

DOKUMENTACJA TECHNICZNO RUCHOWA

M E R A 4 0 0

2. TOM I CZ. 3

TOM I CZ. 4

TOM I CZ. 5

system MERA 400

DOKUMENTACJA TECHNICZNO-RUCHOWA

TOM I

system MERA 400

DOKUMENTACJA TECHNICZNO RUCHOWA
TOM I

część III
OPIS FUNKCJONALNY MINIKOMPUTERA
MERA - 400

Nowości w opisie są zaznaczane na marginesie. Znak przy numerze strony dotyczy numeracji stron.

Z A P I S Z M I A N

Z M I A N A	O P I S
"A" 20-09-78	Oryginał
"B" 31-01-79	Aktualizacja opisu. Rozszerzenie listy Rozkazów o operacje we/wy pamięci PK 1 i SP45DE
"C" 18-10-79	Karta zmian nr 485/79 zmiana na str.5-1

21-004033-01-1

Uwagi kierować do : Centrum Naukowo Produkcyjne Technik Komputerowych i Pomiarów
Im.J.Krasickiego , ul. Łopuszańska 117/123 02-232 Warszawa

Producent zastrzega sobie prawo wprowadzania zmian w dokumentacji.

WYKAZ CZĘŚCI TOMU I

1. Formularz techniczny
2. Lista kompletności do formularza technicznego
3. Opis funkcjonalny minikomputera MERA-400
4. Ogólna Instrukcja eksploatacji minikomputera MERA-400
5. Instrukcja pakowania modułów systemu MERA-400

S P I S T R E Ś C I

	Strona
ROZDZIAŁ I – INFORMACJA O SYSTEMIE MERA 400	1-1
ROZDZIAŁ II – CHARAKTERYSTYKA FUNKCJONALNA	2-1
ROZDZIAŁ III – LISTA ROZKAZÓW	3-1
ROZDZIAŁ IV – OPROGRAMOWANIE	4-1
ROZDZIAŁ V – ROZWIĄZANIE KONSTRUKCYJNE	5-1

1. CHARAKTERYSTYKA OGÓLNA

Minikomputer MERA 400 jest uniwersalną, 16-bitową maszyną cyfrową o architekturze logicznej odpowiadającej wymaganiom stawianym współczesnym systemom cyfrowym, zarówno pod względem struktury logicznej jak i rozwiązań technicznych. Elastyczność budowy funkcjonalnej minikomputera uzyskano dzięki zastosowaniu standardowego interfejsu łączącego procesory, kanały i pamięci operacyjne oraz przyjętej zasadzie asynchronicznej pracy tych modułów na interfejsie. Rozbudowana liczba rozkazów zapewnia efektywne działanie na danych o różnorodnej postaci.

Wysoką niezawodność pracy minikomputera gwarantuje przyjęta technika realizacyjna – elementy scalone TTL średniej skali integracji montowane na dwuwarstwowych pakietach drukowanych, solidna konstrukcja mechaniczna zapewniająca prosty montaż zespołów funkcjonalnych oraz odpowiedni dobór urządzeń peryferyjnych nie wymagających klimatyzacji.

Swobodny i szybki dostęp do wszystkich modułów funkcjonalnych, odpowiednia aparatura testująca przy rozbudowanym zestawie programów testujących i diagnostycznych gwarantują sprawny serwis i obsługę techniczną systemu.

Oprogramowanie, a w szczególności wielozadaniowy system operacyjny czasu rzeczywistego SOM-3 oraz języki wyższego rzędu FORTRAN IV, BASIC wsparte obszerną biblioteką programów, oferowane wraz z niezawodnym sprzętem dają użytkownikom wygodne i silne narzędzie do szerokiego kręgu zastosowań techniki obliczeniowej.

2. PRZEZNACZENIE

Minikomputer MERA 400 może być oferowany w zestawach problemowo-zorientowanych na zagadnienia z zakresu:

- obliczenia naukowo-techniczne
- przetwarzanie danych ekonomicznych i administracyjnych
- automatyzacji prac inżynierskich
- przetwarzanie w czasie rzeczywistym
- zdalne przetwarzanie wsadowe
- zbieranie i dystrybucja danych
- sterowanie procesami.

Modularność sprzętu i oprogramowania zapewnia łatwą rozbudowę i rekonfigurację zestawów u użytkownika.

3. PODSTAWOWE INFORMACJE O SYSTEMIE

3.1. Interfejs systemu

- główna magistrała wymiany danych łączy wszystkie moduły funkcjonalne systemu
- maksymalna konfiguracja systemu: 2 jednostki centralne, 16 bloków pamięci operacyjnej po 32 k słów w bloku, 16 kanałów wejścia/wyjścia
- asynchroniczna współpraca modułów funkcjonalnych poprzez interfejs na zasadzie "pytań-odpowiedzi"
- równoległe i jednoczesne przesyłanie rozkazów, adresów i danych
- dostęp modułów do interfejsu za zasadzie priorytetu geometrycznego.

3.2. Jednostka centralna

- długość słowa – 16 bitów
- 8 16-bitowych rejestrów uniwersalnych, w tym 7 akumulatorów lub rejestrów indeksowych (B-modyfikatorów)
- 32-bitowy rejestr zgłoszeń przerwanych pogrupowanych w 11 poziomów obsługi. Maskowanie przerwanych na 10 poziomach obsługi
- priorytet obsługi przerwanych niezależny od priorytetu dostępu do interfejsu i nie związany z geometrią systemu
- sprzętowa organizacja stosu przerwanych z automatycznym wczytaniem specyfikacji przerwania
- zegar czasu rzeczywistego

- drukarka znakowo-mozaikowa DZM-180
- drukarka wierszowa DW-3*
- drukarka znakowo-mozaikowa z klawiaturą DZM-180-KSR
- monitor ekranowy Videoton 340
- monitor ekranowy MERA-7952*

Pamięci zewnętrzne

- pamięć dyskowa MERA 9425
- pamięć taśmowa PT305*
- pamięć kasetowa PK-1
- pamięć na miękkim dysku SP45DE

3.7. Zasilanie

- sieć jednofazowa 220 V + 10% - 15%, 50 Hz
- układy automatyki zapewniające sekwencyjne włączenie i wyłączenie napięć stabilizowanych w celu ochrony pamięci operacyjnej przed zanikami napięcia
- zabezpieczenie przed wzrostem napięć i przeciążeniem
- zabezpieczenie przed nadmiernym wzrostem temperatury
- stabilizowane źródła napięć: + 5 V, + 15 V, - 5 V, - 15 V.

3.8. Warunki eksploatacji (moduł podstawowy bez urządzeń zewnętrznych)

- temperatura pracy + 5°C ÷ + 35°C
- względna wilgotność powietrza 95% przy temperaturze 30°C
- wytrzymałość na wibracje: amplituda 0,35 m w zakresie 10 ÷ 55 Hz
- wytrzymałość termiczna - 40°C ÷ + 55°C.

4. OPROGRAMOWANIE

- wielozadaniowy system operacyjny czasu rzeczywistego SOM-3 może być dostarczany w różnych wersjach zależnych od konfiguracji zestawu MERA-400 i zastosowań;
- zestaw procesorów systemowych:
 - Job Control steruje przetwarzaniem wsadowym
 - Konsolidator (Link Editor) łączy w jeden program moduły wynikowe, umożliwia tworzenia nakładek (overlay ów)
 - Bibliotekarz (Cataloger) umożliwia tworzenie zbiorów słownikowych modułów wynikowych
 - Aktualizator Biblioteki Sekwencyjnej (Library Update)
 - Aktualizator Tekstowy (Source Update) pozwala przetwarzać zbiory źródłowe
 - Debugger ułatwia uruchomienie programów
- SOM-3 umożliwia korzystanie z następujących translatorów:
 - Macroassembler
 - Basic wielodostępny
 - Fortran IV rozszerzony, umożliwiający wykorzystanie pełnych możliwości systemu operacyjnego SOM-3
 - Symbol* - język do zastosowań z zakresu przetwarzania danych
- biblioteki programów inżynierskich w języku Basic i Fortran.

* w opracowaniu

1. ARCHITEKTURA SYSTEMU MERA 400

System MERA 400 składa się z modułów funkcjonalnych, komunikujących się wyłącznie za pomocą standardowego interfejsu. Modułowa struktura systemu umożliwia jego łatwą rozbudowę poprzez bezpośrednie dołączenie do interfejsu dodatkowych modułów.

W systemie MERA 400 mogą pracować niżej wymienione moduły funkcjonalne:

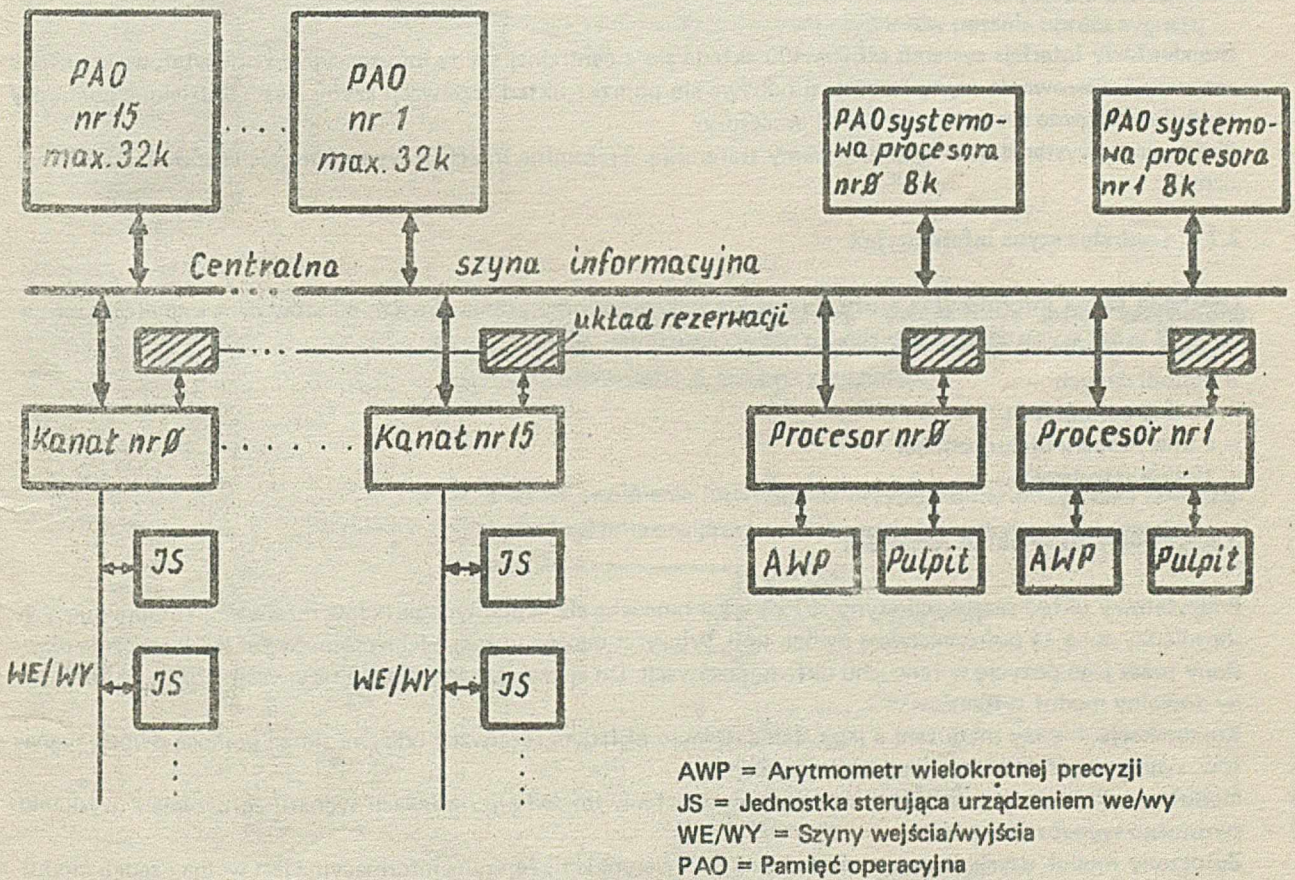
- procesory
- pamięci operacyjne
- kanały

Maksymalna konfiguracja systemu składa się z:

- 2 procesorów
- 17 modułów pamięci operacyjnej
- 16 kanałów

przy czym może to być dowolna kombinacja kanałów:

- znakowych
- pamięciowych (bezpośredniego dostępu)
- automatyki (PI, CAMAC)
- autonomicznych jednostek sterujących.



Architektura systemu MERA 400 w konfiguracji maksymalnej

Pojemność jednego modułu pamięci operacyjnej wynosi max. 32 k słów 16-bitowych.

Sterowanie wszystkich modułów systemu MERA 400 jest całkowicie asynchroniczne. Stąd, poszczególne moduły systemu mogą być wykonywane przy użyciu elementów elektronicznych o dowolnych parametrach dynamicznych. Oprócz łatwości technicznej realizacji oznacza to również pełne otwarcie systemu na nowe, przyszłościowe technologie np. szybkie pamięci półprzewodnikowe.

Interfejs systemu MERA 400 składa się z centralnej szyny informacyjnej i priorytetowego układu rezerwacji szyny.

Centralna szyna informacyjna służy do przesyłania danych, adresów oraz sygnałów sterujących i kontrolnych. Sterowanie dostępem do szyny centralnej jest rozdzielone pomiędzy procesory oraz kanały i realizowane jest przez układ rezerwacji. W danej chwili do szyny centralnej może być dołączony aktywnie tylko jeden moduł nadawczy tj. procesor lub kanał. Ponadto moduł, który aktualnie korzysta z szyny nigdy nie otrzyma zezwolenia na natychmiastową ponowną transmisję, o ile w trakcie pierwszej transmisji inny moduł (lub moduły) zgłosił gotowość zajęcia szyny. Rozwiązanie takie umożliwia współbieżną pracę modułów, a tym samym zwiększa wydajność systemu.

Priorytet dostępu modułu do interfejsu wynika z funkcji spełnianych przez moduł w systemie. Priorytet ten jest jednoznacznie określony przez pozycję modułu w łańcuchu układów rezerwacji i może być łatwo zmieniony nawet po zainstalowaniu systemu u użytkownika np. w wypadku rozbudowy systemu. Priorytet dostępu modułu do interfejsu jest całkowicie niezależny od priorytetu obsługi przerw zgłaszanych przez dany moduł.

2. INTERFEJS

2.1. Przeznaczenie

Interfejs jest zespołem reguł i środków technicznych umożliwiających komunikację między modułami systemu.

2.2. Opis funkcjonalny

Standardowy interfejs systemu MERA 400 składa się z centralnej szyny informacyjnej i priorytetowego układu rezerwacji. Sterowanie szyny centralnej odbywa się poprzez układ rezerwacji szyny i jest rozdzielone pomiędzy moduły nadawcze systemu tj. kanały i procesory.

Każde wykorzystanie interfejsu nazywamy transmisją. Transmisje interfejsowe przebiegają w sposób asynchroniczny.

2.2.2. Centralna szyna informacyjna

Centralna szyna informacyjna służy do dwukierunkowego przesyłania danych, adresów oraz sygnałów sterujących. W skład szyny centralnej wchodzi następujące linie:

- 16 linii danych
- 16 linii adresowych
- 4 linie numeru bloku pamięci
- 12 linii sterujących.

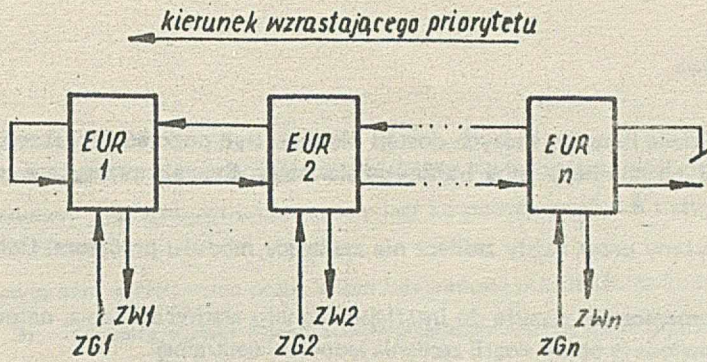
2.2.3. Priorytetowy układ rezerwacji

Priorytetowy układ rezerwacji szyny składa się z łańcucha elementarnych układów rezerwacji, komunikujących się między sobą za pośrednictwem dwóch linii. Priorytet danego układu elementarnego jest jednoznacznie określony przez jego pozycję w łańcuchu układu rezerwacji. Do elementarnego układu rezerwacji może być dołączony dowolny moduł systemu.

Komunikacja między modułem a jego elementarnym układem rezerwacji odbywa się za pomocą dwóch sygnałów: sygnału zgłoszenia oraz sygnału zezwolenia.

Moduł uzyskuje dostęp do szyny centralnej po wysłaniu do układu rezerwacji sygnału zgłoszenia i uzyskaniu zwrotnego sygnału zezwolenia.

Zgłoszony moduł uzyskuje sygnał zezwolenia pod warunkiem, że szyna informacyjna jest wolna i żaden moduł o wyższym priorytecie nie żąda dostępu do niej. Następny dostęp do szyny centralnej uzyska moduł o najwyższym priorytecie, spośród oczekujących na dostęp, nawet w przypadku, gdy moduł aktualnie prowadzący transmisję natychmiast po jej zakończeniu zgłasza żądanie ponownego zajęcia szyny. Celem takiego rozwiązania jest umożliwienie współbieżnej pracy modułów systemu.



