przy operacjach arytmetycznych i w niektórych innych przypadkach.

C /poz.5/ - wskaźnik przeniesienia, ustawiany zgodnie z przeniesieniem z zerowej pozycji arytmetru przy operacjach arytmetycznych.

Y /poz.6/ - wskaźnik przechowujący bit wychodzący poza rejestr przy operacjach przesuwania.

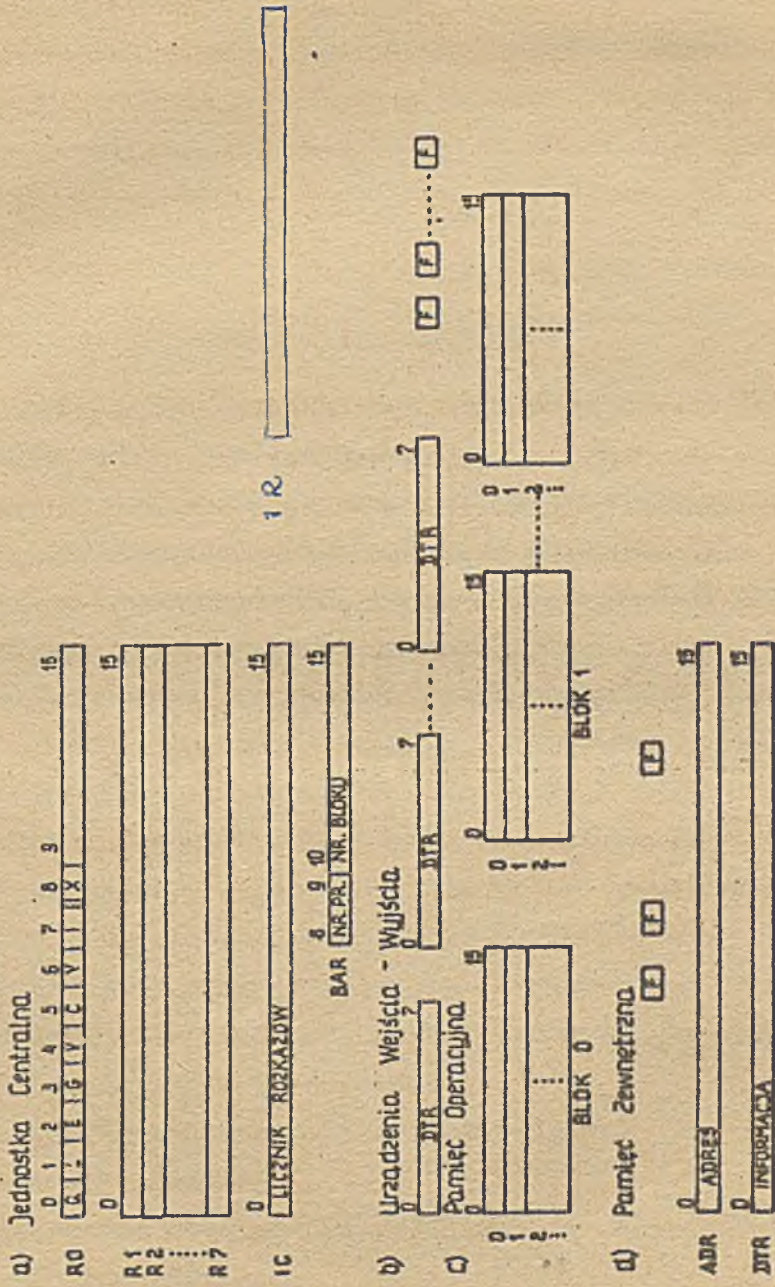
I /poz.7/ - maska przerwania zewnętrznych. Gdy I=0 zgłoszenia przerwania zewnętrznych są przyjmowane, ale przerwania nie są realizowane.

X /poz.8/ - wskaźnik ustawiany programowo,

- poz.9-15 - wskaźniki przeznaczone do użytku programisty.

Rejestr RO może być używany w programach jako rejestr wynikowy operacji, ale wówczas zapis do pozycji 0-7 rejestru RO przy stanie Q=1 nie jest wykonywany.

R1-R7 Rejestry Uniwersalne - 16-bitowe, podstawowe rejestry używane do przechowywania informacji, wykonywania operacji oraz używane jako rejestry indeksowe służące do B-modyfikacji argumentów.