EUR = elementarny układ rezerwacji
 ZG = sygnał zgłoszenia
 ZW = sygnał zezwolenia

Priorytetowy układ rezerwacji szyny

2.3. Rodzaje transmisji

2.3.1. Transmisje normalne

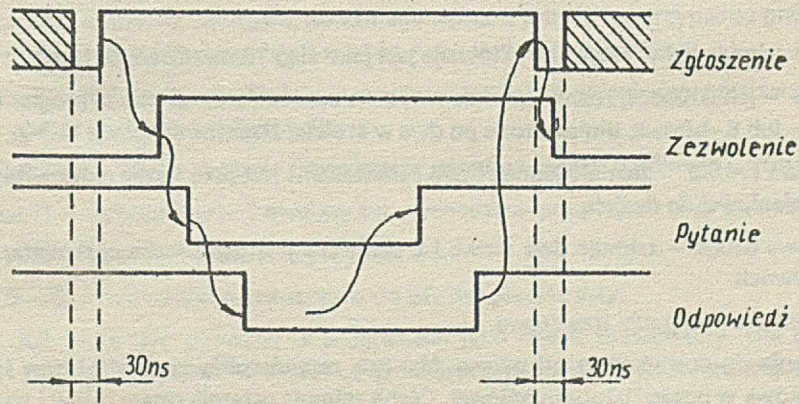
Każda transmisja normalna musi być poprzedzona rezerwacją szyny centralnej. Po dokonaniu rezerwacji moduł nadawczy wysyła "pytanie". Jeżeli pytanie jest prawidłowe adresat przesyła "odpowiedź". Wysyłanie odpowiedzi może nastąpić dopiero po wykorzystaniu informacji zawartej w pytaniu. Od momentu wysłania odpowiedzi aż do jej zakończenia nie wolno korzystać z żadnych sygnałów pytania oprócz sygnału wiodącego pytanie tj. sygnału:

- W – zapis do pamięci
- R – odczyt z pamięci
- S – przestanie
- F – pobranie
- IN – przerwanie

Zanik sygnału wiodącego powinien powodować zakończenie odpowiedzi. Pytanie może zostać zakończone dopiero po całkowitym wykorzystaniu odpowiedzi. Po zakończeniu pytania nie należy korzystać z żadnych sygnałów odpowiedzi oprócz sygnału wiodącego odpowiedzi tj. jednego z sygnałów:

- OK – odpowiedź pozytywna
- EN – odpowiedź negatywna

Zanik sygnału wiodącego odpowiedzi zezwala na zwolnienie interfejsu. W przypadku nie otrzymania odpowiedzi w czasie $2,5 \div 5 \mu s$ transmisja powinna zostać zakończona przez moduł nadawczy w trybie awaryjnym.



Przebieg transmisji normalnej

2.3.2. Transmisje alarmowe

Szyna centralna zawiera dwie linie, do których dostęp nie musi być poprzedzony rezerwacją szyny. Każde wykorzystanie dowolnej z tych linii nazywamy transmisją alarmową. Sygnały przesyłane w trakcie transmisji alarmowych, to:

- alarm zasilania — wysyłany przez każdy zasilacz nie zasilający modułu procesora. Odbiorcą sygnału jest procesor.
Zasilacz, który zasila procesor nie wysyła do interfejsu sygnału alarmu zasilania, natomiast wysyła do procesora inne sygnały, omówione w opisie awarii zasilania jednostki centralnej.
- zerowanie ogólne — wysyłane przez procesor lub pulpit techniczny; odbiorcami sygnału są wszystkie moduły systemu.

3. JEDNOSTKA CENTRALNA

3.1. Przeznaczenie

Jednostka centralna, jako element systemu MERA 400 przeznaczona jest do wykonywania wszystkich instrukcji podanych w liście rozkazów oraz operacji z pulpitu technicznego.

3.2. Opis funkcjonalny

Jednostka centralna spełnia następujące funkcje:

- pobiera rozkazy z pamięci operacyjnej
- interpretuje i wykonuje rozkazy przy użyciu odpowiednich elementów funkcjonalnych takich, jak arytmometr, rejestry itp.
- inicjuje operacje we/wy
- rejestruje i obsługuje przerwania
- wykonuje operacje z pulpitu operatora.

3.2.1. Postać przetwarzanych informacji

Słowo maszynowe ma długość 16 bitów numerowanych od 0 do 15 (od najbardziej do najmniej znaczącej pozycji).

W jednostce centralnej przetwarza się dane, adresy i rozkazy.

3.2.1.1. Dane:

- a/ informacja logiczna — zajmuje jedno słowo i traktowana jest jako ciąg 16 niezależnych bitów.
- b/ znaki alfanumeryczne — ich długość, rozmieszczenie w słowie oraz kody określone są programem; przeważnie stosowane są znaki 7- lub 8-bitowe, umieszczone po dwa w słowie, zapisane w kodzie ISO-7.
- c/ liczba stałoprzecinkowa krótka — zajmuje jedno słowo i traktowana jest jako liczba całkowita; liczba zapisana jest w notacji uzupełnieniowej do dwóch.
- d/ liczba stałoprzecinkowa długa — zajmuje dwa słowa i traktowana jest jako liczba całkowita; zapis w notacji uzupełnieniowej do dwóch.
- e/ liczba zmiennoprzecinkowa — zajmuje trzy słowa.

Jako standard przyjmuje się postać znormalizowaną. Mantysa zajmuje pierwszych 40 bitów i traktowana jest jako liczba zaprzecinkowa w notacji uzupełnieniowej. Cecha zajmuje ostatnie osiem bitów i traktowana jest jako liczba całkowita również w notacji uzupełnieniowej do dwóch.

3.2.1.2. Rozkazy:

a/ rozkazy w podstawowej postaci.

Zajmują 1 słowo maszynowe zwane słowem podstawowym, które zawiera następujące informacje:

- kod operacji – podstawowy kod operacji umieszczony jest na pozycjach $\emptyset \div 5$ i służy do rozróżnienia rozkazów lub grup rozkazowych
- bit D (pozycja 6) – zawiera bit adresowania pośredniego lub stanowi przedłużenie kodu operacji.
- pole A (pozycja 7 \div 9) – zawiera numer rejestru uniwersalnego lub stanowi przedłużenie kodu operacji.
- pole B (pozycja 10 \div 12) – zawiera numer rejestru uniwersalnego służącego do B-modyfikacji argumentu lub stanowi przedłużenie kodu operacji.
- pole C (pozycje 13 \div 15) – zawiera numer rejestru uniwersalnego lub stanowi przedłużenie kodu operacji.

b/ rozkazy z argumentem normalnym bezpośrednim.

Zajmują dwa słowa maszynowe – argument zajmuje następne słowo za słowem podstawowym rozkazu; w tym przypadku znaczenie poszczególnych pozycji jest takie samo, jak w pkt. a/ z tym, że pole C= \emptyset

c/ rozkazy z argumentem krótkim bezpośrednim.

Argument 7-bitowy jest podany na pozycjach D, B, C; bit D wskazuje znak argumentu krótkiego, pola B i C – wartość; znaczenie pozostałych pozycji jak w p.a/.

d/ rozkazy z argumentem bajtowym bezpośrednim.

8-bitowy argument bezpośredni podany jest na pozycjach 8 \div 15, pozycje $\emptyset \div 5$ określają kod operacji, pozycje 6 i 7 stanowią przedłużenie kodu operacji.

3.2.1.3. Adresy:

a/ adres słowa – 15-bitowa, całkowita liczba binarna z przedziału $\emptyset \div 32767$.

b/ adres bajtu – 16-bitowa, całkowita liczba binarna z przedziału $\emptyset \div 65535$; adres parzysty oznacza bajt starszy (pozycje $\emptyset \div 7$), nieparzysty oznacza bajt młodszy (pozycje 8 \div 15).


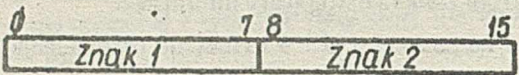
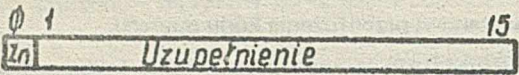
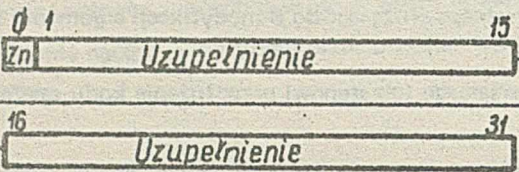
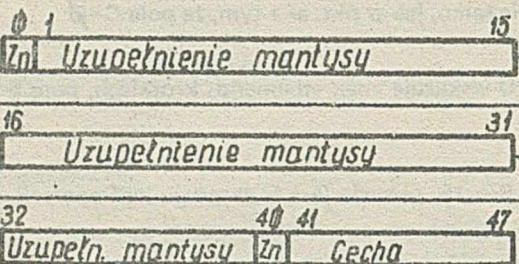
3.3. Rejestry i wskaźniki jednostki centralnej

Symbol Nazwa i funkcja

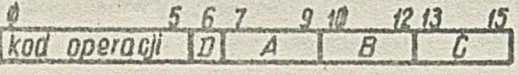
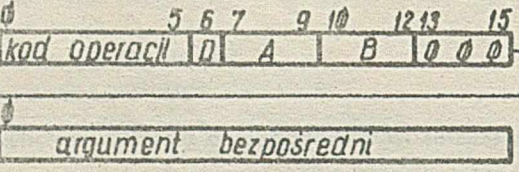
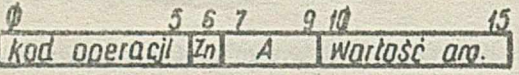
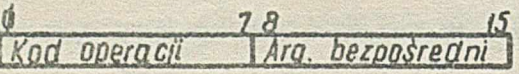
- $R\emptyset$ – Rejestr stanu programu 16-bitowy. Stanowi zbiór wskaźników decydujących o sposobie wykonywania programu.
Pozycje rejestru $R\emptyset$ oznaczają:
- Z (poz. \emptyset) – wskaźnik zera, ustawiany w przypadku otrzymania zera w wyniku działań arytmetycznych i logicznych,
 - M (poz.1) – wskaźnik znaku minus, ustawiany w przypadku otrzymania liczby ujemnej w wyniku działań arytmetycznych,
 - V (poz.2) – wskaźnik nadmiaru, do którego wpisane jest "1" w przypadku przekroczenia zakresu liczb przy operacjach arytmetycznych i w niektórych innych przypadkach,
 - C (poz.3) – wskaźnik przeniesienia ustawiany zgodnie z przeniesieniem z zerowej pozycji arytmetru przy operacjach arytmetycznych,
 - L,E,G (poz.4 – 6) – wskaźniki ustawione w wyniku operacji porównań arytmetycznych i logicznych,
 - Y (poz.7) – wskaźnik przechowujący bit wychodzący poza rejestr przy operacjach przesuwania,
 - X (poz.8) – wskaźnik ustawiany programowo,
 - (poz.9–15) – wskaźniki przeznaczone do użytku programisty.

Rejestr $R\emptyset$ może być używany w programach jako rejestr uniwersalny. Dla programów użytkowych (Q = 1) zapis do pozycji $\emptyset \div 7$ rejestru $R\emptyset$ nie jest wykonywany (z wyjątkiem rozkazu LPC).

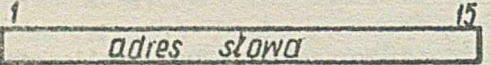
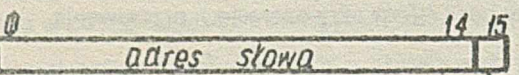
Postać danych

- a.  Informacja logiczna
- b.  Znaki alfanumeryczne
- c.  Liczba stałoprzecinkowa krótka
- d.  Liczba stałoprzecinkowa długa
(dwa słowa)
- e.  Liczba zmiennoprzecinkowa
(trzy słowa)

Postać rozkazów

- a.  Podstawowa postać rozkazu
- b.  Rozkaz z argumentem
normalnym bezpośrednim (dwa słowa)
- c.  Rozkaz z argumentem
krótkim bezpośrednim
- d.  Rozkaz z argumentem
bajtowym bezpośrednim

Postać adresów

- a.  Adres słowa
- b.  Adres bajtu
- { 0 - wskazuje bajt lewy
1 - wskazuje bajt prawy

- R1 ÷ R7 — Rejestry uniwersalne — 16-bitowe; podstawowe rejestry używane do przechowywania informacji, wykonywania operacji oraz używane jako rejestry indeksowe służące B-modyfikacji argumentów.
- IC — Licznik rozkazów 16-bitowy, zawierający adres miejsca pamięci operacyjnej, z którego pobrany zostanie rozkaz.
- SR — Rejestr stanu.
Pozycje SR oznaczają:
RM (poz. 0 ÷ 9) — rejestr masek przerw
Q (poz.10) — wskaźnik systemu; stan Q = 1 odpowiada pracy programu użytkowego;
Stan Q = 0 pracy systemu operacyjnego,
BS (poz.11) — wskaźnik specjalny,
NB (poz.12–15) — rejestr numeru bloku pamięci operacyjnej.
- RZ — 32-pozycyjny rejestr zgłoszeń przerw.
- AR — Rejestr adresowy pamięci.
- IR — Rejestr rozkazów; zawiera aktualnie wykonywaną instrukcję.
- AC — Rejestr akumulatora.
- P — Wskaźnik przeskoku.

3.4. Zasady wykonywania rozkazów

Pełny cykl czynności minikomputera związanych z wykonywaniem każdego rozkazu jest następujący:

- pobranie rozkazu z miejsca pamięci operacyjnej, wskazanego przez aktualną zawartość licznika rozkazów IC,
- określenie efektywności rozkazu,
- wyznaczenie efektywnego argumentu,
- wykonanie czynności 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 długości rozkazu (liczoną w słowach).

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

- zatrzymanie minikomputera w stanie STOP w wyniku wyłączenia klucza START
- wykonanie przerwania
- przejście w stan CZEKAJ (wykonanie rozkazu HLT)
- wykonanie następnego cyklu rozkazowego

3.4.1. Pobranie rozkazu

Rozkaz pobierany jest z bloku systemowego pamięci operacyjnej, gdy wskaźnik Q = 0 lub z bloku o numerze wskazanym zawartością rejestru NB, gdy Q = 1.

Adres pierwszego słowa rozkazu wskazany jest zawartością licznika rozkazów. Liczba pobranych słów zależy od długości rozkazu.

3.4.2. Określenie efektywności rozkazu

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

- błędny kod operacji,
- po raz czwarty użyty rozkaz MD (modyfikuj),
- rozkaz nielegalny użyty w programie użytkowym.

Dla tych przypadków następuje:

- wyzerowanie modyfikatora (MOD),
- wpisanie "1" do pozycji 6 rejestru zgłoszeń przerwań RZ
- natychmiastowe zakończenie rozkazu,

Po wykryciu nieefektywnego rozkazu żadne inne warunki nie są badane. Zawartość rejestrów programowych i pamięci nie ulega zmianie. Licznik rozkazów zawiera adres pierwszego słowa tego rozkazu zwiększony o 1.

3.4.3. Wyznaczenie argumentu efektywnego.

Argument efektywny (oznaczony symbolem N) 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 kilkoma sposobami, zależnie od postaci argumentu pierwotnego rozkazu i rodzaju użytej modyfikacji argumentu.

Argument pierwotny rozkazu może należeć do jednego z następujących rodzajów:

a/ argument normalny jest liczbą 16-bitową znajdującą się w miejscu wskazanym przez pole C rozkazu

- jeżeli $C = \emptyset$ argument stanowi zawartość następnego słowa za słowem podstawowym rozkazu;
- jeżeli $C \neq \emptyset$ argument stanowi zawartość rejestru uniwersalnego o numerze wskazanym polem C;

b/ 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;

c/ argument bajtowy bezpośredni zajmuje pozycje 8 ÷ 15 słowa podstawowego rozkazu.

Argument bajtowy jest przekształcany do postaci pełnej 16-bitowej liczby.

Modyfikacja argumentu

Możliwe są trzy 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, zawierającym pierwotny argument i ma miejsce wówczas, gdy zostanie on poprzedzony rozkazem MD (modyfikuj). Argument efektywny rozkazu MD nazywamy modyfikatorem (MOD). Pre-modyfikacja polega na dodaniu modyfikatora do argumentu pierwotnego rozkazu.

Dodawanie jest wykonywane z zaniedbaniem ewentualnego nadmiaru. Rozkaz MD może być również poprzedzony rozkazem MD, ale czwarty kolejny rozkaz MD jest traktowany jako nieprawidłowy i nie jest wykonywany, natomiast powoduje zgłoszenie przerwania.

b/ B-modyfikacja jest możliwa w rozkazach z argumentem normalnym, 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.

c/ D-modyfikacja jest możliwa w rozkazach z normalnym argumentem. Polega ona na tym, że uzyskany w wyniku poprzednich modyfikacji argument traktowany jest jako adres argumentu efektywnego. Zerowa zawartość pola D oznacza brak D-modyfikacji. Jeśli $D = 1$, to w celu otrzymania argumentu efektywnego, dokonywany jest odczyt z bloku zerowego pamięci operacyjnej (przy $Q = \emptyset$) lub z boku o numerze wskazanym zawartością pozycji rejestru NB (przy $Q = 1$). Jeśli $D = \emptyset$, argumentem efektywnym jest argument wyznaczony w wyniku poprzednich modyfikacji.

Argument efektywny rozkazu obliczany jest wg wzoru:

$$N = R/C/ + M + MOD + R/B/ \quad \text{gdy } D = \emptyset$$

$$N = S(R/C/ + M + MOD + R/B/) \quad \text{gdy } D = 1$$

gdzie:

R/C/ – zawartość rejestru uniwersalnego o numerze wskazanym w polu C rozkazu. Występuje w przypadku $C \neq \emptyset$

M – zawartość następnego słowa za słowem podstawowym rozkazu. Występuje, gdy $C = \emptyset$

MOD – modyfikator, argument efektywny rozkazu MD. Występuje, gdy poprzednio użytym rozkazem był MD.

R/B/ – zawartość rejestru uniwersalnego o numerze wskazanym w polu B rozkazu. Występuje, gdy $B \neq \emptyset$.

S// – zawartość miejsca pamięci o adresie podanym w nawiasach.