Rys. 4 - Rejestry i wskaźniki maszyny K-202

- IC Licznik Rozkazów 16-bitowy, zawierający adres miejsca Pamięci Operacyjnej, z którego pobrany zostanie rozkaz.
- BAR Rejestr Numeru Bloku 8-bitowy. Pozycje 8-9 rejestru BAR zawierają numer Procesora wykorzystywany przy pracy wieloprocesorowej, pozycje 10-15 wskazują numer bloku PAO, z którym odbywa się współpraca.
- P Wskaźnik Przeskoku. Ustawienie wskaźnika P w stan 1 powoduje ominięcie najbliższego rozkazu.

b/ Rejestry i wskaźniki Urządzeń Wejścia - Wyjścia /Rys. 4b/. Każde Urządzenie Znakowe posiada 8-bitowy rejestr informacji DTR, z którego można pobrać lub do którego można zapisać informację rozkazem z Jednostki Centralnej. Ponadto, każde Urządzenie Znakowe może mieć pewną ilość wskaźników P ustawianych programowo.

c/ Pamięć Operacyjna - PAO /Rys. 4c/.

Pamięć Operacyjna składa się z bloków w ilości od 1 do 64, numerowanych od 0 do 63.

Pojemność zerowego bloku może wynosić 4K, 8K, 12K lub 16K, a pojemność każdego z dalszych bloków 16K, 32K, lub 64K słów 16-bitowych.

Określenia miejsca Pamięci dokonuje się przez podanie dwóch współrzędnych, to znaczy numeru bloku oraz adresu słowa wewnątrz bloku, zwanego w dalszym ciągu krótko adresem. Odwołanie do nie istniejącego słowa PAO rozpoznawane jest jako błąd i powoduje przerwanie.

d/ Pamięci Zewnętrzne /Rys. 4d/

Każdy moduł Pamięci Zewnętrznej posiada 16-bitowy Rejestr Adresu /ADR/ oraz 16-bitowy Rejestr Informacji DTR, dostępne programowo. Ponadto każdy moduł może posiadać pewną ilość wskaźników F ustawianych programowo.

4. OGÓLNE ZASADY WYKONYWANIA ROZKAZÓW

Pełny cykl czynności maszyny związanych z wykonaniem każdego rozkazu jest następujący:

- pobranie rozkazu z miejsca Pamięci Operacyjnej wskazanego przez aktualną zawartość Licznika Rozkazów IC,
- określenie efektywności rozkazu /warunek W, P i inne/,
- wyznaczenie efektywnego argumentu,
- wykonanie operacji określonej kodem operacji rozkazu.

W skład czynności związanych z wykonaniem cyklu rozkazowego wchodzi ustalenie nowej zawartości Licznika Rozkazów. W przypadku gdy nie jest to rozkaz typu skokowego i nie zachodzą przypadki szczególne, polega ono na zwiększeniu zawartości IC o długość rozkazu /liczoną w słowach/.

Po zakończeniu każdego cyklu rozkazowego w zależności od warunków zaistniałych w maszynie następuje przejście do jednej z następujących czynności:

- wykonanie następnego cyklu rozkazowego,
- wykonanie przerwania,
- zatrzymanie maszyny, która przechodzi wówczas w stan STOP, lub w stan CZEKAJ.

4.1. Pobranie rozkazu

Rozkaz pobierany jest z bloku zerowego Pamięci Operacyjnej gdy wskaźnik $Q=0$ lub z bloku o numerze wskazanym za wartością pozycji 10-15 rejestru BAR gdy $Q=1$.

Adres pierwszego słowa rozkazu wskazany jest zawartością Licznika Rozkazów. Ilość pobranych słów zależy od długości rozkazu.

4.2. Określenie efektywności rozkazu

Rozkaz jest nieefektywny, to znaczy nie zostaje wykonany w następujących przypadkach:

a/ Rozkaz nieprawidłowy. Rozpoznanie nieprawidłowego rozkazu może być spowodowane jedną z trzech przyczyn:

- Błędny kod operacji,
- Czwartą raz z rzędu użyty rozkaz mod /modyfikuj/,
- Rozkaz nielegalny użyty przy wskaźniku $Q=1$.

Wykrycie nieprawidłowego rozkazu powoduje wykonanie w maszynie następujących czynności:

- wpisanie "0" do wskaźnika przeskoku P,
- wyzerowanie modyfikatora /MOD/,
- wpisanie "1" do odpowiedniego Wskaźnika Zgłoszenia Przerwania,
- natychmiastowe zakończenie rozkazu.

Po wykryciu nieprawidłowego rozkazu żadne inne warunki /pole WC, wskaźnik P itp./ nie są badane. Zawartość rejestrów programowych i pamięci maszyny nie ulega zmianie. Li

cznik Rozkazów w przypadku błędnego kodu operacji rozkazu zawiera adres pierwszego słowa tego rozkazu zwiększony o 1. W obu pozostałych przypadkach Licznik Rozkazów zawiera adres pierwszego słowa następnego rozkazu.

b/ Wskaźnik przeskoku P zawiera "1".

c/ Rozkaz warunkowy /bit $W=1$ / przy zawartości rejestru stanu maszyny RO niezgodnej z zawartością pola WC. Wykrycie niezgodności następuje, jeśli istnieje pozycja taka, że RO zawiera "0" a WC zawiera "1" na tej pozycji.

W obu przypadkach b/ i c/ następuje wykonanie w maszynie następujących czynności:

- wpisanie "0" do wskaźnika P,
- wyzerowanie modyfikatora,
- natychmiastowe zakończenie rozkazu.

Zawartość rejestrów programowych i pamięci maszyny nie ulega zmianie. Licznik Rozkazów zawiera adres pierwszego słowa następnego rozkazu.

4.3. Wyznaczenie efektywnego argumentu

Argument efektywny rozkazu jest to argument uzyskany z argumentu pierwotnego rozkazu przez wykonanie wszystkich wskazanych modyfikacji i jest zawsze pełną liczbą 16-bitową. Jeśli żadna z modyfikacji nie jest wykonywana, argumentem efektywnym jest argument pierwotny rozkazu wydłużony w razie potrzeby do pełnej 16-bitowej długości. Argument efektywny wyznaczany jest na kilka sposobów zależnie od postaci argumentu pierwotnego rozkazu.

Argument pierwotny rozkazu może należeć do jednego z następujących rodzajów:

Argument normalny jest liczbą 16-bitową znajdującą się w miejscu wskazanym przez pole C rozkazu:

- Jeśli $C=0$, argument stanowi zawartość pola M rozkazu,
- Jeśli $C \neq 0$, argument stanowi zawartość rejestru uniwersalnego o numerze wskazanym polem C.

Argument krótki jest liczbą 7-bitową zapisaną w notacji znak - wartość bezwzględna i umieszczoną bezpośrednio w rozkazie:

- bit D zawiera znak argumentu,
- pola B i C zawierają łącznie 6-bitową wartość bezwzględną argumentu.

Krótki argument jest przekształcany do postaci pełnej, 16-bitowej liczby stałoprzecinkowej.

Modyfikacje argumentu

W maszynie K-202 możliwe są 3 różne modyfikacje argumentu: pre-modyfikacja, B-modyfikacja i D-modyfikacja.

Są one wykonywane w podanej niżej kolejności:

- a/ pre-modyfikacja możliwa jest w każdym rozkazie i ma miejsce wówczas, gdy zostanie on poprzedzony rozkazem `mod /modyfikuj/`. Argument efektywny rozkazu `mod`, nazywany modyfikatorem `/MOD/`. Pre-modyfikacja polega na dodaniu modyfikatora do argumentu pierwotnego rozkazu.

Dodawanie jest wykonywane z zaniedbaniem ewentualnego nadmiaru. Rozkaz `mod` może być również poprzedzony rozkazem

mod, ale czwarty kolejny rozkaz mod jest traktowany jako nieprawidłowy i nie jest wykonywany, natomiast powoduje przerwanie programu.

b/ B-modyfikacja jest możliwa w rozkazach, w których pole B oznacza numer rejestru indeksowego. Polega ona na dodaniu zawartości rejestru indeksowego do argumentu rozkazu. Dodawanie jest wykonywane z zaniedbaniem ewentualnego nadmiaru. Jako rejestry indeksowe mogą być użyte Rejestry R1-R7. Zerowa zawartość pola B oznacza brak B - modyfikacji. B-modyfikacja nie jest możliwa w rozkazach z krótkim argumentem.

c/ D-modyfikacja jest możliwa w rozkazach, w których D oznacza bit argumentu pośredniego.

Polega ona na tym, że jeśli $D=1$, wówczas uzyskany w wyniku poprzednich modyfikacji argument traktowany jest jako adres argumentu efektywnego.

W celu otrzymania argumentu efektywnego dokonywany jest odczyt z bloku zerowego Pamięci Operacyjnej gdy $Q=0$, lub z bloku o numerze wskazanym zawartością pozycji 10-15 rejestru BAR gdy $Q=1$. Jeśli $D=0$, argumentem efektywnym jest argument wyznaczony w wyniku poprzednich modyfikacji. D-modyfikacja nie jest możliwa w rozkazach z krótkim argumentem, w rozkazach przesyłania grupowego /log i stg/.

Zestawienie przypadków wyznaczania argumentu efektywnego.

a/ Dla rozkazów o normalnym argumencie /z wyjątkiem log, stg/ możliwe są wszystkie modyfikacje.

Zestawienie dla nich przedstawia poniższa tabela.