Dla rozkazów z normalnym argumentem możliwe są wszystkie przypadki modyfikacji.

Dla rozkazów z krótkim argumentem możliwa jest tylko pre-modyfikacja.

Dla rozkazów, których pole C jest przedłużeniem kodu operacji czynności wyznaczania argumentu efektywnego nie są wykonywane.

Po wyznaczeniu argumentu efektywnego następuje przejście do wykonania operacji określonej kodem operacji rozkazu zgodnie z opisem zawartym w liście rozkazów.

3.5. Przerwania

Jednostka centralna posiada układ przerwań, który jest w stanie przyjmować i obsługiwać 32 przerwania. Układ przerwań jest układem priorytetowym. Ustalanie priorytetów związane jest z numerem zgłoszonego przerwania, a mianowicie im niższy jest numer przerwania, tym wyższy jest jego priorytet.

Układ przerwań zawiera następujące rejestry:

- RZ – 32-bitowy rejestr zgłoszeń przerwań, w którym zapamiętywane są wszystkie zgłoszenia przerwań
- RM – 10-bitowy rejestr służący do maskowania przerwań

Znaczenie poszczególnych pozycji rejestru zgłoszeń przerwań RZ:

- | | |
|---------|-------------------------------------------------------------------------------|
| 0 | alarm zasilania (z danego procesora) |
| 1 | błąd parzystości pamięci operacyjnej |
| 2 | brak pamięci operacyjnej |
| 3 | zgłoszenie przerwania z drugiego procesora (przerwanie o wyższym priorytecie) |
| 4 | zanik zasilania (z interface u) |
| 5 | przerwania zegarowe |
| 6 | nieprawidłowy rozkaz |
| 7 | nadmiar dzielenia stałoprzecinkowego |
| 8 | podmiar zmiennoprzecinkowy |
| 9 | nadmiar zmiennoprzecinkowy |
| 10 | błąd danych zmiennoprzecinkowych lub próba dzielenia przez zero |
| 11 | do zastosowań specjalnych |
| 12 ÷ 27 | przerwania kanałowe |
| 28 | zgłoszenie operatora |
| 29 | zgłoszenie przerwania z drugiego procesora (przerwanie o niższym priorytecie) |
| 30 ÷ 31 | przerwania programowe |

Wszystkie pozycje rejestru zgłoszeń z wyjątkiem pozycji przerwań kanałowych są dostępne programowo (rozkazami KI i FI).

Wszystkie pozycje (z wyjątkiem zerowej) rejestru RZ są maskowane. Maskę stanowi rejestr maskujący RM o długości 10 bitów. Przerwania, które zostały zgłoszone, mogą być obsługiwane tylko w przypadku, gdy zapalona jest odpowiadająca im pozycja w rejestrze maskującym.

Przyporządkowanie pozycji rejestru RZ pozycjom rejestru maskującego ilustruje poniższa tabela:

POZYCJA RZ	POZYCJA RM
\emptyset	niemaskowana
1	\emptyset
2	1
3	2
4	3
5 ÷ 11	4
12 ÷ 13	5
14 ÷ 15	6
16 ÷ 21	7
22 ÷ 27	8
28 ÷ 31	9

Obsługa przerwania wykonuje się gdy:

- nastąpi zgłoszenie przerwania do rejestru RZ
- zapalona jest maska, odpowiadająca zgłoszonemu przerwaniu
- wskaźnik przeskoku $P = \emptyset$
- ostatnim wykonywanym rozkazem nie był rozkaz modyfikacji MD
- jednostka centralna nie jest w stanie STOP.

Obsługa przerwania (sprzętowa) składa się z następujących czynności:

- zapamiętanie aktualnej zawartości licznika rozkazów IC
- zapamiętanie aktualnej zawartości rejestru $R\emptyset$
- zapamiętanie aktualnej zawartości rejestru stanu SR
- zapamiętanie specyfikacji przerwania w przypadku przerwania zewnętrznego
- wyzerowanie rejestru $R\emptyset$
- wyzerowanie pozycji rejestru RM odpowiadającej danemu przerwaniu oraz wszystkich mniej znaczących pozycji rejestru RM
- wyzerowanie $SR_1\emptyset$ (wskaźnik Q)
- wpisanie nowej zawartości licznika rozkazów IC odpowiadającej numerowi przerwania
- uaktualnienie wskaźnika stosu.

Zapamiętywanie zawartości IC, $R\emptyset$, SR oraz specyfikacji przerwania odbywa się począwszy od adresu wskazanego przez wskaźnik stosu. Wskaźnik stosu znajduje się w komórce pamięci o adresie 97. Uaktualnienie wskaźnika stosu polega na zwiększeniu jego wartości o 4.

Wpisanie nowej zawartości IC odbywa się wg tabeli przerwań. Tabela przerwań zawiera adresy początków podprogramów obsługi przerwań. Tabela umieszczona jest w komórkach pamięci o adresach 64 ÷ 95 w bloku zerowym (systemowym), przy czym kolejne adresy odpowiadają numerem przerwań. Wpisanie nowej zawartości IC polega na wpisaniu zawartości komórki pamięci odpowiadającej numerowi przyjętego przerwania + 64.

Zgłoszenie przerwania typu nieprawidłowy rozkaz może być spowodowane jedną z trzech przyczyn:

- błędny kod operacji
- użycie rozkazu MD-modyfikuj po raz czwarty
- użycie rozkazu nielegalnego w programie użytkowym.

3.6. Zegar

Jednostka centralna wyposażona jest standardowo w zegar czasu rzeczywistego, który składa się z generatora kwarcowego 1 MHz i dzielnika częstotliwości. Dostępne są następujące częstotliwości zgłaszania przerwań zegarowych:

2, 10, 20, 40, 80 ms (standard – 10 ms). Przerwania zegarowe są podawane na pozycję 5 rejestru zgłoszeń RZ.

3.7. Alarm zasilania i restart systemu

Minikomputer posiada układ zabezpieczający przed zanikami sieci zasilania. W przypadku zaniku napięcia sieciowego zasilacz generuje sygnał przerwania ALARM ZASILANIA (przerwanie o najwyższym priorytecie). W wyniku obsługi tego przerwania następuje zapamiętanie stanu systemu, po czym zanikają napięcia w układach. Ponowne pojawienie się napięcia sieciowego powoduje wysłanie sygnału zerującego, a po ustaleniu się wszystkich napięć zasilających, wygenerowanie sygnału RESTARTU. Gdy klucz START/STOP na pulpicie operatora znajduje się w stanie START, system rozpoczyna wykonywanie programu począwszy od adresu \emptyset w bloku systemowym.

3.8. Pulpit techniczny

Na płycie czołowej minikomputera znajduje się pulpit techniczny, na którym umieszczone są przełączniki i lampki przeznaczone do obsługi maszyny jak również czynności serwisowych.

Przełącznik wybierania rejestrów – 15-pozycyjny, obrotowy przełącznik, służący do wybrania dowolnego rejestru programowego lub roboczego jednostki centralnej. Poszczególne pozycje tego przełącznika oznaczają:

R \emptyset -R7 – rejestry uniwersalne

IC – licznik rozkazów

AC – akumulator

AR – rejestr adresowy pamięci

IR – rejestr rozkazów

SR – rejestr stanu

RZ – rejestr zgłoszeń przerwań (bez pozycji odpowiadających przerwaniom zewnętrznym)

KB – wybranie zawartości kluczy informacyjnych na pulpicie

Klucze:

KB – 16 kluczy stabilnych służących do ustawienia dowolnej informacji 16-bitowej.

START – start-stop – klucz stabilny, służący do uruchamiania i zatrzymywania maszyny.

CYCLE – cykl – klucz niestabilny, którego naciśnięcie powoduje wykonanie jednego cyklu rozkazowego. Klucz działa tylko w stanie STOP maszyny. Po wykonaniu operacji następuje przejście w stan STOP.

OPRQ – zgłoszenie operatora – klucz niestabilny, którego naciśnięcie powoduje zgłoszenie przerwania.

BIN – wprowadź binarnie – klucz niestabilny, którego naciśnięcie powoduje wykonanie operacji wstępnego wprowadzenia programu; klucz działa tylko w stanie STOP maszyny. Po wykonaniu operacji następuje przejście w stan STOP.

STORE – pamiętaj – przełącznik niestabilny, działający tylko w stanie STOP; naciśnięcie powoduje zapamiętanie zawartości wybranego rejestru w komórce pamięci wskazanej zawartością rejestrów AR i rejestru numeru bloku NB, a następnie zwiększenie zawartości AR o 1 i przejście w stan STOP.

FETCH – pobierz – przełącznik niestabilny, działający w stanie STOP; naciśnięcie powoduje odczytanie jednego słowa z komórki pamięci o adresie wskazanym przez rejestr AR i rejestr numeru bloku NB, umieszczenie tego słowa w rejestrze, zwiększenie zawartości AR o 1 i przejście w stan STOP.

LOAD – umieść – przełącznik niestabilny, którego naciśnięcie powoduje umieszczenie w wybranym rejestrze informacji ustawionej przy pomocy kluczy informacyjnych KB; przełącznik działa tylko w stanie STOP.

MODE – reżim pracy – przełącznik stabilny, ustalający jeden z następujących reżimów pracy:

– praca ciągła, w której maszyna wykonuje rozkazy z pełną szybkością (lampa MODE zgaszona).

– praca krokowa, w której maszyna działa start–stopowo wykonując elementarny krok operacji przy każdym naciśnięciu klucza STEP (lampa MODE zapalona).

- STEP – krok – przełącznik niestabilny działający tylko w reżimie pracy krokowej; naciśnięcie powoduje wykonanie elementarnego kroku operacji, a następnie zawieszenie działania do chwili ponownego naciśnięcia klucza.
- STOP N – stop na adresie – przełącznik niestabilny, służy do zatrzymania maszyny przy odwołaniu do komórki pamięci o adresie ustawionym na kluczach informacyjnych KB na pozycjach 1÷15, pozycja zerowa wybiera przy ustawieniu w stan 1 – blok systemowy, przy ustawieniu w stan 0 – blok użytkowy. Stan przełącznika STOP N wyświetlany jest na lampce nad przełącznikiem.
- CLOCK – zegar – przełącznik stabilny, którego włączenie powoduje uruchomienie zegara czasu rzeczywistego; przy włączeniu zegara zapala się lampka CLOCK.
- CLEAR – zerowanie – przełącznik niestabilny, naciśnięcie powoduje ustawienie systemu w stan początkowy.

Elementy sygnalizacyjne:

Lampki informacyjne – 16 lampek, na których wyświetlana jest zawartość wybranego rejestru w stanie STOP lub CZEKAJ

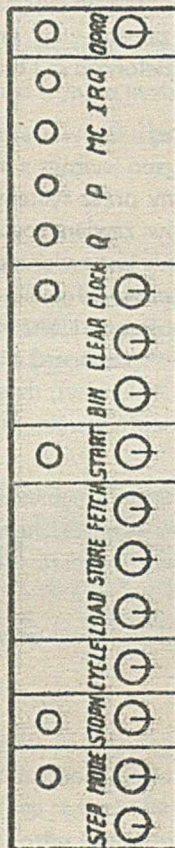
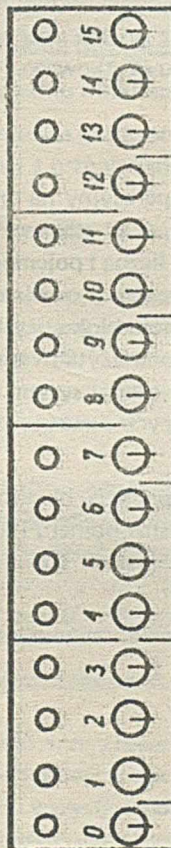
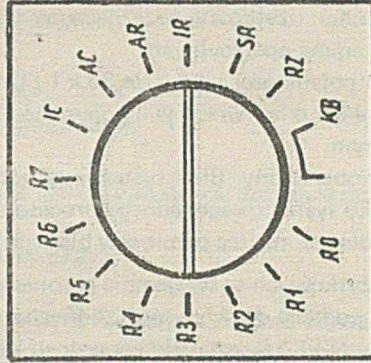
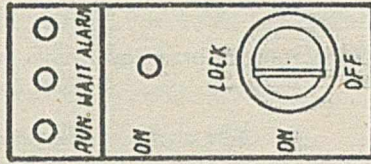
- RUN – praca – lampka zapalona, gdy maszyna jest w stanie START
- WAIT – czekaj – lampka zapalona, gdy maszyna jest w stanie CZEKAJ
- Q – lampka wyświetlająca zawartość wskaźnika Q w rejestrze stanu systemu
- IRQ – lampka zapalona, gdy przyjęte zostało przerwanie
- MC – lampka zapalona, gdy ostatnim wykonanym rozkazem był rozkaz modyfikuj MD
- P – lampka wyświetlająca zawartość wskaźnika przeskoku P
- ALARM – lampka zapalona, gdy wystąpi błąd (parzystość lub brak pamięci) w bloku systemowym pamięci operacyjnej
- MODE – lampka zapalona, gdy maszyna jest w reżimie pracy krokowej
- STOP N – lampka zapalona, gdy w maszynie uruchomiony został mechanizm zatrzymania się na adresie wskazanym na kluczach informacyjnych
- CLOCK – lampka sygnalizująca włączenie zegara czasu rzeczywistego
- ON – lampka zapalona, gdy zasilanie maszyny jest włączone.

Stacyjka

Stacyjka jest 3-pozycyjnym przełącznikiem.

Znaczenie poszczególnych pozycji jest następujące:

- OFF – zasilanie wyłączone
- ON – zasilanie włączone
- LOCK – zasilanie włączone i blokada kluczy (z wyjątkiem QPRO).



4. PAMIĘĆ OPERACYJNA

4.1. Przeznaczenie

Pamięć operacyjna minikomputera jest przeznaczona do przechowywania programów i danych, wymagających szybkiego i swobodnego dostępu.

4.2. Organizacja pamięci.

Wyróżnia się dwa podstawowe obszary pamięci: użytkowy, przeznaczony do działania programów użytkowych oraz systemowy, przewidziany do pracy systemów operacyjnych.

Obszar użytkowy pamięci stanowią bloki o pojemności od 4 k do 32 k (z przyrostem co 4 k) o numerach 1÷15. Obszar systemowy złożony jest z dwóch bloków o numerach ϕ o pojemności 4 k lub 8 k każdy. Każdy z procesorów dysponuje jednym blokiem systemowym.

W bloku systemowym zapisany jest system operacyjny. Blok systemowy jest niedostępny dla programu użytkowego, natomiast możliwy jest dostęp z bloku systemowego jednego procesora, do bloku systemu drugiego procesora. System operacyjny ma dostęp do dowolnego miejsca pamięci w blokach użytkowych.

Programy rezydujące w różnych blokach użytkowych są wzajemnie chronione. Procesory i kanały bezpośredniego dostępu adresują słowa pamięci poprzez podanie dwóch współrzędnych: numeru bloku NB oraz adresu słowa wewnątrz bloku (patrz odpowiednie rejestry NB i AR jednostki centralnej). Zawartość rejestru NB jednostki centralnej może być zmieniana rozkazem nielegalnym MB, efektywnym wyłącznie w obszarze systemowym pamięci. Podobnie ładowanie rejestrów kanałowych, przechowujących numer bloku pamięci operacyjnej, odbywa się pod kontrolą systemu operacyjnego. Tak więc, zainicjowany przez system operacyjny program użytkowy może operować adresami wyłącznie w obrębie przydzielonego mu bloku użytkowego.

4.3. Programowy podział pamięci operacyjnej.

Obszar użytkowy pamięci jest organizowany przez system operacyjny na drodze programowej. Konstrukcyjnie pamięć operacyjna złożona jest z modułów zawierających pakiet sterowania i blok nośnika przy czym liczba i pojemność modułów fizycznych pamięci nie jest związana z liczbą i pojemnością opisanych wyżej bloków użytkowych. W module konstrukcyjnym pamięci wyróżnia się segmenty o pojemności 4 k (maksimum 8 segmentów w module). Każdy z segmentów posiada 4-bitowy rejestr numeru bloku użytkowego oraz 4-bitowy rejestr adresu logicznego segmentu, określający usytuowanie segmentu w bloku użytkowym. Rejestry te są ładowane rozkazem OU (zgodnie z opisem zawartym w liście rozkazów), dzięki czemu system operacyjny ma możliwość dynamicznego tworzenia bloków użytkowych z segmentów występujących nawet w różnych modułach konstrukcyjnych pamięci.

Operacja zapisu/odczytu dokonywana jest w tym segmencie pamięci, dla którego wystąpi zgodność numeru bloku i 3-bitowej starszej części adresu słowa z zawartością rejestru numeru bloku i zawartością rejestru adresu logicznego segmentu (odpowiednio). 12-bitowa młodsza część adresu słowa określa miejsce pamięci w obrębie segmentu 4 k.

Programowy podział pamięci zapewnia efektywne wykorzystanie zasobów pamięci systemu.

5. ARYTMOMETR WIELOKROTNEJ PRECYZJI (arytmometr zmiennoprzecinkowy)

5.1. Przeznaczenie

Arytmometr wielokrotnej precyzji zwiększa zakres operacji wykonywanych hardware'owo przez minikomputer o następujące działania:

- dodawanie i odejmowanie liczb długich – rozkazy AD i SD
- mnożenie i dzielenie stałoprzecinkowe – rozkazy MW i DW
- normalizację, dodawanie, odejmowanie – rozkazy NRF, AF, SF, MF, DF
- mnożenie i dzielenie liczb zmiennoprzecinkowych

5.2. Działania na liczbach stałoprzecinkowych

Liczba stałoprzecinkowa traktowana jest jako liczba całkowita w notacji uzupełnionej do dwóch.