- b/ Dla rozkazów log, stg, obowiązują tylko te wiersze tabeli, w których nie występuje D-modyfikacja.
- c/ W rozkazach o krótkich argumentach możliwa jest tylko modyfikacja za pomocą modyfikatora.
- d/ W rozkazach nie posiadających drugiego argumentu operacja wyznaczania efektywnego argumentu nie jest wykonywana.

MODYFIKACJE			C=0	Argument efektywny
MOD	B /B;≠0/	D /D=1/		
n	n	n	n	$N = R(C)$
n	n	t	n	$N = S(R(C))$
n	t	n	n	$N = R(C) + R(B)$
n	t	t	n	$N = S(R(C) + R(B))$
t	n	n	n	$N = R(C) + MOD$
t	n	t	n	$N = S(R(C) + MOD)$
t	t	n	n	$N = R(C) + MOD + R(B)$
t	t	t	n	$N = S(R(C) + MOD + R(B))$
n	n	n	t	$N = M$
n	n	t	t	$N = S(M)$
n	t	n	t	$N = M + R(B)$
n	t	t	t	$N = S(M + R(B))$
t	n	n	t	$N = M + MOD$
t	n	t	t	$N = S(M + MOD)$
t	t	n	t	$N = M + MOD + R(B)$
t	t	t	t	$N = S(M + MOD + R(B))$

4.4. Wykonanie operacji określonej rozkazem

Po wyznaczeniu argumentu efektywnego następuje przejście do wykonania operacji określonej kodem operacji rozkazu. Maszyna K-202 posiada rozkazy dwuargumentowe, jedno argumentowe i bezargumentowe. Pierwszym argumentem operacji jest zawsze zawartość rejestru uniwersalnego wskazanego po - lem A rozkazu lub zawartość komórki pamięci operacyjnej. Drugim argumentem jest argument efektywny rozkazu. Rozkazy jednoargumentowe mogą posiadać tylko pierwszy argument lub tylko drugi argument.

Wykonanie operacji odbywa się zgodnie z opisem zawartym w rozdziale 5. Operacja wykonywana jest zawsze do zakończenia zgodnego z opisem. Jedynym wyjątkiem jest przypadek, gdy operacja zawiera odwołanie do nieistniejącego słowa Pamięci Operacyjnej lub Urządzenia Zewnętrznego. Wówczas zawartość rejestru wynikowego operacji jest nieokreślona i następuje przerwanie programu.

5. LISTA ROZKAZÓW

Poniższe zestawienie zawiera wszystkie rozkazy maszyny K-202 zebrane grupami w zależności od znaczenia poszczególnych pól rozkazu.

Przy opisie każdego rozkazu obowiązują następujące zasady:

- 1/ Jeżeli w opisie rozkazu nie jest wymieniony Licznik Rozkazów to przyjmujemy zawsze, że rozkaz ten powoduje zwiąk -

szenie zawartości IC o długość rozkazu. Ponadto wynikiem rozkazu są tylko te zmiany, które wynikają jednoznacznie z treści rozkazu,

2/ Adresowanie miejsc Pamięci Operacyjnej dokonywane jest następująco:

$S(x; y)$ oznacza zawartość komórki o adresie x w bloku Pamięci o numerze y

$S(x)$ oznacza zawartość komórki pamięci o adresie x w bloku zerowym gdy $Q=0$, lub w bloku o numerze wskazanym pozycjami 10-15 rejestru BAR gdy $Q=1$,

3/ Symbole R_0, \dots, R_7, IC, S itp. oznaczają zawartość rejestrów programowych i miejsc Pamięci, na przykład $R(A)$ oznacza zawartość rejestru uniwersalnego o numerze wskazanym polem A rozkazu,

4/ Indeksy przy symbolach rejestrów oznaczają numery pozycji, na przykład $R_5 [0-8, 10-15]$ oznacza "zawartość pozycji: od 0 do 8 oraz od 10 do 15 rejestru o numerze 5",

5/ Symbolem argumentu efektywnego jest N i w zależności od rozkazu może oznaczać liczbę stałoprzecinkową, informację logiczną, parametr rozkazu WE-WY lub adres,

6/ Użyte symbole logiczne mają następujące znaczenie:

- nad symbolem oznacza negację

\wedge oznacza iloczyn logiczny

\vee oznacza sumę logiczną

\oplus oznacza różnicę symetryczną

\equiv oznacza tożsamość,

7/ Przez funkcję logiczną określoną na dwóch słowach rozumiemy funkcję wykonaną na każdej pozycji niezależnie,

- 8/ Jeśli funkcja logiczna określona na dwóch słowach występu je jako warunek, to uważamy ten warunek za spełniony, jeśli spełniony jest na wszystkich pozycjach równocześnie,
- 9/ Napis "ust.V" oznacza, że jeśli w wyniku wykonania wskazanej operacji występuje nadmiar, do wskaźnika V wpisywana jest "1", w przeciwnym razie V pozostaje bez zmiany. W przypadku wystąpienia nadmiaru wynik operacji może być nieprawidłowy,
- 10/ Napis "ust.C" oznacza, że do wskaźnika C wpisywana jest wartość przeniesienia z zerowej pozycji arytmometru w czasie wykonania wskazanej operacji,
- 11/ MOD oznacza modyfikator,
- 12/ Jeżeli RO jest rejestrem wynikowym wskazanym przez pole A oraz w rozkazie reex, zapis do $RO[0-7]$ przy wskaźniku $Q=1$ nie jest wykonywany,
- 13/ $R\uparrow$ oznacza przesunięcie o jeden bit w lewo, to znaczy operację $Y := R[0]; R[0-14] := R[1-15];$
 $R\downarrow$ oznacza przesunięcie o jeden bit w prawo, to znaczy operację $Y := R[15]; R[1-15] := R[0-14].$
- 14/ Jeżeli w wykonaniu operacji używana jest zawartość IC, to przyjmuje się, że jest ona równa pierwszemu adresowi następnego rozkazu.

LISTA ROZKAZÓW

A. Rozkazy dwuargumentowe

Pola rozkazu oznaczają:

- A - numer rejestru uniwersalnego
- B - numer rejestru indeksowego
- C - wskazuje argument normalny
- D - bit argumentu pośredniego

<u>Skrót</u>	<u>Nazwa</u>	<u>Treść</u>
ad	Dodaj	$R(A) := R(A) + N$; ust. V, C
adc	Dodaj z przeniesieniem	$R(A) := R(A) + N + C$; ust. V, C
su	Odejmij	$R(A) := R(A) - N$; ust. V, C
co	Porównaj	Jeśli $R(A) < N$ to $L:=1, E:=0, G:=0$; Jeśli $R(A) = N$ to $L:=0, E:=1, G:=0$; Jeśli $R(A) > N$ to $L:=0, E:=0, G:=1$;
and	Pomnóż logicznie	$R(A) := R(A) \wedge N$
or	Dodaj logicznie	$R(A) := R(A) \vee N$
orn	Pomnóż logicznie przez negację	$R(A) := R(A) \wedge \bar{N}$
clmo	Porównaj logicznie przez maskę i przeskocz	Jeśli $(R(A) \wedge R7) \equiv (N \wedge R7)$ to $P:=1$
clbo	Porównaj logicznie iloczyn i przeskocz	Jeśli $(R(A) \wedge N) \equiv N$ to $P:=1$
lo	Umieść	$R(A) := N$
lom	Umieść przez maskę	$R(A) := (R(A) \wedge \bar{R7}) \vee (N \wedge R7)$
los	Umieść w pamięci	$S(R(A)) := N$; $R(A) := R(A) + 1$
lob	Umieść według BAR	$R(A) := S(N; BAR)$

st	Pamiętaj	$S(N) := R(A)$
stb	Pamiętaj według BAR	$S(N; BAR) := R(A)$
jpar	Skocz ze śladem w rejestrze	$R(A) := IC; IC := N$

B. Rozkazy bez pierwszego argumentu

Pola rozkazu oznaczają:

- A - przedłużenie kodu operacji
- B - numer rejestru indeksowego
- C - wskazuje argument normalny
- D - bit argumentu pośredniego

<u>Skrót</u>	<u>A</u>	<u>Nazwa</u>	<u>Treść</u>
jp	0	Skocz	$IC := N$
jpl	1	Skocz przy L	Jeśli $L = 1$ to $IC := N$
jpe	2	Skocz przy E	Jeśli $E = 1$ to $IC := N$
jpg	3	Skocz przy G	Jeśli $G = 1$ to $IC := N$
jpr	4	Skocz ze śla dem	$S(N) := IC; IC := N + 1$
jpri	5	Skocz ze śla dem przy L	Jeśli $L = 1$ to $S(N) := IC;$ $IC := N + 1$
jpri	6	Skocz ze śla dem przy E	Jeśli $E = 1$ to $S(N) := IC;$ $IC := N + 1$
jprg	7	Skocz ze śla dem przy G	Jeśli $G = 1$ to $S(N) := IC;$ $IC := N + 1$
sbar	2	Pamiętaj BAR	$S(N) [0-7] := 0; S(N) [8-15] := BAR$
mod	3	Modyfikuj na stępnym rozkaz	$MOD := N$
ex	4	Wywołanie extrakodu	$R4 := N;$ $S(0) := IC; S(1) := RO;$ $IC := S(128 + NAZWA; 0);$ $RO [8] := Q RO [0-7, 9-15] := 0$ Rozkaz powoduje przejście do podpro gramu w Systemie Operacyjnym