5.2.1. Dodawanie i odejmowanie liczb długich

Każdy z argumentów oraz wynik zajmują 32 bity czyli, dwa słowa maszyny. Pierwszy argument pobierany jest z rejestrów uniwersalnych jednostki centralnej R1 i R2, drugi z dwóch kolejnych komórek pamięci. Adres pierwszej komórki pamięci określony jest argumentem efektywnym rozkazu. Wyniki działań przesyłane są do R1 i R2. Rozkazy ustawiają wskaźniki Z, M, V, C w rejestrze R0 stanu programu.

- Wskaźnik Z zapalany jest w przypadku otrzymania w wyniku zera
- Wskaźnik M zapalany jest w przypadku otrzymania w wyniku liczby ujemnej
- Wskaźnik V zapalany jest w przypadku przekroczenia w wyniku zakresu liczb
- Wskaźnik C ustawiany jest zgodnie z przeniesieniem z zerowej pozycji sumatora

5.2.2. Mnożenie liczb stałoprzecinkowych

Każdy z czynników zajmuje jedno słowo, iloczyn zaś dwa słowa maszyny. Mnożna odczytywana jest z pamięci, mnożnik z rejestru R2; wynik zapisywany jest do R1 i R2. Rozkaz ustawia wskaźniki Z, M, V na podstawie wartości iloczynu, analogicznie jak w pkt. 5.2.1. Wskaźnik C pozostaje bez zmiany.

5.2.3. Dzielenie liczb stałoprzecinkowych

Dzielna zapisana w rejestrach R1 i R2, zajmuje dwa słowa maszyny. Dzielnik długości jednego słowa, zajmuje jedno słowo w pamięci; iloraz i reszta zapisywane są odpowiednio w rejestrach R2 i R1. Znak reszty zgodny jest ze znakiem dzielnej. Rozkaz dzielenia ustawia wskaźniki Z i M. Wskaźnik V i C pozostaje bez zmiany. W przypadku otrzymania ilorazu nadmiarowego oraz przy próbie dzielenia przez zero zgłaszane są przerwania. W obu wypadkach rejestry dzielnej (R1 i R2) oraz wskaźniki Z, M, V, C pozostają bez zmiany.

5.3. Działania na liczbach zmiennoprzecinkowych

Liczba zmiennoprzecinkowa zajmuje trzy słowa maszyny. Mantysa m zajmuje pierwszych czterdzieści bitów i traktowana jest jako liczba zaprzecinkowa w notacji uzupełnień do dwóch. Argumenty czterech podstawowych operacji arytmetycznych oraz wyniki wszystkich operacji są liczbami znormalizowanymi, tzn. $\frac{1}{2} < m < 1$ lub $-1 < m < -\frac{1}{2}$. Cecha zajmuje ostatnie osiem bitów liczby i traktowana jest jako liczba całkowita w notacji uzupełnień do dwóch. Wszystkie bity mantysy i cechy zerowej znormalizowanej liczby zmiennoprzecinkowej są zerami.

Poza normalizacją rozkazy zmiennoprzecinkowe są dwuargumentowe.

Argument pierwszy zajmuje rejestry uniwersalne jednostki centralnej R1, R2 i R3, argument drugi – trzy kolejne komórki pamięci.

Rozkaz normalizacji ma jeden argument, umieszczony w rejestrach R1, R2 i R3. Wyniki wszystkich rozkazów zmiennoprzecinkowych wpisywane są do rejestrów R1, R2 i R3. Na podstawie wartości wyniku ustawiane są wskaźniki Z, M i C. Wskaźnik Z zapalany jest, gdy wynik równa się zero. Rejestry R1, R2 i R3 są wówczas zerowane. Wskaźnik M jest zapalany, gdy wynik jest liczbą ujemną. Wskaźnik C przechowuje bit chroniony wyniku. Przy normalizacji jest on zawsze zerem. Wskaźnik V pozostaje bez zmiany.

Gdy któryś z argumentów jest liczbą nieznormalizowaną lub zachodzi próba dzielenia przez zero, to niezależnie od wartości drugiego argumentu zgłaszane jest przerwanie (wyjątek stanowi zero nieznormalizowane). Rejestry R1, R2 i R3 oraz wskaźniki Z, M, C pozostają wówczas bez zmiany. W przypadku otrzymania w wyniku podmiaru lub nadmiaru cechy zgłaszane są odpowiednie przerwania, co nie blokuje zapisu wyniku do rejestrów R1, R2 i R3 oraz ustawienia wskaźników Z, M, C.

5.3.1. Dodawanie i odejmowanie liczb zmiennoprzecinkowych

Jeden ze składników sumy jest odczytywany z rejestrów, drugi składnik sumy z pamięci.

Odejmna odczytywana jest z rejestrów, odjemnik z pamięci. W przypadku, gdy moduł różnicy cech argumentów jest większy lub równy 40, mniejszy argument traktowany jest jako zerowy.

Działania na mantysach wykonywane są w sumatorze przedłużonym o dalszych 40 pozycji. Po normalizacji wyniku wykonywane jest zaokrąglenie. Po zaokrągleniu wynik ponownie sprowadzony jest do postaci znormalizowanej.

5.3.2. Mnożenie i dzielenie liczb zmiennoprzecinkowych

Mnożna umieszczona jest w pamięci, mnożnik w rejestrach. Działania na mantysach czynników wykonywane są z dokładnością analogiczną jak w dodawaniu i odejmowaniu. Dzielną umieszczona jest w rejestrach, dzielnik w pamięci. Wyznacza się 40 bitów mantysy ilorazu.

5.3.3. Normalizacja

Rozkaz normalizacji pobiera liczbę zmiennoprzecinkową z rejestrów R1, R2 i R3 i umieszcza ją tam po znormalizowaniu. W procesie normalizacji mantysa uzupełniana jest zerami.

5.4. Przerwania

Rozróżniane są 4 rodzaje przerw, generowanych przez arytmometr wielokrotnej precyzji:

- nadmiar w wyniku dzielenia stałoprzecinkowego (pozycja 7 rejestru RZ)
- podmiar cechy liczby zmiennoprzecinkowej (pozycja 8 rejestru RZ)
- nadmiar cechy liczby zmiennoprzecinkowej (pozycja 9 rejestru RZ)
- nieznormalizowany argument zmiennoprzecinkowy lub dzielenie przez zero (pozycja 10 rejestru RZ).

6. KANAŁY

Procesory systemu MERA 400 komunikują się z urządzeniami i pamięciami zewnętrznymi poprzez kanały wejścia/wyjścia. W maksymalnej konfiguracji systemu może występować do 16 kanałów dowolnego typu: znakowych, pamięciowych (selektorowych), multipleksorowych i automatyki. Kanały z reguły sterują większą liczbą urządzeń i pamięci zewnętrznych poprzez jednostki sterujące (kontrolery), które dopasowują indywidualne charakterystyki sygnałów i funkcje urządzeń peryferyjnych do standardowych interfejsów kanałów.

Kanały, podobnie jak inne moduły funkcjonalne, dołączone są do interfejsu systemu. Wyróżnia się następujące transmisje interfejsu dotyczące kanałów:

Procesor	↔	Kanał	:	przesłanie, pobranie
Kanał	→	Procesor	:	przerwanie
Kanał	↔	Pamięć	:	zapis, odczyt

Kanały uzyskują dostęp do interfejsu poprzez przydzielone im elementarne układy rezerwacji połączone w łańcuch, tworzący priorytetowy układ rezerwacji. Z reguły kanałom bezpośredniego dostępu (pamięciowym, multiplexorowym) przydziela się wyższe priorytety dostępu do interfejsu w stosunku do kanałów znakowych i jednostek centralnych. Fakt ten wynika z dużych szybkości transmisji i często synchronicznego charakteru pracy pamięci zewnętrznych działających w tych kanałach. Kanały rezerwują interfejs w celu zgłaszania przerwania do jednostek centralnych oraz w przypadku bezpośredniego dostępu, w celu odczytu/zapisu informacji z/do pamięci operacyjnej.

6.1. Rozkazy WE/WY

Lista rozkazów MERA 400 zawiera dwa rozkazy wejścia/wyjścia:

OU – rozkaz przesłania i IN – rozkaz pobrania. Jednostki centralne wykorzystują rozkazy OU i IN do inicjowania operacji WE/WY w kanałach i jednostkach sterujących, pobierania informacji o stanie kanałów i jednostek sterujących, itp. Rozkaz OU powoduje w jednym cyklu rezerwacji interfejsu jednoczesne i równoległe przesłanie do kanału poniższych informacji (pytania):

– argument efektywny rozkazu N, przy czym:

$N_{\phi} \div 7$	– kod operacji WE/WY
$N_{12} \div 14$	– numer kanału,
$N_8 \div 10$	– numer jednostki sterującej,
N_{15}	– ϕ

– argument wskazany w polu A rozkazu (zawartość rejestru uniwersalnego o numerze A), zawierający zwykłe dane lub parametry operacji WE/WY, kierowanej do kanału

– numer bloku pamięci operacyjnej NB, w którym jest wykonywany rozkaz

– numer procesora wysyłającego rozkaz

– wskaźnik pracy systemu operacyjnego Q

– sygnał wiodący rozkazu S

Działanie rozkazu IN jest analogiczne z wyjątkiem kierunku przesyłania danych, które pobierane są z kanału do rejestru R/A/. Sygnałem wiodącym rozkazu IN jest sygnał F interfejsu.

W odpowiedzi na rozkaz kanały i jednostki sterujące wysyłają odpowiedź OK, EN lub OK + PN. Poniższa tabela podaje treść odpowiedzi na rozkaz w zależności od rodzaju rozkazu i stanu kanału.

Typ rozkazu	Niezgodność nr kanału lub nr j.s. lub bitu Q lub nr bloku NB	Zgodność nr kanału, nr jed. ster. i numeru bloku pao NP		
		Kanał lub j.s. nie gotowa do wykonania operacji WE/WY	Kanał i j.s. gotowa do wykonania operacji WE/WY	
			Błąd danych	
			nie	tak
OU	brak odpowiedzi	EN	OK	OK + PE
IN	brak odpowiedzi	EN	OK, dane	OK + PE, dane

W zależności od rodzaju odpowiedzi ma miejsce rozgałęzienie programu zgodnie z ustawianą zawartością licznika rozkazów IC (podana w tabeli).

Rozkazy WE/WY są rozkazami legalnymi i mogą występować w programach użytkowych. Właściwość ta umożliwia w prostych przypadkach oprogramowanie urządzeń zewnętrznych dołączonych przez użytkownika poza systemem operacyjnym. Urządzenia systemowe (zestaw objęty nadzorem systemu operacyjnego) są chronione przed programami użytkowymi. Jednostki sterujące urządzeń systemowych akceptują tylko te rozkazy WE/WY na których wskaźnik pracy systemu $Q = \emptyset$.

W przypadku dołączenia poza systemem operacyjnym urządzeń działających na rzecz różnych programów użytkowych, istnieje możliwość ochrony tych urządzeń przed wzajemną ingerencją programów użytkowych poprzez sprawdzenie w jednostkach sterujących zgodności numeru bloku pamięci operacyjnej NB.

6.2. Przerwania

W celu zgłoszenia przerwania kanał rezerwuje interfejs i przesyła do procesora sygnał przerwania wraz z numerem kanału.

Przerwanie kierowane jest do procesora aktualnie współpracującego z urządzeniem zgłaszającym przerwanie.

Przesyłane są przy tym następujące informacje:

- sygnał wiodący przerwanie IN
- numer kanału zgłaszającego przerwanie
- numer procesora, do którego jest kierowane przerwanie

W odpowiedzi na sygnał przerwania procesor zawsze wysyła sygnał odpowiedzi pozytywnej – OK. W rezultacie zgłoszenia przerwania, zapala się odpowiednia pozycja rejestru zgłoszeń RZ procesora, związana z numerem kanału. Gdy warunki przyjęcia przerwania są spełnione, procesor przerywa pracę bieżącego programu, automatycznie rezerwuje interfejs, pobiera z kanału specyfikację przerwania, po czym zapisuje ją wraz ze stanem maszyny na stos i wykonuje skok do miejsca pamięci związanego z numerem obsługiwanego pozycji rejestru RZ. Specyfikacja przerwania zawiera numer jednostki sterującej zgłaszającej przerwanie w danym kanale oraz kod przerwania. W kanałach i jednostkach sterujących systemem MERA 400 przyjęto zasadę kodowania przerwania, polegającą na przydzielaniu poszczególnym przyczynom przerwania kolejnych liczb binarnych. Podprogramy obsługi przerwania zewnętrznych mogą wykorzystywać specyfikacje przerwania do tworzenia tablic skoków, uwarunkowanych przyczynami przerwania. Czas reakcji systemu na przerwanie sprowadzony jest do minimum potrzebnego do ewentualnego zapamiętania zawartości rejestrów uniwersalnych.

Numery kanałów określające priorytet obsługi przerwania tych modułów, ustalone są podczas instalacji systemu poprzez ustawienie zworek na pakietach. Priorytet obsługi przerwania kanału jest niezależny od priorytetu dostępu do interfejsu, który wynika z miejsca włączenia kanału w łańcuch rezerwacji interfejsu.

Szczegółowy opis mechanizmu zgłaszania przerwania oraz wykaz operacji WE/WY i specyfikacji przerwania podstawowych urządzeń i pamięci zewnętrznych zawiera LISTA ROZKAZÓW MERA 400 (rozdz.3).

6.3. Kanał znakowy

Kanał znakowy przeznaczony jest do podłączenia ośmiu urządzeń zewnętrznych pracujących start-stopowo lub synchronicznie z niewielką szybkością transmisji danych. Praktycznie, górna granica szybkości transmisji nie przekracza 20 tys. znaków na sekundę. Kanał znakowy, będący modułem pośredniczącym między jednostką centralną i jednostkami sterującymi urządzeniami zewnętrznymi, spełnia niżej wymienione funkcje:

- zapewnia przesyłanie danych znakami, słowami lub dowolnym formatem informacji nie przekraczających 16 bitów pomiędzy rejestrami uniwersalnymi procesora i rejestrami buforowymi jednostek sterujących (lub bezpośrednio urządzeniami zewnętrznymi),
- koncentruje przerwania z jednostek sterujących i zgłasza je poprzez główną szynę informacyjną do procesora. Rejestry alokacji zawarte w kanale, zapewniają kierowanie przerwania do odpowiednich jednostek centralnych,
- centralnie dekoduje numery jednostek sterujących i wybrane operacje WE/WY w celu wypracowania indywidualnych sygnałów wiodących rozkazów (strobów) dla poszczególnych jednostek sterujących,
- separuje i wzmacnia sygnały interfejsu systemu.

Każde przesłanie danych lub informacji sterującej odbywa się na rozkaz z jednostki centralnej. Rozkazy OU i IN są dekodowane na poziomie kanałów lub jednostek sterujących i inicjują wykonanie operacji WE/WY, charakterystycznych dla danego typu urządzenia.

W systemie MERA 400, mając na uwadze minimalizację ilości przerwania od urządzeń zewnętrznych, przyjęto zasadę warunkowego zgłaszania przerwania z jednostek sterujących po zakończeniu operacji WE/WY. Operacja WE/WY kończy się przerwaniem tylko w tym przypadku, gdy podczas jej wykonywania jednostka centralna ponowiła rozkaz WE/WY i został on odrzucony odpowiedzią negatywną EN (ze względu na zajętość jednostki sterującej).

6.4. Kanał pamięciowy

Kanał pamięciowy zapewnia prowadzenie autonomicznej transmisji danych między pamięcią operacyjną a jednostkami sterującymi szybkich urządzeń pamięciowych po zainicjowaniu jej przez procesor rozkazem OU. Kanał pamięciowy pośredniczy także w przekazywaniu rozkazów sterujących z procesora do jednostek sterujących oraz w zgłaszaniu przerwania z jednostek sterujących do procesora.

Do kanału pamięciowego może być dołączonych 8 jednostek sterujących. Kanał umożliwia dołączanie do systemu szybkich urządzeń działających asynchronicznie lub synchronicznie. Maksymalna szybkość transmisji danych poprzez kanał nie przekracza 500 tys. słów 16-bitowych na sekundę. Kanał pamięciowy może współpracować z dwoma procesorami.

Głównym zadaniem kanału jest zapewnienie prawidłowego przekazywania informacji z urządzenia pamięciowego do pamięci operacyjnej lub odwrotnie, oraz przesyłanie rozkazów sterujących.

W kanale realizowane są następujące rodzaje transmisji:

- czytanie informacji z pamięci zewnętrznej i zapisywanie jej do pamięci operacyjnej
- przepisywanie informacji z pamięci operacyjnej do pamięci zewnętrznej
- czytanie z porównaniem, służące do sprawdzania prawidłowości poprzedniego zapisu. W operacji tej kanał czyta kolejne słowa jednocześnie z pamięci operacyjnej i pamięci zewnętrznej celem ich porównania.

Kanał pamięciowy jest kanałem selektorowym i może prowadzić jednocześnie transmisję tylko z jednym urządzeniem. W czasie transmisji może przyjmować rozkazy nietransmisyjne oraz zgłaszać przerwania i przysyłać specyfikacje dotyczące innych urządzeń.

Kanał realizuje transmisję w czterech fazach:

- 1^o – przyjęcie rozkazu typu OK inicjującego czynności związane z transmisją bloku danych
- 2^o – autonomiczne pobranie przez kanał pola sterującego z pamięci operacyjnej
- 3^o – właściwa transmisja bloku danych
- 4^o – zgłoszenie przerwania informującego procesor o zakończeniu transmisji.

Rozkazy przyjmowane przez kanał pamięciowy mogą dotyczyć samego kanału lub urządzeń.