reex	5	Wróć	$IC := S(N; BAR \wedge (Q \vee X)) ;$ $RO [4] := S(N+1; BAR \wedge (Q \vee X)) [4] \vee RO [4]$ $RO [0-3, 5-15] := S(N+1;$ $BAR \wedge (Q \vee X)) [0-3, 5-15] ;$
ados	6	Dodaj 1 do pamięci i przeskocz przy zerze	$S(N) := S(N) + 1; \text{ ust. } V, C;$ Jeśli $S(N) = 0$ to $P := 1$
z3	7	Zeruj pamięć	$S(N) := 0$
puf	-	Wykonaj operację zmienno-przecinkową	$(R1, R2, R3) := f((R1, R2, R3), \sqrt{S(N)}, S(N+1), S(N+2))$ gdzie f - jest operacją określoną przez pole A

C. Rozkazy bez drugiego argumentu

Pola rozkazu' oznaczają:

A - numer rejestru, uniwersalnego

B, C, D - przedłużenie kodu operacji

<u>Skrót</u>	<u>B</u>	<u>C</u>	<u>D</u>	<u>Nazwa</u>	<u>Treść</u>
neg	=0	1	0	Neguj	$R(A) := \overline{R(A)}$
nec	≠0	1	0	Neguj z przeniesieniem	$R(A) := \overline{R(A)} + C; \text{ ust. } V, C.$
nega	-	1	1	Neguj arytmetycznie	$R(A) := -R(A); \text{ ust. } V, C.$
chan	-	2	0	Zamień znaki	$R(A) [0-7] := R(A) [8-15]$ i $R(A) [8-15] := R(A) [0-7]$
rkey	-	3	0	Czytaj klucze	$R(A) := INF;$ gdzie INF oznacza informację ustawioną za pomocą kluczy
ric	-	3	1	Umieść IC	$R(A) := IC$
shl	0	4	0	Przesuń w lewo	$Y := R(A) [Q]; R(A) \uparrow;$ $R(A) [15] := 0;$

shv	×0 4 0	Przesuń w lewo i badaj nadmiar	$Y := R(A) [\bar{0}]; R(A) \uparrow; R(A) [\bar{15}] := 0; \text{ust. } V$
shly	0 5 0	Przesuń w lewo z Y	$z := R(A) [\bar{0}]; R(A) \uparrow; R(A) [\bar{15}] := Y; Y := z$
shvy	×0 5 0	Przesuń w lewo z Y i badaj nadmiar	$z := R(A) [\bar{0}]; R(A) \uparrow; R(A) [\bar{15}] := Y; Y := z; \text{ust. } V$
shlx	0 6 0	Przesuń w lewo z X	$Y := R(A) [\bar{0}]; R(A) \uparrow; R(A) [\bar{15}] := X;$
shvx	×0 6 0	Przesuń w lewo z X i badaj nadmiar	$Y := R(A) [\bar{0}]; R(A) \uparrow; R(A) [\bar{15}] := X; \text{ust. } V$
shr	- 4 1	Przesuń w prawo	$Y := R(A) [\bar{15}]; R(A) \downarrow 0; R(A) [\bar{0}] := 0;$
shry	- 5 1	Przesuń w prawo z Y	$z := R(A) [\bar{15}]; R(A) \downarrow; R(A) [\bar{0}] := Y; Y := z$
shrx	- 6 1	Przesuń w prawo z X	$Y := R(A) [\bar{15}]; R(A) \downarrow 0; R(A) [\bar{0}] := X;$
stxa	- 7 0	Ustaw X według bitu 0	$X := R(A) [\bar{0}]$
stxz	- 7 1	Ustaw X według bitu 15	$X := R(A) [\bar{15}]$

Pozostałe kombinacje w polach B, C i D rozkazu są przeznaczone do późniejszego wykorzystania.

D. Rozkazy z krótkim argumentem

Pola rozkazu oznaczają:

A - numer rejestru uniwersalnego

B, C, D - krótki argument

<u>Skrót</u>	<u>Nazwa</u>	<u>Treść</u>
adt	Dodaj krótko	$R(A) := R(A) + N$; ust. V, C
adot	Dodaj krótko i przeskocz	$R(A) := R(A) + N$; ust. V, C; Jeśli $R(A) = 0$ to $P := 1$
adjt	Dodaj jedynekę i przeskocz krótko	$R(A) := R(A) + 1$; ust. V, C; Jeśli $R(A) \neq 0$ to $IC := IC + N$
cot	Porównaj krótko	Jeśli $R(A) < N$ to $L := 1, E := 0, G := 0$ Jeśli $R(A) = N$ to $L := 0, E := 1, G := 0$ Jeśli $R(A) > N$ to $L := 0, E := 0, G := 1$
lot	Umieść krótko	$R(A) := N$
lts	Umieść zawar- tość pamięci krótko	$R(A) := S(IC + N)$
sts	Pamiętaj krót- ko	$S(IC + N) := R(A)$

E. Rozkazy przeskoków krótkich

Pola rozkazu oznaczają:

A - przedłużenie kodu operacji

B, C, D - krótki argument

<u>Skrót</u>	<u>A</u>	<u>Nazwa</u>	<u>Treść</u>
jpt	0	Przeskocz krótko	$IC := IC + N$
jptl	1	Przeskocz krótko przy L	Jeśli $L = 1$ to $IC := IC + N$
jpte	2	Przeskocz krótko przy E	Jeśli $E = 1$ to $IC := IC + N$
jptg	3	Przeskocz krótko przy G	Jeśli $G = 1$ to $IC := IC + N$
jptv	4	Przeskocz krótko przy V	Jeśli $V = 1$ to $IC := IC + N; V := 0$
jptx	5	Przeskocz krótko przy X	Jeśli $X = 1$ to $IC := IC + N$
jpty	6	Przeskocz krótko przy Y	Jeśli $Y = 1$ to $IC := IC + N$
jpti	7	Przeskocz krótko przy I	Jeśli $I = 1$ to $IC := IC + N$

F. Rozkazy przesyłania grupowego

Pola rozkazu oznaczają:

A, D - przedłużenie kodu operacji

B - numer rejestru indeksowego

C - wskazuje argument normalny

<u>Skrót</u>	<u>D</u>	<u>A</u>	<u>Nazwa</u>	<u>Treść</u>
log	0		Umieść grupowo:	
		0		$R1:=S(N;BAR);R2:=S(N+1;BAR)$
		1		$R1:=S(N;BAR);R2:=S(N+1;BAR);R3:=S(N+2;BAR)$
		2		$(R1,R2,\dots,R7):=(S(N;BAR), S(N+1;BAR),\dots, S(N+6;BAR))$
		3		$R5:=S(N;BAR);R6:=S(N+1;BAR);R7:=S(N+2;BAR)$
		4		$R1:=S(N);R2:=S(N+1)$
		5		$R1:=S(N);R2:=S(N+1);R3:=S(N+2)$
		6		$(R1,R2,\dots,R7):=(S(N), S(N+1),\dots,S(N+6))$
	7		$R5:=S(N);R6:=S(N+1);R7:=S(N+2)$	
stg	1		Pamiętaj grupowo:	
		0		$S(N;BAR):=R1;S(N+1;BAR):=R2$
		1		$S(N;BAR):=R1;S(N+1;BAR):=R2;S(N+2;BAR):=R3$
		2		$(S(N;BAR),S(N+1;BAR),\dots,S(N+6;BAR)):= (R1, R2,\dots, R7)$
		3		$S(N;BAR):=R5;S(N+1;BAR):=R6;S(N+2;BAR):=R7$
		4		$S(N):=R1;S(N+1):=R2$
		5		$S(N):=R1;S(N+1):=R2;S(N+2):=R3$
		6		$(S(N),S(N+1),\dots,S(N+6)):= (R1,R2,\dots,R7)$
	7		$S(N):=R5;S(N+1):=R6;S(N+2):=R7$	

6. STANY MASZYNY

Maszyna może znajdować się w jednym z trzech stanów: STOP, START lub CZEKAJ.

STOP jest stanem, w którym żadne rozkazy nie są wykonywane, przerwania są przyjmowane, ale nie są wykonywane. Maszyna znajduje się w stanie STOP w wyniku jednej z następujących przyczyn:

- Wstępne wyzerowanie maszyny po włączeniu zasilania przełącznikiem na Pulpicie Technicznym,
- Naciśnięty został klucz STOP.

Wyjście maszyny ze stanu STOP jest możliwe tylko poprzez naciśnięcie klucza START.

START jest stanem, w którym maszyna wykonuje kolejne rozkazy z pełną szybkością. Przerwania są przyjmowane i wykonywane. Maszyna znajduje się w stanie START w wyniku jednej z przyczyn:

- Naciśnięty został klucz START,
- W stanie CZEKAJ zostało przyjęte przerwanie.

Wyjście maszyny ze stanu START następuje w wyniku naciśnięcia klucza STOP lub w wyniku wykonywania rozkazu "stop".