6.4.1. Rozkazy dotyczące samego kanału

Rozkazy typu OU:

- Sprawdź istnienie kanału
- Blokuj przerwania do procesora wysyłającego rozkaz
- Blokuj przerwania do drugiego procesora
- Przydziel urządzenie procesorowi wysyłającemu rozkaz

Rozkazy typu IN:

- Sprawdź istnienie kanału
- Podaj specyfikację przerwania

- Podaj stan licznika słów przetransmitowanych
- Podaj stan rejestru alokacji urządzeń

6.4.2. Rozkazy transmisji

Rozkazy transmisyjne są rozkazami typu OU. Argument wskazany w polu A rozkazu zawiera adres początku pola sterującego transmisją. Długość pola sterującego zależy od urządzenia, ale nie może być większa od 16 i mniejsza od 4. Ustalona jest zawartość słów Nr 0, 1, 2 oraz ostatniego:

- zerowe słowo zawiera długość pola sterującego, numer bloku pamięci operacyjnej, z którą ma być prowadzona transmisja bądź numer procesora, jeśli transmisja będzie prowadzona z blokiem zerowym (systemowym)
- słowo 1 definiuje typ operacji transmisji: zapis lub odczyt z urządzenia, odczyt do przodu lub do tyłu, odczyt z porównaniem, zapis/odczyt adresów lub danych, itp.
- słowo 2 zawiera długość transmisji w słowach
- słowo ostatnie zawiera adres początku transmisji w bloku pamięci operacyjnej określonym w słowie zerowym.

Pozostałe słowa sterujące zawierają informacje wykorzystywane przez urządzenie (np. numer cylindra lub głowicy w pamięci dyskowej, itp.).

6.4.3. Rozkazy sterujące

Rozkazy sterujące są przeznaczone dla urządzeń. Informacja sterująca może być przesyłana bądź bezpośrednio w czasie odbierania rozkazu, bądź za pośrednictwem bufora kanałowego. W tym pierwszym przypadku rozkazy sterujące mogą być przesyłane także w trakcie transmisji.

6.4.4. Jednostka sterująca pamięciami dyskowymi MERA 9425

Jednostka sterująca zapewnia współpracę między kanałem pamięciowym, a pamięciami dyskowymi MERA 9425. Standardowy układ jednostki sterującej przeznaczony jest do współpracy z dwoma modułami pamięci dyskowych. Przewidziano możliwość dołączenia dwu dalszych modułów. Jednostka sterująca jest układem selektywnym, tzn. umożliwia prowadzenie transmisji z jedną wybraną pamięcią dyskową. Niezależnie od tego operacje ruchu pozycjonera mogą być wykonywane jednocześnie w różnych modułach pamięci dyskowej.

Dane w pamięci dyskowej zapisywane są wg określonego formatu. Każda ścieżka podzielona jest na 14 sektorów. Na początku każdego sektora znajduje się pole adresowe składające się z 4 słów 16-to bitowych. W polu tym podczas operacji pisania adresów zapisuje się:

- bity identyfikatora, numer cylindra
- numer talerza, numer głowicy, numer sektora
- klucz (słowo protekcji danych)
- słowo statusu (do użytku programisty).

Pole adresowe wykorzystywane jest przez jednostkę sterującą do kontroli wybranego adresu przy operacjach czytania i pisania danych.

Oprócz pola adresowego sektor zawiera pole danych o pojemności 256 słów 16-to bitowych.

Jednostka sterująca sprawdza poprawność adresów i danych przy operacjach zapisu/odczytu dzięki układowi kontroli cyklicznej. Jednostka sterująca umożliwia transmisję danych blokami informacyjnymi o dowolnej długości nie przekraczającej pojemności jednego cylindra (6144 słów 16-to bitowych). Wykrywa również znacznik końca zbioru, zapisywany jako pierwsze słowo danych w sektorze i kończy transmisję na tym sektorze. Możliwa jest również transmisja bez wykrywania znacznika.

kasety: stała i wymienna są adresowane oddzielnie.

7. KANAŁ AUTOMATYKI PI

7.1. Przeznaczenie

Blok sprzężenia PI-MERA 400 jest adapterem łączącym magistralę systemu "INTELDIGIT-PI" z interfejsem m

nikomputera MERA 400. Blok sprzężenia stanowi wraz z innymi blokami PI tzw. Kanał Automatyki PI mini-komputera, będąc jednocześnie jego modułem sterującym. Blok sprzężenia jako element systemu MERA 400 przeznaczony jest do sterowania programowego przez jednostkę centralną modułami systemu PI przy pomocy odpowiedniego repertuaru operacji oraz do współpracy z modułem testującym PI.

Minikomputer MERA 400 wraz z Kanałem Automatyki PI przeznaczony jest do tworzenia systemów Centralnej Rejestracji i Sterowania pracujących w czasie rzeczywistym w układach otwartych lub zamkniętych.

7.2. Opis funkcjonalny

Blok sprzężenia jako sterownik kanału automatyki spełnia następujące funkcje:

- przyjmuje i wykonuje rozkazy z jednostki centralnej
- generuje odpowiednie sygnały do magistrali PI realizując dwukierunkową transmisję danych
- przyjmuje zgłoszenia z kaset PI, określa priorytet i generuje sygnał przerwania do jednostki centralnej
- blokuje poszczególne zgłoszenia z PI

Każda transmisja danych pomiędzy rejestrami wybranego modułu kasety PI i jednostki centralnej wymaga użycia tylko jednego rozkazu we/wy. Wynik transmisji przekazywany jest w czasie trwania cyklu rozkazowego. Czas trwania jednego cyklu transmisji danych wynosi poniżej 2,5 us.

W zestawie MERA 400 PI może pracować do 8 bloków sprzęgających, z których każdy możeysterować zestawem do 16 kaset PI.

7.3. Podstawowe dane o systemie PI sterowanym przez blok sprzężenia

Poniższe dane zaczerpnięto z dokumentu "Sprzężenia Komputerów z Elementami Automatyki i Pomiarów – Urządzenia INTEL DLIGIT – PI" Wyd. Przemysłowy Instytut Automatyki i Pomiarów "MERA – PIAP".

PI jest zestawem urządzeń do sprzężenia komputerów z elementami automatyki i pomiarów. Urządzenia te umożliwiają automatyzację pomiarów lub sterowania dowolnego obiektu lub procesu, zawsze przy zastosowaniu komputera, wykonującego całość zadań przetwarzania informacji i sterowania przekazywaniem informacji.

Urządzenia PI umożliwiają efektywną, opłacalną ekonomicznie automatyzację obiektów różnych wielkości – od pojedynczych stanowisk do wielkich zakładów, budowę układów rejestracji i sterowania z zastosowaniem mini-komputerów.

Do preferowanych zastosowań urządzeń PI należą:

- obsługa zautomatyzowanych stanowisk pomiarowych i kontrolnych w laboratoriach zakładowych i naukowo-badawczych;
- obsługa produkcyjnych stanowisk kontrolnych i sortujących;
- centralna rejestracja i przetwarzanie danych;
- sterowanie procesów w systemie doradczym i bezpośrednim;
- kontrola i sterowanie procesów przesyłania i dystrybucji w systemach energetycznych oraz przy transporcie cieczy, gazów i materiałów sypkich (telemechanika);
- sterowanie cyfrowe maszyn i agregatów produkcyjnych;
- automatyczne sterowanie składowaniem i magazynowaniem.

System PI ma strukturę pakietową. Poszczególne pakiety, wykonujące zadania sprzężenia komputera z obiektem, składają się najczęściej z jednej płytki o wymiarach zgodnych z wymaganiami JS EMC (140 x 150 mm).

W celu uproszczenia obsługi, obniżenia łącznych kosztów automatyzacji obiektu i wzrostu niezawodności, PI przenosi możliwie dużo zadań na sprzęt i oprogramowanie komputera. System jest jednostopniowy, charakteryzuje się bezpośrednim adresowaniem każdego pakietu przez komputer i wymianą informacji między komputerem i pakietem. Układy pośredniczące (sterownik kasety) nie przetwarzają informacji.

Ogólną strukturę i powiązania urządzeń PI w zestawie przedstawia rysunek na stronie 2–38.

Zestaw PI, stanowiący dla komputera jeden kanał wejścia – wyjścia, obejmuje: jeden blok sprzęgający i od 1 do 16 kaset z sterownikami kaset. Maksymalna pojemność zestawu PI wynosi 256 stanowisk adresowanych.

Magistrala zestawu jest to wieloprzewodowa sieć bierna, łącząca sterownik kaset z blokiem sprzęgającym, wykonana w postaci dwóch giętkich kabli, ze złączami typu DD50.

Kaseta jest to standardowa konstrukcja mechaniczna, służąca do umieszczenia pakietów i sterownika kasety. PI stosuje kasety 19-calowe wg standardów IEC o wymiarach 483 x 279 x 221 mm/5U/, zawierające 18 stanowisk o module 22 mm. Dwa skrajne stanowiska z prawej strony zajmuje sterownik kasety. Pozostałe stanowiska o adresach 00 . . . 15 wypełniają pakiety. Dla zestawów wielokasetowych stosuje się zamiast pojedynczych kaset ramy odchylne wielosegmentowe zawierające do 8 segmentów. Każdy segment ramy jest odpowiednikiem kasety.

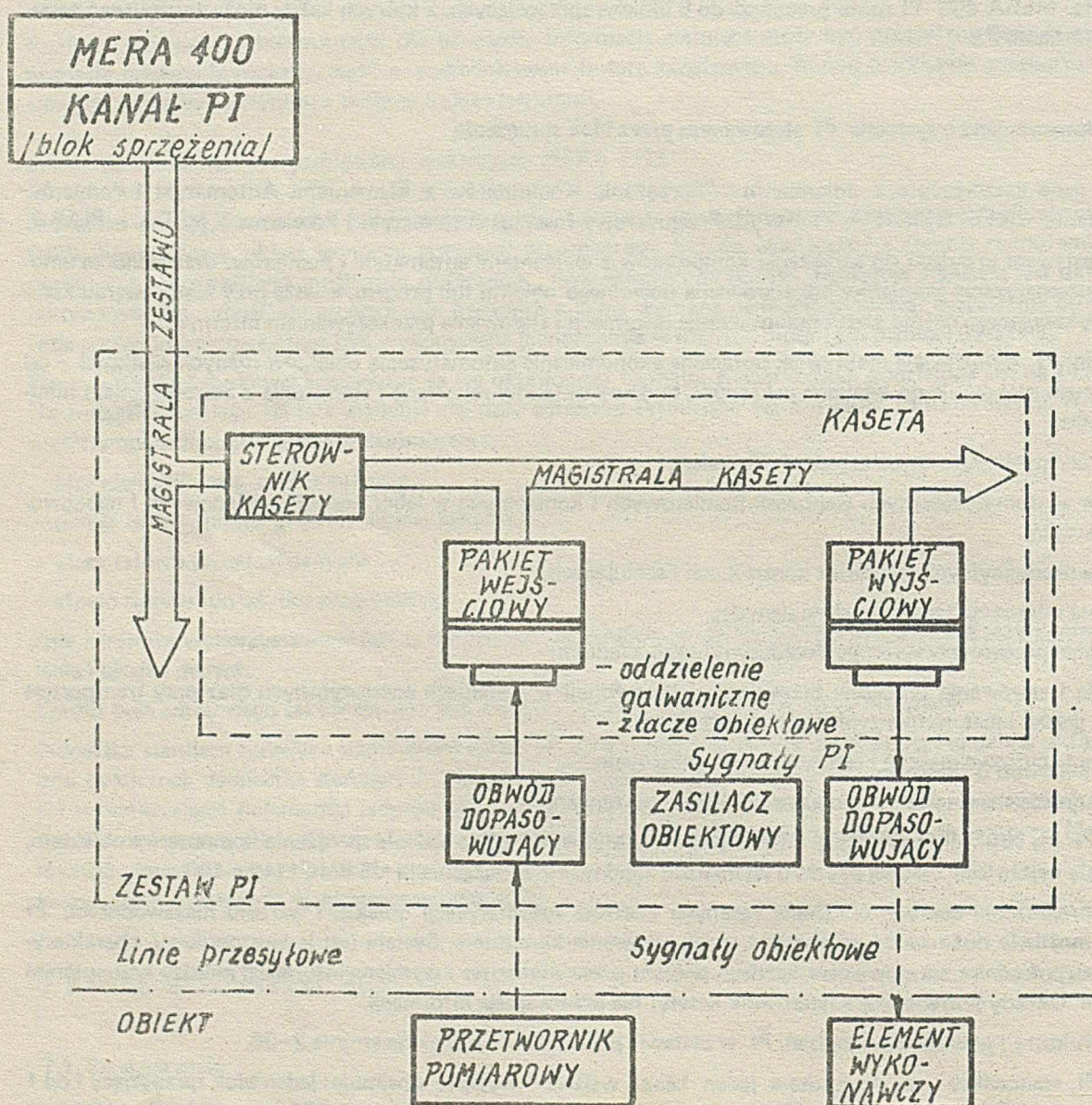
Sterownik kasety jest rozgałęźnikiem magistrali PI o działaniu przepływowym, przekazuje dwukierunkowo sygnały między magistralą kasety, a magistralą zestawu PI.

Magistrala kasety jest to wieloprzewodowa sieć bierna, łącząca styki gniazd pakietów ze stykami gniazda sterownika kasety (złącza typu G06D64A3BDBL), wykonana metodą owijania w postaci stałego okablowania złączy na tylnej ścianie kasety.

Pakiet wejściowy jest pakietem sprzęgającym, przyjmującym sygnały z obiektu doprowadzone za pośrednictwem złącza lub złączy obiektowych; najczęściej pakiet wykonuje także oddzielenie galwaniczne.

Pakiet wyjściowy jest pakietem sprzęgającym, wydającym do obiektu sygnały, wyprowadzane za pośrednictwem złącza lub złączy obiektowych; najczęściej pakiet wykonuje także oddzielenie galwaniczne.

Obwody dopasowujące zapewnią dopasowanie między sygnałami obiektowymi a sygnałami PI.



Ogólna struktura zestawu MERA-400 - PI

LISTA ROZKAZÓW MERA 400 (objaśnienie symboli)

- A – numer rejestru uniwersalnego wskazany w polu A rozkazu (pozycje 7÷9 słowa rozkazowego)
- B – numer rejestru uniwersalnego (indeksowego) wskazany w polu B rozkazu (pozycje 10÷12)
- b – argument bajtowy rozkazu (poz. 8÷15 rozkazu)
- C – liczba zawarta w polu C rozkazu wskazująca argument pierwotny rozkazu
 gdy C=1,2...7 – arg. pierwotnym jest R/C/
 gdy C=0 – arg. pierwotnym jest M
- D – bit pośredniego adresowania
- IC – licznik rozkazów; jeśli nie wskazano inaczej, rozkaz powoduje: $IC=IC + \text{długość rozkazu}$
- M – zawartość następnego słowa za słowem podstawowym rozkazu
- MOD – modyfikator, wynik działania rozkazu MD
- N – argument efektywny rozkazu obliczany wg wzoru
 gdy D=0 $N = R/C/ + M + MOD + R/B/$
 gdy D=1 $N = \underbrace{S/R/C/}_{\text{składniki sumy}} + \underbrace{M}_{\text{składniki sumy}} + \underbrace{MOD}_{\text{składniki sumy}} + \underbrace{R/B/}_{\text{składniki sumy}}$
 występują gdy $C \neq \emptyset$ $C = \emptyset$ był MD $B \neq 0$
- P – wskaźnik przeskoku; gdy P=1 to $IC=IC+dł.roz.+1$
- $R\emptyset \div R7$ – zawartość rejestrów uniwersalnych
- rA,rB,rC – oznaczenia argumentów rozkazu, związanych z polami A,B i C rozkazu
- R/A/,R/B/,R/C/ – zawartość rejestrów uniwersalnych o numerach A,B i C (odpowiednio)
- RZ – 32-pozycyjny rejestr zgłoszeń przerwai; przeznaczenie pozycji rejestru RZ:
- | | |
|------------------------------------------|----------------------------------|
| 0 – alarm zasilania CPU | 8 – podmiar zm.przecink. |
| 1 – parzystość pamięci | 9 – nadmiar zm.przecink. |
| 2 – brak pamięci | 10 – :0, bł. danych zm.przecink. |
| 3 – inny procesor – U | 11 – |
| 4 – alarm zasilania I/F | 12 ÷ 27 – przerwania kanałowe |
| 5 – zegar | 28 – OPRQ zgł. operatora |
| 6 – rozkaz nieprawidłowy | 29 – inny procesor – L |
| 7 – nadmiar dzielenia stałoprzecinkowego | 30 – program |
| | 31 – program |
- SR – 16-pozycyjny rejestr stanu systemu
- RM – (pozycje $\emptyset \div 9$ SR) – rejestr masek przerwai
 Pogrupowanie przerwai w poziomy obsługi:
- | | | | | | | | | | | | |
|---------|---|---|---|---|---|--------|---------|---------|---------|---------|---------|
| poziom | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| poz. RM | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | |
| poz. RZ | 0 | 1 | 2 | 3 | 4 | 5 ÷ 11 | 12 ÷ 13 | 14 ÷ 15 | 16 ÷ 21 | 22 ÷ 27 | 28 ÷ 31 |
- Q – (pozycja 10 SR) – wskaźnik pracy systemu
 stan Q = \emptyset odpowiada pracy systemu operacyjnego
 stan Q = 1 odpowiada pracy programu użytkowego
- BS – (poz.11 SR) – wskaźnik zezwolenia na dostęp do bloku systemowego pamięci innego procesora
- NB – (poz. 12÷15 SR) – rejestr numeru bloku pamięci
- S/x;NB/ – zawartość słowa pamięci o adresie x w bloku pamięci o numerze NB
- S/x/ – zawartość słowa pamięci o adresie x
 a) w bloku systemowym, gdy wskaźnik Q= \emptyset
 b) w bloku użytkowym o numerze NB, gdy Q=1
- T – argument krótki rozkazu zawarty w polach D (znak) i BC (6-bitowa wartość) rozkazu
 Wskaźniki zawarte w rejestrze $R\emptyset$:
- Z – (poz. \emptyset $R\emptyset$) – wskaźnik zera
- M – (poz. 1 $R\emptyset$) – wskaźnik znaku minus
- V – (poz. 2 $R\emptyset$) – wskaźnik nadmiaru
- C – (poz. 3 $R\emptyset$) – wskaźnik przeniesienia
- L – (poz. 4 $R\emptyset$) – wskaźnik mniejszości
- E – (poz. 5 $R\emptyset$) – wskaźnik równości