CZEKAJ jest stanem, w którym maszyna nie wykonuje rozkazów, ale przerwania są przyjmowane i wykonywane. Wejście maszyny w stan CZEKAJ następuje w wyniku wykonania rozkazu "stop".

Wyjście maszyny ze stanu CZEKAJ następuje w wyniku przyjęcia przerwania, co powoduje przejście w stan START, lub naciśnięcia klucza STOP, co powoduje przejście w stan STOP.

7. PULPIT TECHNICZNY MASZINY

Maszyna K-202 posiada Pulpit Techniczny, zawierający przełączniki i lampki potrzebne do jej obsługi i konserwacji.

Na Pulpicie znajdują się następujące elementy:

Klucze i przełączniki

Symbol Nazwa i funkcja

	WYBIERANIE REJESTRÓW - przełącznik obrotowy 15-pozycyjowy, pozwalający na wybranie dowolnego rejestru programowego lub roboczego Jednostki Centralnej.
	INFORMACJA - 16 przełączników dwupozycyjnych pozwalających na ustawienie dowolnej informacji 16-bitowej.
ST	START - STOP - przełącznik posiadający dwa położenia stabilne - START /dolne/ i STOP /górne/.
CYCLE	CYKL - przełącznik labilny, działający tylko w stanie STOP maszyny. Naciśnięcie powoduje wykonanie jednego cyklu rozkazowego i ponowne przejście w stan STOP.
OPRQ	ZGŁOSZENIE OPERATORA - przełącznik labilny. Naciśnięcie powoduje zgłoszenie przerwania.
BIN	WPROWADŹ BINARNIE - przełącznik labilny, działający tylko w stanie STOP maszyny. Naciśnięcie powoduje wykonanie operacji "Wstępne wprowadzanie programu" opisanej w rozdz.9, a następnie przejście w stan STOP.

- FETCH POBIERZ - przełącznik labilny, działający tylko w stanie STOP. Naciśnięcie powoduje odczytanie jednego słowa z miejsca PAO wskazanego zawartością roboczych rejestrów AR i BAR Procesora, umieszczenie tego słowa w rejestrze wybranym przełącznikiem WYBIERANIE REJESTRÓW, a następnie zwiększenie zawartości rejestru AR o 1 i przejście w stan STOP.
- STORE PAMIĘTAJ - przełącznik labilny, działający tylko w stanie STOP. Naciśnięcie powoduje zapamiętanie zawartości wybranego rejestru w miejscu PAO wskazanym zawartością rejestrów AR i BAR, a następnie zwiększenie zawartości AR o 1 i przejście w stan STOP.
- LOAD ŁADUJ - przełącznik labilny, działający tylko w stanie STOP. Naciśnięcie powoduje umieszczenie informacji ustawionej za pomocą kluczy INFORMACJA w wybranym rejestrze.
- MODE REŻIM PRACY - przełącznik - 2-pozycyjny; posiadający dwie pozycje stabilne:
N - CIĄGŁA, w której maszyna wykonuje rozkazy z pełną szybkością,
S - KROKOWA /na dół/, w której maszyna działa start - stopowo wykonując elementarny krok operacji za każdym naciśnięciem przełącznika STEP.
- STEP KROK - przełącznik labilny, działający tylko w pozycji S przełącznika MODE. Przełączenie powoduje wykonanie elementarnego kroku operacji, a następnie zawieszenie działania do chwili ponownego przełączenia.

ZASILANIE - przełącznik 4-pozycyjny, posiadający po-
 zycje - WYŁĄCZONE, WŁĄCZONE i ponadto pozycje:
 - ZERUJ - w której zasilanie jest włączone, a równo-
 cześnie wytwarzany jest sygnał powodujący ustawienie
 wskaźników i sterowania maszyny w stan początkowy.
 - BLOKUJ - w której zasilanie jest włączone, a klu-
 cze Pulpitu nie działają.

Lampki

Symbol

Nazwa i funkcja

INFORMACJA - 16 lampek na których w stanie STOP lub
 CZEKAJ maszyny wyświetlana jest zawartość wybranego
 rejestru.

RUN PRACA - lampka zapalona, gdy maszyna jest w stanie
 START.

Q - lampka wyświetlająca zawartość wskaźnika Q

I - " " " " I

V - " " " " V

IRQ PRZERW- lampka zapalona, gdy w maszynie spełniony
 jest warunek przerwania,

MC - lampka zapalona, gdy ostatnim wykonanym roz-
 kazem był rozkaz "mod",

P - lampka wyświetlająca zawartość wskaźnika P.

ON ZASILANIE - lampka zapalona, gdy zasilanie maszyny
 jest włączone.

SIMP-ZEDOK z/ 507/72 n. 500