LISTA ROZKAZÓW MERA 400 (objaśnienie symboli)

- G — (poz. 6 RO) — wskaźnik większości
 Y — (poz. 7 RO) — wskaźnik przechowania bitu
 X — (poz. 8 RO) — wskaźnik ustawiany programowo

Znaki specjalne:

- znak użyty nad symbolem oznacza negację
 ^ — iloczyn logiczny
 v — suma logiczna
 : — różnica symetryczna
 ≡ — tożsamość
 ↵ — przesunięcie o 1 bit w lewo
 ↷ — przesunięcie o 1 bit w prawo

LISTA ROZKAZÓW MERA 400 (Skorowidz kodowy rozkazów)
 ☆ — kody w polach rozkazu w zapisie oktalnym

Pola rozkazu				Pola rozkazu					
Kod	D	A	B C	Kod	D	A B C	Nazwa	Lp	Leg
☆									
20	—	----		72	0	—11	RPC	105	
21	—	----		72	1	—00	RKY	101	
22	—	----		72	1	—01	ZRB	69	
23	—	----		72	1	—02	SXL	55	
24	—	----		72	1	—03	NGC	34	
25	—	----		72	1	—04	SVZ	59	
26	—	----		72	1	—05	SVY	60	
27	—	----		72	1	—06	SVX	61	
30	—	----		72	1	—07	SRX	64	
31	—	----		72	1	—10	SRZ	62	
32	—	----		72	1	—11	LPC	106	
33	—	----		72	1	—2—	SHC	65	
34	—	----		73	—	0—	HLT	112	NL
35	—	----		73	—	1—	MCL	113	NL
36	—	----		73	0	200	CIT	116	NL
37	—	0—		73	0	201	SIL	115	NL
37	—	1—		73	0	202	SIU	114	NL
37	—	2—		73	0	203	SIT	117	NL
37	—	3—		73	0	3—	GIU	118	NL
37	—	4—		73	1	3—	GIL	119	NL
37	—	5—		73	—	4—	LIP	120	NL
37	—	6—		74	—	0—	UJ	71	
37	—	7—		74	—	1—	JL	76	
40	—	----		74	—	2—	JE	78	
41	—	----		74	—	3—	JG	80	
42	—	----		74	—	4—	JZ	82	
43	—	----		74	—	5—	JM	83	
44	—	----		74	—	6—	JN	75	
45	—	----		74	—	7—	LJ	73	
46	—	----		75	—	0—	LD	4	
47	—	----		75	—	1—	LF	5	
50	—	----		75	—	2—	LA	6	

LISTA ROZKAZÓW MERA 400
Orientacyjne czasy wykonania rozkazów

Czasy wykonywania rozkazów wydłużają się przy premodyfikacji o $0,3 \mu s$, B – modyfikacji o $0,3 \mu s$, D – modyfikacji o $1,05 \mu s$, działaniach na argumentach zawartych w pamięci operacyjnej o $0,85 \mu s$. Podane czasy obowiązują dla minikomputera MERA 400 z wykorzystaniem elementów scalonych typu H, wyposażonego w pamięć operacyjną rdzeniową z czasem cyklu $1,1 \mu s$. Rozkazów nielegalnych nie uwzględniono.

Nazwa rozkazu	czas w μs	Nazwa rozkazu	czas w μs
AC	1,75	NGA	1,65
AD	5,2	NGC	1,65
AF	$5,4 \div 19^*$	NGL	1,35
AW	1,75	NM	3,4
AWT	1,75	NR	1,7
BB	$1,8 \div 3,2^P$	NRF	$3,8 \div 15,8$
BC	$1,8 \div 3,2^P$	OM	3,4
BLC	$1,7 \div 3,1^P$	OR	1,7
BM	$2,8 \div 4,2^P$	OU	$2 \div 5^R$
BN	$1,8 \div 3,2^P$	PA	8,3
BRC	$1,7 \div 3,1^P$	PD	3,5
BS	$1,8 \div 3,2^P$	PF	4,5
CB	3,85	PL	4,5
CL	1,8	PW	2,2
CW	1,75	RA	8,3
CWT	1,8	RB	4,4
DF	$6,8 \div 30^*$	RD	3,5
DW	$4,9 \div 15,3^*$	RF	4,5
DRB	2,2	RI	2,6
EM	3,5	RIC	1,4
ER	1,7	RJ	1,7
EXL	V	RKY	1,4
FI	2,4	RL	4,5
IB	$3,6 \div 5^P$	RPC	1,3
IM	2,4	RW	2,2
IN	$2 \div 5^R$	RWS	2,5
IRB	2,2	RZ	2,2
IS	$2,8 \div 4,2^P$	SD	5,2
JCS	2,0	SF	$5,4 \div 19^*$
JE	1,65	SHC	$1,9 \div 3,5^*$
JES	2,0	SLX	2,0
JG	1,65	SLY	2,0
JGS	2,0	SLZ	2,0
JL	1,65	SP	2,4
JLS	2,0	SRX	2,0
JM	1,65	SRY	2,0
JN	1,65	SRZ	2,0
JXS	2,0	SVX	2,0
JYS	2,0	SVY	2,0
JZ	1,65	SVZ	2,0
KI	2,2	SW	1,75
LA	9,0	SXL	1,35
LB	3,8	SXU	1,35
LF	4,9	TA	9,0
LD	3,85	TD	3,85
LJ	2,5	TF	4,9
LL	4,9	TL	4,9
LPC	1,3	TRB	$1,75 \div 3,15^P$
LS	2,6	TW	1,65

Nazwa rozkazu	czas w μs	Nazwa rozkazu	czas w μs
LW .	1,65	UJ	1,65
LWS	2,85	UJS	2,0
LWT	1,7	XM	3,5
MF	6,8 ÷ 29 [*]	XR	1,8
MB	2,4	ZLB	1,4
MW	3,5 ÷ 11,9 [*]	ZRB	1,4

★ – czas zależny od argumentów, P – od przeskoku, V – różny, zależny od ekstrakodu, R – od reakcji jednostek sterujących urządzeniami zewnętrznymi.

- 1/ - zamiast litery "n" podstawić przyrostki: R - gdy D=0, C≠0 A - gdy D=1, C≠0 D - gdy D=0, C=0
 l - gdy D=1, C=0
 2/ - zamiast argumentu rC w przypadku C=0 występuje M (słowo następne za słowem rozkazowym)
 3/ - argument rB nie występuje w przypadku B=0 lub gdy rozkaz nie może ulegać B-modyfikacji

Lp.	Zapis symboliczny (assembler)	Nazwa rozkazu	Zapis oktalny kod D'ABC	Treść rozkazu	Ustaw wskaźnik
1	LWn ¹ ,rA,rC ² ,rB ³)	Umieścić w rejestrze	20 D ABC	R/A/:=N	-
2	LWT,rA,T	Umieścić krótki argument	65 T ATT	R/A/:=T	-
3	LWS,rA,T	Pobierz względnie	66 T ATT	R/A/:=S/IC+T/	-
4	LDn,rC,rB	Umieść w R1 i R2	75 D 0BC	[R1,R2]:=[S/N/,S/N+1/]	-
5	LFn,rC,rB	Umieść w R1,R2 i R3	75 D 1BC	[R1,R2,R3]:=[S/N/,S/N+1/,S/N+2/]	-
6	LAn,rC,rB	Umieść w R1,R2,...,R7	75 D 2BC	[R1,R2,...,R7]:=[S/N/S/N+1/,...,S/N+6/]	-
7	LLn,rC,rB	Umieść w R5,R6 i R7	75 D 3BC	[R5,R6,R7]:=[S/N/,S/N+1/,S/N+2/]	-
8	LSn,rA,rC,rB	Umieść przez maskę wg R7	22 D ABC	R/A/:=[R/./∧R7]M[N∧R7]	-
9	TWn,rA,rC,rB	Umieść wg NB	21 D ABC	R/A/:=S/N;NB/	-
10	TDn,rC,rB	Pobierz do R1 i R2	75 D 4BC	[R1,R2]:=[S/N;NB/,S/N+1;NB/]	-
11	TFn,rC,rB	Pobierz do R1,R2 i R3	75 D 5BC	[R1,R2,R3]:=[S/N;NB/,S/N+1;NB/,S/N+2;NB/]	-
12	TAn,rC,rB	Pobierz do R1,R2,...,R7	75 D 6BC	[R1,R2,...,R7]:=[S/N;NB/,...,S/N+6;NB/]	-
13	TLn,rC,rB	Pobierz do R5,R6 i R7	75 D 7BC	[R5,R6,R7]:=[S/N;NB/,S/N+1;NB/,S/N+2;NB/]	-
14	RWn,rA,rC,rB	Pamiętaj rejestr	24 D ABC	S/N/:=R/A/	-
15	RWS,rA,T	Pamiętaj względnie	67 T ATT	S/IC+N/:=R/A/	-
16	RDn,rC,rB	Pamiętaj R1 i R2	76 D 0BC	[S/N/,S/N+1/]:=[R1,R2]	-
17	RFn,rC,rB	Pamiętaj R1,R2 i R3	76 D 1BC	[S/N/,S/N+1/,S/N+2/]:=[R1,R2,R3]	-
18	RAn,rC,rB	Pamiętaj R1,R2,...,R7	76 D 2BC	[S/N/,S/N+1/,...,S/N+6/]:=[R1,R2,...,R7]	-
19	RLn,rC,rB	Pamiętaj R5,R6 i R7	76 D 3BC	[S/N/,S/N+1/,S/N+2/]:=[R5,R6,R7]	-
20	PWn,rA,rC,rB	Pamiętaj wg NB	25 D ABC	S/N;NB/:=R/A/	-
21	PDn,rC,rB	Prześlij R1 i R2	76 D 4BC	[S/N;NB/,S/N+1;NB/]:=[R1,R2]	-
22	PFn,rC,rB	Prześlij R1,R2 i R3	76 D 5BC	[S/N;NB/,S/N+1;NB/,S/N+2;NB/]:=[R1,R2,R3]	-
23	PAn,rC,rB	Prześlij R1,R2,...,R7	76 D 6BC	[S/N;NB/,...,S/N+6;NB/]:=[R1,R2,...,R7]	-
24	PLn,rC,rB	Prześlij R5,R6 i R7	76 D 7BC	[S/N;NB/,S/N+1;NB/,S/N+2;NB/]:=[R5,R6,R7]	-
25	RZn,rC,rB	Zeruj słowo w pamięci	77 D 6BC	S/N/:=0	-
26	RIn,rA,rC,rB	Umieść w pamięci	23 D ABC	S/R/A//:=N R/A/:=R/A/+1	-
27	AWn,rA,rC,rB	Dodaj	40 D ABC	R/A/:=R/A/+N	Z,M,V,C
28	AWT,rA,T	Dodaj krótki argument	60 T ATT	R/A/:=R/A/+T	Z,M,V,C
29	ACn,rA,rC,rB	Dodaj z przeniesieniem	41 D ABC	R/A/:=R/A/+N+C	Z,M,V,C
30	SWn,rA,rC,rB	Odejmij	42 D ABC	R/A/:=R/A/-N	Z,M,V,C
31	CWn,rA,rC,rB	Porównaj	43 D ABC	R/A/≐N	L,E,G
32	CWT,rA,T	Porów naj z krótkim arg.	64 T ATT	R/A/≐T	L,E,G
33	NGA,rA	Neguj arytmetycznie	72 0 A03	R/A/:=R/A/+1	Z,M,V,C
34	NGC,rA	Neguj z przeniesieniem	72 1 A03	R/A/:=R/A/+C	Z,M,V,C
35	ADn,rC,rB	Dodaj liczby długie	37 D 0BC	[R1,R2]:=[R1,R2]+[S/N/S/N+1/]	Z,M,V,C
36	SDn,rC,rB	Odejmij liczby długie	37 D 1BC	[R1,R2]:=[R1,R2]-[S/N/S/N+1/]	Z,M,V,C
37	MWn,rC,rB	Mnóż	37 D 2BC	[R1,R2]:=R2xS/N/	Z,M,V
38	DWn,rC,rB	Dziel	37 D 3BC	R2:=[R1,R2]:S/N/ R1:= reszta	Z,M
39	AFn,rC,rB	Dodaj liczby zmiennoprz.	37 D 4BC	[R1,R2,R3]:=[R1,R2,R3]+[S/N/S/N+1/,S/N+2/]	Z,M,C
40	SFn,rC,rB	Odejmij liczby zm.prz.	37 D 5BC	[R1,R2,R3]:=[R1,R2,R3]-[S/N/S/N+1/,S/N+2/]	Z,M,C
41	MFn,rC,rB	Pomnóż liczby zm.prz.	37 D 6BC	[R1,R2,R3]:=[R1,R2,R3]x[S/N/S/N+1/,S/N+2/]	Z,M,C
42	DFn,rC,rB	Podziel liczby zm.prz.	37 D 7BC	[R1,R2,R3]:=[R1,R2,R3]:[S/N/S/N+1/,S/N+2/]	Z,M,C
43	NRF	Normalizuj	71 1 700	[R1,R2,R3]:= znormalizowana liczba zm.prz.	-
44	ORn,rA,rC,rB	Dodaj logicznie	44 D ABC	R/A/:=R/A/∨N	Z
45	OMn,rA,rC,rB	Dodaj log. do pamięci	45 D ABC	S/N;NB/:=S/N;NB/∨R/A/	Z
46	NRn,rA,rC,rB	Pomnóż logicznie	46 D ABC	R/A/:=R/A/∧N	Z
47	NMn,rA,rC,rB	Pomnóż log. do pamięci	47 D ABC	S/N;NB/:=S/N;NB/∧R/A/	Z
48	XRn,rA,rC,rB	Różnica symetryczna	52 D ABC	R/A/:=R/A/⊕N	Z
49	XMn,rA,rC,rB	Różnica sym. w pamięci	52 D ABC	S/N;NB/:=S/N;NB/⊕R/A/	Z
50	ERn,rA,rC,rB	Zeruj bity	50 D ABC	R/A/:=R/A/∧N	Z
51	EMn,rA,rC,rB	Zeruj bity w pamięci	51 D ABC	S/N;NB/:=S/N;NB/∧R/A/	Z
52	CLn,rA,rC,rB	Porównaj logicznie	54 D ABC	R/A/≐N	L,E,G
53	NGL,rA	Neguj logicznie	72 0 A10	R/A/:=R/A/	Z
54	SXU,rA	Ustaw wsk. X wg R/A/0	72 0 A02	X:=R/A/0	X
55	SXL,rA	Ustaw wsk. X wg R/A/15	72 1 A02	X:=R/A/15	X

lp.	Zapis symboli	Nazwa rozkazu	Kod D ABC	Treść rozkazu	ust.wsk.
56	SLZ, rA	Przesuń w lewo	72 0 A04	$R/A/\bar{7}[R/A,0] Y:=R/A/0$	Y
57	SLY, rA	Przesuń w lewo z Y	72 0 A05	$R/A/\bar{7}[R/A,Y] Y:=R/A/0$	Y
58	SLX, rA	Przesuń w lewo z X	72 0 A06	$R/A/\bar{7}[R/A,X] Y:=R/A/0$	Y
59	SVZ	Przesuń w lewo, ustaw V	72 1 A04	$R/A/\bar{7}[R/A,0] Y:=R/R/0$	Y,V
60	SVY	Przesuń w lewo z Y, us.V	72 1 A05	$R/A/\bar{7}[R/A,Y] Y:=R/A/0$	Y,V
61	SVX, rA	Przesuń w lewo z X, us. V	72 1 A06	$R/A/\bar{7}[R/A,X] Y:=R/A/0$	Y,V
62	SRZ	Przesuń w prawo	72 1 A10	$[0,R/A/] R/A/\bar{7}Y:=R/A/15$	Y
63	SRY	Przesuń w prawo z Y	72 0 A07	$[Y,R/A/] R/A/\bar{7}Y:=R/A/15$	Y
64	SRX	Przesuń w prawo z X	72 0 A07	$[X,R/A/] R/A/\bar{7}Y:=R/A/15$	Y
65	SHC, rA, T	Przesuń cyklicznie	72 T A2T	$R/A/\bar{7}R/A/ T - \text{liczba przesunięć}$	-
66	LBn, rA, rC, rB	Umieść bajt (znak)	55 D ABC	$R/A/_{8+15} := S/N; NB/$	-
67	RBn, rA, rC, rB	Pamiętaj bajt	56 D ABC	$S/N; NB/ := R/A/_{8+15}$	-
68	ZLB, rA	Zeruj bajt lewy	72 0 A01	$R/A/_{0+7} := 0$	-
69	ZRB, rA	Zeruj bajt prawy	72 1 A01	$R/A/_{8+15} := 0$	-
70	CBn, rA, rC, rB	Porównaj bajt	57 D ABC	$R/A/_{8+15} \geq S/N; NB/$	L,E,G
71	UJn, rC, rB	Skocz	74 D 0BC	$IC:=N$	-
72	UJS	Skocz względnie	70 T 0TT	$IC:=IC+T$	-
73	LJn, rC, rB	Skocz ze śladem	74 D 7BC	$S/N/ := IC IC:=N+1$	-
74	RJn, rA, rC, rB	Skocz ze śladem w rej.	26 D ABC	$R/A/ := IC IC:=N$	-
75	JNn, rC, rB	Skocz gdy \bar{E}	74 D 6BC	Jeśli $E=0$ to $IC:=N$	-
76	JLn, rC, rB	Skocz przy L	74 D IBC	Jeśli $L=1$ to $IC:=N$	-
77	JLS, T	Skocz względnie przy L	70 T 1TT	Jeśli $L=1$ to $IC:=IC+T$	-
78	JEn, rC, rB	Skocz przy E	74 D 2BC	Jeśli $E=1$ to $IC:=N$	-
79	JES, T	Skocz względnie przy E	70 T 2TT	Jeśli $E=1$ to $IC:=IC+T$	-
80	JGn, rC, rB	Skocz przy G	74 D 3BC	Jeśli $G=1$ to $IC:=N$	-
81	JGS, T	Skocz względnie przy G	70 T 3TT	Jeśli $G=1$ to $IC:=IC+T$	-
82	JZn, rC, rB	Skocz przy Z	74 D 4BC	Jeśli $Z=1$ to $IC:=N$	-
83	JMn, rC, rB	Skocz przy M	74 D 5BC	Jeśli $M=1$ to $IC:=N$	-
84	JVS, T	Skocz względnie przy V	70 T 4TT	Jeśli $V=1$ to $IC:=IC+T V:=0$	V
85	JCS, T	Skocz względnie przy C	70 T 7TT	Jeśli $C=1$ to $IC:=IC+T$	-
86	JYS, T	Skocz względnie przy Y	70 T 6TT	Jeśli $Y=1$ to $IC:=IC+T$	-
87	JXS, T	Skocz względnie przy X	70 T 5TT	Jeśli $X=1$ to $IC:=IC+T$	-
88	TRB, rA, T	Dodaj arg. krótki przesk.	61 T ATT	$R/A/ := R/A/+T$ jeśli $R/A/=0$ to $P:=1$	P
89	IRB, rA, T	Dodaj 1, skocz względnie	62 T ATT	$R/A/ := R/A/+1$ jeśli $R/A/=0$ to $IC:=IC+T$	-
90	DRB, rA, T	Odejmij 1, skocz względnie	63 T ATT	$R/A/ := R/A/-1$ jeśli $R/A/=0$ to $IC:=IC+T$	-
91	BBn, rA, rC, rB	Porównaj logicznie iloczyn	30 D ABC	Jeśli $R/A/\wedge N \equiv N$ to $P:=1$	P
92	BMn, rA, rC, rB	Por.log.iloczyn w pamięci	31 D ABC	Jeśli $S/N; NB/\wedge R/A/ \equiv R/A/$ to $P:=1$	P
93	BŠn, rA, rC, rB	Por.log. przez maskę	32 D ABC	Jeśli $R/A/\wedge R7 \equiv N \wedge R7$ to $P:=1$	P
94	BCn, rA, rC, rB	Por. log. iloczyn	33 D ABC	Jeśli $R/A/\wedge N \equiv N$ to $P:=1$	P
95	BNn, rA, rC, rB	Przeskocz gdy iloczyn=0	34 D ABC	Jeśli $R/A/\wedge N \equiv 0$ to $P:=1$	P
96	BRC, b	Badaj prawy bajt R0	71 1 ⁰ /b bb	Jeśli $R0_{8+15} \wedge b \neq b$ to $P:=1$	P
97	BLC, b	Badaj lewy bajt R0	71 0 ⁰ /b bb	Jeśli $R0_{0+7} \wedge b \neq b$ to $P:=1$	P
98	IBn, rC, rB	Następnik pamięci	77 D 7BC	$S/N/ := S/N/+1$ jeśli $S/N/=0$ to $P:=1$	P
99	MDn, rC, rB	Modyfikuj	77 D 5BC	$MOD:=N$	-
100	NOP	Nic nie rób	70 0 000	traktowany przez assembler jako UJS, 0	-
101	RKY, rA	Czytaj klucze	72 1 A00	$R/A/ := \text{klucze KB pulpitu technicznego}$	-
102	RIC, rA	Czytaj licznik rozkazów	72 0 A00	$R/A/ := IC$	-
103	ISn, rA, rC, rB	Ustaw semafor	27 D ABC	Jeśli $S/N; NB/\wedge R/A/ \equiv R/A/$ to $P:=1$, jeśli $S/N; NB/\wedge R/A/ \equiv R/A/$ to $S/N; NB/ := S/N; NB/\vee R/A/$	P
104	EXL, b	Ekstrakod b - numeru ekstrakodu	71 0 ¹ /b bb	$[S/S/97; 0; \dots; S/S/97; 0/+3; 0] :=$ $= [IC, R0, SR, b] IC := S/96/ R0 := 0 SR_{9,10} := 0$ $S/97; 0 := S/97; 0/+4$	-
105	RPC, rA	Prześlij rejestr R0	72 1 A11	$R/A/ := R0$	-
106	LPC, rA	Umieść w rejestrze R0	72 1 A11	$R0 := R/A/$	-
107	MBn, rA, rC, rB	Umieść w SR	77 D 0BC	$SR_{10+15} := S/N/_{10+15}$	-
108	IMn, rA, rC, rB	Umieść w RM	77 D 1BC	$SR_{0+9} := S/N/_{0+9}$	-
109	KIn, rA, rC, rB	Pamiętaj RZ	77 D 2BC	$S/N/ := RZ_{0+11, 28+31}$	-
110	FIN, rA, rC, rB	Umieść w RZ	77 D 3BC	$RZ_{0+11, 28+31} := S/N/$	-
111	SPn, rA, rC, rB	Powrót do programu	77 D 4BC	$IC := S/N; NB/ R0 := S/N+1; NB/ SR := S/N+2; NB/$	-

Lp.	Zapis symbol	Nazwa rozkazu	Kod D ABC	Treść rozkazu	ust.wsk.
112	HLT	Czekaj	73 - 0--		-
113	MCL	Zeruj	73 - 1--	RZ:=0 SR:=0 R0:=0 zeruj kanały, urządzenia, rejestry podziału pro.	-
114	SIU	Ustaw przerwanie starsze	73 - 202	RZ ₃₀ :=1	-
115	SIL	Ustaw przerwanie młodsze	73 - 201	RZ ₃₁ :=1	-
116	CIT	Zeruj przerw. programowe	73 - 200	RZ _{30÷31} :=00	-
117	SIT	Ustaw przerw. programowe	73 - 203	RZ _{30÷31} :=11	-
118	GIU	Wyślij przerwanie typu U	73 0 3--	RZ ₃ /drug. procesora/ :=1] jeśli odpowiedź OK	P
119	GIL	Wyślij przerwanie typu L	73 1 3--	RZ ₂₉ /dr.ptocedora/ :=1] pozytywna to P:=1	P
120	LIP	Powrót z przerwania	73 - 4--	[IC,R0,SR]:=[S/S/97; 0/-4; 0/S/S/97; 0/-3; 0/S/S/97; 0/-2; 0/] S/97; 0/:=S/97; 4/-4	-
121	OUn, rA, rC, rB	Wyjście	35 D ABC	a/ rozkaz do kanału lub jednostki steruj. N ₁₅ =0 N _{0÷7} - kod operacji N _{8÷10} - nr j.s. N _{11÷14} - numer kanału R/A/ - informacja do kanału lub j.s. b/ rozkaz do pamięci /podział programowy/ N ₁₅ =1 N _{11÷14} - numer fizyczny modułu pamięci N _{8÷10} - adres fizyczny segmentu 4k R/A/ _{12÷15} - numer logiczny bloku /nadawany/ R/A/ ₀₋₃ - adres logiczny seg.4k /nadawany/ IC:= { S/IC/ - jeśli brak odpowiedzi S/IC+1/ - jeśli odp. negatywna EN S/IC+2/ - jeśli odp. pozytywna OK S/IC+3/ - jeśli odp. syg. błąd PE a/ i b/	-
122	INn, rA, rC, rB	Wejście	36 D ABC	N i IC - jak dla rozkazu OÜ a/ R/A/:= informacja z kanału lub j.s.	-

Przerwanie: jeśli RZ_i =1, gdzie i ∈ [0,31] oraz RM_j =1, gdzie j ∈ [0,9] oznacza numer pozycji RM odpowiadającej przerwaniu o numerze i, to:

[S/S/97; 0,S/S/97; 0/+1; 0/,...,S/S/97; 0/+3; 0/]:=[IC,R0,SR, specyfikacja przerwania]

R0:=0, RM_{j;÷9}:=0 dla i ∈ [1,31], RM_{0÷9} :=0 dla i=0, RZ_i :=0, IC:=S/64+1; 0/, S/97; 0/:=S/97; 0/+4

LISTA ROZKAZÓW MERA 400 Operacje WE - WY

Nazwa modułu	Treść operacji (njs - nr jedn. sterującej, nk - nr kanału)	Typ rozk	N - argument efektywny rozkazu				R/A/ - argument (rejestr o nr A)		Odpowiedź na rozkaz
			01234567	890	11111234	15	01234567	11111189012345	
Kanał znakowy	Sprawdź istnienie kanału	OU	0000	---	nk	0	-	-	OK
	Blokuj przerwania do procesora o nr PN	OU	00001	---	nk	0	-	-	OK
	Blokuj przerwania do procesora o nr PN	OU	00010	---	nk	0	-	-	OK
	Przydziel urząd. procesorowi o nr PN	OU	00011	---	njs nk	0	-	-	OK
	Sprawdź istnienie kanału	IN	00000	---	nk	0	-	-	OK
	Podaj stan rejestru alokacji	IN	00011	---	nk	0	stan all.	-	OK
	Podaj specyfikację przerwania	IN	00001	---	nk	0	specyfikacja	-	OK
CT2100	Sprawdź istnienie urządzenia	IN	10000	---	njs nk	0	-	-	OK
	Czytaj	IN	10100	---	njs nk	0	-	inf z js	OK,EN,OK+PE
	Zeruj	OU	10000	---	njs nk	0	-	-	OK
	Odłącz	OU	10100	---	njs nk	0	-	-	OK
	Badaj parzystość znaku	OU	11100	---	njs nk	0	-	-	OK
	Badaj nieparzystość znaku	OU	11101	---	njs nk	0	-	-	OK
	Nie badaj parzystości znaku	OU	11111	---	njs nk	0	-	-	OK
Fomijaj znaki 8x"0" i 7x"1"	OU	11100	---	njs nk	0	-	-	OK	
	Specyfikacje przerwań						00000001	njs-----	
		Ponowna gotowość					00000010	njs-----	
	zgłaszanych przez j,s,	Awaria					00000100	njs-----	
		Ostrzeżenie					00000000	njs-----	
Dt105S	Sprawdź istnienie urządzenia	IN	10000	---	njs nk	0	-	-	OK
	Zeruj	OU	10000	---	njs nk	0	-	-	OK
	Odłącz	OU	10100	---	njs nk	0	-	-	OK
	Pisz	OU	11000	---	njs nk	0	-	inf do js	OK,EN
	Dobijaj bit parzystości	OU	11101	---	njs nk	0	-	-	OK
	Dobijaj bit nieparzystości	OU	11110	---	njs nk	0	-	-	OK
	Wyłącz parzystość	OU	11111	---	njs nk	0	-	-	OK
	Specyfikacje przerwań						00000001	njs-----	
		Ponowna gotowość					00000010	njs-----	
		Awaria					00000000	njs-----	
		Przerwanie nieaktualne							
Drukarka DZM180	Sprawdź istnienie urządzenia	IN	10000	---	njs nk	0	-	-	OK
	Zeruj	OU	10000	---	njs nk	0	-	-	OK
	Odłącz	OU	10100	---	njs nk	0	-	-	OK
	Pisz	OU	11000	---	njs nk	0	-	inf do js	OK,EN
	Specyfikacje przerwań						00000001	njs-----	
		Ponowna gotowość					00000010	njs-----	
		Awaria					00000000	njs-----	
		Przerwanie nieaktualne							
VT340 DD390 DZM180 -KSR	Sprawdź istnienie urządzenia	IN	10000	---	njs nk	0	-	-	OK
	Czytaj	IN	10100	---	njs nk	0	-	inf z js	OK,EN,OK+PE
	Zeruj	OU	10000	---	njs nk	0	-	-	OK
	Odłącz	OU	10100	---	njs nk	0	-	-	OK,EN
Modem	Pisz	OU	11000	---	nk	0	-	inf do js	OK,EN
	Specyfikacje wysyłanych przerwań						00000001	njs-----	
		Ponowna gotowość					00000101	njs-----	
		Nienadążanie transmisji					00000011	njs-----	
	Zgłoszenie operatora					00000000	njs-----		
	Przerwanie nieaktualne								

LISTA ROZKAZÓW MERA 400 Operacje WE - WY

Nazwa modułu	Treść operacji (njs - nr jedn.sterującej, nk - nr kanału)		Typ rozk	N - argument efektywny rozkazu				R/A/ - argument (rejestr o nr A)		Odpowiedź na rozkaz
				01234567	1 890	1111 1234	1 5	01234567	111111 89012345	
Kanał pamięciowy	Sprawdź istnienie kanału		OU	00000	-	nk	0	-	-	OK
	Blokuj przerwania do procesora o nr PN		OU	00001	-	nk	0	-	-	OK
	Blokuj przerwania do procesora o nr PN		OU	00010	-	nk	0	-	-	OK
	Przydziel urządzenie procesorowi o nr PN		OU	00011	njs	nk	0	-	-	OK
	Sprawdź istnienie kanału		IN	00000	njs	nk	0	-	-	OK
	Podaj stan kanału		IN	00010	-	nk	0	stan kanału		OK
	Podaj stan rejestru alokacji		IN	00011	-	nk	0	stan all.		OK
	Podaj specyfikację przerwania		IN	00001	-	nk	0	specyfikacja		OK
Pamięć dyskowa	Sprawdź istnienie urządzenia		IN	10000	njs	nk	0	-	-	OK,EN
	Testowanie techniczne urządzenia		IN	10010	njs	nk	0	-	-	OK
	Pobranie statusu ścieżki		IN	01000	njs	nk	0	status ścieżki		OK,EN
	Zerowanie jednostki sterującej		OU	10000	njs	nk	0	-	-	OK
	Testowanie techniczne		OU	10010	njs	nk	0	-	-	OK
	Odłączenie urządzenia		OU	10100	njs	nk	0	-	-	OK,EN
	Wycofanie głowic na cylinder zerowy		OU	01000	njs	nk	0	-	-	OK,EN
	Ustaw głowicę wg nr cylindra		OU	11100	njs	nk	0	-	nr cyl.	OK,EN
	Transmisja wg poprzednich adresów		OU	11000	njs	nk	0	adr. pola ster.		OK,EN
	Transmisja wg nowych adresów		OU	11010	njs	nk	0	adr. pola ster.		OK,EN
MERA 9425	Specyfikacje wysyłanych przerw	Przerwanie nieaktualne	E0000000	njs	-----					
		Nienadążanie transmisji	E0000001	njs	-----					
		Brak pamięci operacyjnej	E0000010	njs	-----					
		Błąd porównania	E0000011	njs	-----					
		Błąd parzystości	E0000100	njs	-----					
		Wykonanie operacji zerowania	E0000101	njs	-----					
		Wykonanie operacji testowania	E0000110	njs	-----					
		Wykonanie operacji wycofania głowic na cylinder zer.	E0001000	njs	-----					
		Wykonanie operacji ustawiania głowic wg nr cylindra	E0001001	njs	-----					
		Alarm	E0001010	njs	-----					
	Specyfikacja przerw	Blokada	E0001011	njs	-----					
		Brak urządzenia	E0001100	njs	-----					
		Błąd ustawienia	E0001101	njs	-----					
		Błąd w polu sterującym	E0001110	njs	-----					
		Niezakończona faza sprawdzania pola adresowego	E0010001	njs	-----					
		Niezakończona faza sprawdzania pola danych	E0010010	njs	-----					
		Błąd kontroli adresów	E0010011	njs	-----					
		Błąd kontroli danych	E0010100	njs	-----					
		Niezgodność nr cylindra w polu sterującym i adresowym	E0010101	njs	-----					
		Niezgodność nr głowicy w polu sterującym i adresowym	E0010110	njs	-----					
		Niezgodność nr sektora w polu sterującym i adresowym	E0010111	njs	-----					
		Niezgodność klucza w polu sterującym i adresowym	E0011000	njs	-----					
		Blokada zapisu	E0011001	njs	-----					
Blokada sektora	E0011010	njs	-----							
Koniec cylindra	E0011011	njs	-----							
Znacznik końca	E0011100	njs	-----							
Prawidłowy koniec transmisji	E0011101	njs	-----							

LISTA ROZKAZÓW MERA-400 Operacje WE-WY

Nazwa modułu	Treść operacji (njs -- nr jedn. ster., nk -- nr kanału)	Typ rozk.	Nr - argument efektywny rozkazu					R/A/ - argument (rejestr o nr A)		Odpowiedź na rozkaz
			1	5	6	7	8	9		
1	2	3	01234567	890	1111 1234	5	01234567	11111 89012345	10	
Pamięć kasetowa PK-1	Sprawdź istnienie urządzenia	IN	10000---	njs	nk	0	—	—	OK	
	Podaj specyfikację przerwania	IN	00001---	njs	nk	0	specyfikacja	—	OK	
	Czytaj	IN	10100---	njs	nk	0	inf.z js	—	OK, EN	
	Zeruj	OU	10000---	njs	nk	0	—	—	OK	
	Odtącz	OU	10100---	njs	nk	0	—	—	OK, EN	
	Pisz	OU	11000---	njs	nk	0	—	—	OK, EN	
	Steruj - Ładuj	OU	11100---	njs	nk	0	inf.do js.	—	OK, EN	
	Steruj - Pisz	OU	11110---	njs	nk	0	—	—	OK, EN	
	Specyfikacje przerwań									
		Nieaktualne								
	Ponowna gotowość									
	Ponowna gotowość z wykryciem "Control Block"									
	Awaria - kaseca niezaladowana, awaria zasilania									
	Awaria zabronienia zapisu									
	Awaria - błąd kontroli cyklicznej CRC									
	Awaria - błąd formalny lub brak informacji									
	Ostrzeżenie - koniec nośnika									
Pamięć na dyskach elastycznych SP45DE	Sprawdź istnienie urządzenia	IN	1000---	njs	nk	0	—	—	OK	
	Pisz	OU	1100---	njs	nk	0	inf. z js	—	OK, EN	
	Czytaj	IN	1010---	njs	nk	0	inf. do js	—	OK, EN	
	Zeruj	OU	1000---	njs	nk	0	—	—	OK	
	Odtącz	OU	1010---	njs	nk	0	—	—	OK, EN	
	Steruj I (zapisz bez kontroli)	OU	1110---	njs	nk	0	inf. z js	—	OK, EN	
	Steruj II (zapisz z kontrolą)	OU	1111---	njs	nk	0	inf. z js	—	OK, EN	
	Steruj III (zapisz sektor anulowany)	OU	11111---	njs	nk	0	inf. z js	—	OK, EN	
	Steruj IV (czytaj od zadanego adresu)	OU	11101---	njs	nk	0	inf. z js	—	OK, EN	
	Specyfikacja przerwań									
	Ponowna gotowość									
	Awaria									
	Błąd położenia sektor anulowany									
	Sektor anulowany									
	Koniec nośnika									
	Przerwanie nieaktualne									

LISTA ROZKAZÓW MERA 400 Operacje WE – WY

Nazwa modułu	Treść operacji (njs – nr jedn. sterującej, nk – nr kanału)	Typ rozk	N – argument efektywny rozkazu				R/A – argument (rejestr o nr A)		Odpowiedź
			01234567	1890	11111234	15	01234567	11111189012345	
Pamięć dyskowa MERA – 9425	1* E-bit specyfikacji przerwania; przyjmuje wartość "1", gdy w czasie od poprzedniego wysłania specyfikacji do obecnego, na dowolną operację do urządzenia została udzielona odpowiedź EN.								
	0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15								
	Do użytku systemu operacyjnego		Ilość słów pola sterującego			NR JC	Numer bloku pamięci operacyjnego		
	POR		PISZ	ADR					
			CZYT	DANE					
	Ilość słów transmisji								
	IBZ	IBU	IK	NZK	Nr cylindra				
				NT	NG	Nr sektora			
	Klucz								
	Adres początku transmisji								

POR – czytanie z porównaniem
 IBZ – ignoruj blokadę zapisu
 IBU – ignoruj blokadę uszkodzeń
 IK – ignoruj klucz
 NZK – nie reaguj na znacznik końca
 ING – numer głowicy
 NT – numer talerza

LISTA ROZKAZÓW MERA 400 Operacje WE – WY

Nazwa modułu	Treść operacji WE – WY (ap – adres pakietu PI, ak – adres kasety PI, nk – nr kanału)	Typ rozk	N – argument efektywny rozkazu				R/A – argument (rejestr o nr A)		Odpowiedź na rozkaz		
			012	3456	7890	111112345	01234567	11111189012345			
Kanał automatyki PI (blok sprzężenia)	Zeruj kanał	OU	000	00	---	0	nk	0	-	-	OK
	Blokuj przerwania do procesora o nr PN	OU	000	01	---	0	nk	0	-	-	OK
	Blokuj przerwania do procesora o nr PN	OU	000	10	---	0	nk	0	-	-	OK
	Ustaw rejestr masek	OU	000	11	---	0	nk	0	stan rej. masek		OK
	Sprawdź istnienie bloku sprzężenia	IN	000	00	---	0	nk	0	-	-	OK,OK+PE
	Podaj specyfikację przerwania	IN	000	01	---	0	nk	0	stan zgł. z kaset		OK
	Podaj spec. przerw. o najwyższym prioryt.	IN	000	10	---	0	nk	0	*) / ap / ak / nk / 0		OK,OK+PE
Ustaw nr procesora	IN	000	11	---	0	nk	0	-	-	OK	
Zestaw PI	Funkcja dodatkowa	OU	100	ap	ak	0	nk	0	inf. do pakietu PI		OK,EN,OK+PE
	Pisz do pakietu pierwsze słowo	OU	-01	ap	ak	0	nk	0	"	"	OK,EN,OK+PE
	Funkcja dodatkowa	OU	-10	ap	ak	0	nk	0	"	"	OK,EN,OK+PE
	Pisz do pakietu drugie słowo	OU	-11	ap	ak	0	nk	0	"	"	OK,EN,OK+PE
	Czytaj z pakietu pierwsze słowo	IN	-01	ap	ak	0	nk	0	inf z pakietu PI		OK,EN,OK+PE
	Czytaj zgłoszenie przerwania	IN	-10	ap	ak	0	nk	0	inf z kasety PI		OK,EN,OK+PE
	Czytaj z pakietu drugie słowo	IN	-11	ap	ak	0	nk	0	inf z pakietu PI		OK,EN,OK+PE

*) kod 011 lub 001 zaszyty na stałe w kanale

PRZYKŁADY UŻYCIA ROZKAZÓW

LWn, rA, rC, rB	Umieść w rejestrze R/A/:=N
LWR, 5, 7	Umieść w rejestrze 5 zawartość rejestru 7
LWR, 5, 7, 1	Umieść w rejestrze 5 zawartość rejestru 7 zmodyfikowanego zawartością rejestru 1
LWA, 5, 7	Umieść w rejestrze 5 zawartość komórki pamięci o adresie podanym w rejestrze 7
LWA, 5, 7, 1	Umieść w rejestrze 5 zawartość komórki pamięci o adresie podanym w rejestrze 7 zmodyfikowanym zawartością rejestru 1
LWD, 5, 7	Umieść w rejestrze 5 liczbę 7
LWD, 5, 7, 1	Umieść w rejestrze 5 liczbę 7 zmodyfikowaną zawartością rejestru 1
LWI, 5, 7	Umieść w rejestrze 5 zawartość komórki o adresie 7
LWI, 5, 7, 1	Umieść w rejestrze 5 zawartość komórki o adresie 7, zmodyfikowaną zawartością rejestru 1

1. SYSTEM OPERACYJNY SOM-3

1.1. Wersje systemu operacyjnego

System operacyjny minikomputera MERA 400 może pracować (w zależności od przewidywanej konfiguracji urządzeń peryferyjnych) w trzech podstawowych wersjach: CORE, BASIC, EXTENDED. Charakterystyki poszczególnych wersji mogą się zmieniać w szerokim zakresie, dopuszczając opcjonalnie (w miarę potrzeb użytkownika) dołączanie niestandardowych elementów systemu. Dodatkowo, umożliwia się użytkownikom rozszerzenie firmowego systemu o własne procedury, w celu dostosowania go do szczególnych potrzeb konkretnej instalacji.

Wersja CORE przeznaczona jest głównie dla zamkniętej klasy systemów czasu rzeczywistego (tzw. "pod klucz"), które nie wymagają zewnętrznych pamięci magnetycznych. Zakres wprowadzanych zmian w programach po dokonaniu instalacji jest ograniczony. Dopuszcza się opcjonalnie możliwość rozszerzenia systemu o przetwarzanie wsadowe i overlayowanie zadań, jeżeli przewidziano odpowiednie urządzenia we/wy i pojemność pamięci operacyjnej.

Wersja BASIC jest najszerzej stosowaną odmianą systemu operacyjnego. Współpracuje ona z zewnętrznymi pamięciami magnetycznymi. Głównie przeznaczona jest do przygotowywania programów w reżimie automatycznej pracy wsadowej. Dopuszcza instalacje dużych systemów użytkowych dzięki standardowej możliwości overlayowania, łatwego wykorzystania bibliotek oraz translatorów języków wyższego rzędu. Użytkowanie pamięci dyskowej zapewnia wygodę i łatwość pracy oraz zwiększenie możliwości systemu.

Wersja EXTENDED wymaga obecności zewnętrznych pamięci dyskowych dostarcza szerokich możliwości ingerencji użytkownika w wewnętrzną strukturę oprogramowania, elastycznego wykorzystania czasu maszynowego, pamięci operacyjnej i dyskowej. Dopuszcza tworzenie skomplikowanych nadsystemów kontrolujących i planujących przebieg prac w maszynie. Zapewnia ekonomiczne wykorzystanie zasobów systemu, w szczególności rozwiązuje kłopotliwy problem użytkownika nielicznych urządzeń we/wy. Głównie przeznaczona jest do: obsługi złożonych problemów programowania, w szczególności czasu rzeczywistego, organizacji dużych ośrodków o różnym profilu zastosowań, prób wdrożeń nowych instalacji systemu i rozwoju oprogramowania.

1.2. System w wersji "CORE"

Wersja CORE pracuje z konfiguracjami urządzeń peryferyjnych bez zewnętrznych pamięci magnetycznych i charakteryzuje się następującymi możliwościami:

- Operowanie tylko na zadaniach rezydujących. Ilość zadań ustalona w czasie generacji systemu.
- Procedura planująca systemu dzieli czas maszyny, niewykorzystany na obsługę przerw, pomiędzy poszczególne zadania zgodnie z priorytetową kolejką prac systemu.
- Każde zdarzenie systemowe powoduje przekazanie sterowania najstarszemu zadaniu. Zadania mogą powodować przekazanie sterowania najstarszemu lub następnemu zadaniu.
- Zadania mogą być wykonywane na jednym z 256 poziomów priorytetu. Każdy poziom dopuszcza istnienie własnej kolejki zadań.
- W zależności od priorytetu dokonuje się podziału zadań na systemowe (pełny dostęp do sprzętu i wewnętrznej struktury systemu), uprzywilejowane (prawo korzystania ze wszystkich ekstrakodów) i zwykłe.
- Przydzielanie zadaniom pamięci operacyjnej w czasie generacji systemu i tworzenie dowolnej konfiguracji bloków pamięci (tzw. podział programowy). Wstępnie ustalone przydziały nie podlegają żadnym zmianom.
- Ochrona pamięci w przypadku lokalizacji jednego zadania w bloku. Dopuszczalne jest istnienie wielu zadań w bloku bez ochrony. Możliwość utworzenia wspólnych dla zadań obszarów komunikacji (COMMON) w bloku pamięci.
- System zapewnia standardową obsługę przerw. Użytkownik ma możliwość dołączenia własnej obsługi przerw specjalnych (w przypadku występowania krytycznych czasów obsługi).
- Program użytkowy nie jest zaangażowany w obsługę operacji we/wy. Są one kolejgowane do autonomicznego podsystemu we/wy i obsługiwane zgodnie z priorytetem zadań.

- Standardowa obsługa przerwania sprzętowego zegara pozwala w dowolny sposób zdefiniować programowy dzielnik częstotliwości. Możliwe jest dołączanie Monitora Czasu Rzeczywistego, kontrolującego systemowy zegar i datownik dostępnym zadaniom.
- Dopuszcza się rozszerzenie zakresu możliwości Monitora Czasu Rzeczywistego o obsługę dowolnej ilości lokalnych i globalnych zegarów programowych. Zegary mogą działać jako timery (wielokrotne sygnały) lub buźniki (sygnał pojedynczy). Pełną kontrolę programową zapewniają odpowiednie ekstrakody.
- Dołączenie Monitora Przerwania Zewnętrznych, obsługującego zewnętrzne sygnały dwustanowe. Pełną kontrolę programową zapewniają odpowiednie ekstrakody.
- Zadania mogą być sterowane (startowanie, zawieszanie, wznowianie, odrzucanie, zmiana priorytetu) przez: operatora, podsystem we/wy, zegar programowy, zewnętrzne przerwania lub inne zadania.
- Każdemu zadaniu system przydziela zasoby z globalnej puli systemowej (pamięć operacyjna, urządzenia, strumienie we/wy, zegary programowe, przerwania zewnętrzne). Przydziału można dokonać w trakcie generacji systemu, przez operatora lub też dynamicznie (samo zadanie w trakcie działania).
- Zasoby dzielą się na permanentne (przydzielone na stałe) i tymczasowe (przydzielone na czas działania). Wszystkie zasoby są rejestrowane na bieżąco i podlegają zwrotowi do puli systemu, w momencie zakończenia zadania (tymczasowe) lub też są zatrzymywane do dalszego wykorzystania (permanentne).
- Ekstrakody wielodostępne mogą być używane we wszystkich zadaniach. Użytkownik ma możliwość dołączenia własnych ekstrakodów. Podstawowa lista ekstrakodów zapewnia: wykonywanie operacji we/wy, sterowanie przebiegiem zadań, obsługę sygnałów zewnętrznych, kontrolę zasobów zadania, analizę ciągów znakowych, konwersję postaci informacji i inne funkcje użytkowe.
- Specjalny podsystem we/wy zapewnia obsługę wszystkich rzeczywistych urządzeń peryferyjnych i pseudo-urządzeń. Każdy typ urządzenia posiada indywidualną procedurę obsługi, zwaną handlerem. Pojedynczy handler może obsługiwać dowolną ilość urządzeń tego samego typu. Wszystkie handlery działają pod kontrolą podsystemu we/wy.
- Operacja we/wy może wstrzymać bieg zadania lub wykonywać się współbieżnie z nim (time-sharing). Dopuszcza się równoległe prowadzenie wielu operacji we/wy (wielodostępność) w jednym zadaniu.
- Wszystkie operacje we/wy są buforowane. Podsystem we/wy gwarantuje konwersję kodów do jednolitej postaci wewnętrznej (ISO-7). Obsługa błędów operacji we/wy może być w zadaniu lub w systemie.
- Wszystkie operacje we/wy odbywają się za pośrednictwem pseudokanałów przesyłania (tzw. strumieni we/wy), które są niezależne od typu urządzenia. Interpretacja poszczególnych operacji dokonywana jest w handlerach. Dopuszcza się istnienie strumieni lokalnych (prywatnych dla zadania) i globalnych (wspólnych dla wszystkich zadań, przy narzuceniu ograniczeń dostępu).
- Handlerzy zapewniają jednolity sposób obsługi urządzeń zewnętrznych. Strumienie zadania przydzielone są do urządzeń: w czasie generacji systemu, przez operatora lub przez program użytkowy.
- System dopuszcza standardowe i niestandardowe, binarne i znakowe typy organizacji danych. Organizacja standardowa umożliwia jednolitą współpracę z każdym urządzeniem, niezależnie od jego typu. Wraz z koncepcją strumieni we/wy stanowi to całkowite uniezależnienie pisanych programów od zestawu urządzeń we/wy.
- Opcjonalne wyposażenie systemu stanowić mogą: procedura SYSLOADER (ładująca overlaye zadań z peryferyjnych urządzeń sekwencyjnych), symulator arytmometru wielokrotnej precyzji (uruchamiany standardowymi rozkazami).
- System obejmuje rezydujące zadanie komunikacji (interpreter języka operacyjnego maszyny); standardowy pakiet dyrektyw zapewnia podstawowe funkcje z zakresu kontroli zadań, urządzeń, zasobów i inne funkcje użytkowe. Możliwe jest uzupełnienie zadania komunikacji własnymi dyrektywami.
- Dołączenie zadania BACKGROUND wraz z programami zwanymi procesorami systemowymi, przeznaczonymi do przetwarzania wsadowego w zakresie: kompilacji, translacji, konsolidacji programów, organizacji źródłowych i binarnych bibliotek systemu, uruchamiania programów.

1.3. System w wersji "BASIC"

Wersja BASIC współpracuje z zewnętrznymi pamięciami magnetycznymi (Taśmy, dyski). Charakteryzuje się następującymi rozszerzeniami w stosunku do wersji CORE:

- System wyposażony jest standardowo w procedurę SYSLOADER, zapewniającą możliwość ładowania overlay'ów z urządzeń o dostępie bezpośrednim lub sekwencyjnym. Overlay'e mogą być zorganizowane w biblioteczne zbiory sekwencyjne lub słownikowe.
- Dołączenie zadania BACKGROUND umożliwia automatyczne przetwarzanie wsadowe. Konieczne jest wydzielenie przewijalnych urządzeń pamięciowych (Taśmy magnetyczne, dyski) na bibliotekę procesorów systemowych, oraz na buforowe zbiory robocze.
- Możliwe jest dołączenie nierezydującego zadania komunikacji z overlayowanym pakietem obsługi dyrektyw. Użytkownik może dołączać własne dyrektywy. Wybrane dyrektywy mogą rezydować w pamięci operacyjnej.
- Wydzielona pula pamięci dyskowej może być dynamicznie przydzielona do zadań.
- Obecność dysków umożliwia stosowanie "Spoolerów" (specjalnych zadań—symbiontów dostępnych poprzez pseudourządzenia), które pozwalają na skoordynowane wykorzystanie nielicznych, rzeczywistych urządzeń we/wy przez wiele zadań jednocześnie. Użytkownik ma możliwość dołączania własnych spoolerów.

1.4. System w wersji "EXTENDED"

Wersja EXTENDED przewiduje wydzielenie co najmniej jednej stacji dyskowej w charakterze urządzenia systemowego. Zachowując wszystkie zalety wersji CORE i BASIC, dopuszcza znaczne rozszerzenie aparatu systemowego:

- System umożliwia kontrolę zadań nierezydujących (katalogowanych na słownikowych zbiorach dyskowych) i zadań rezydujących. Ciąła zadań (programy użytkowe) katalogowane są wraz ze specyfikacjami zasobów niezbędnych do ich działania. Użytkownik może w trakcie pracy systemu dołączać nowe zadania.
- Zadania dzieli się (wg priorytetu) na cztery poziomy:
 - SYSGROUND (uprzywilejowane zadania systemowe — w systemowym bloku pamięci), FORGROUND (zadania uprzywilejowane — we wspólnym wydzielonym bloku pamięci), MIDDLEGROUND (zwykłe zadania w oddzielnych blokach) i BACKGROUND (w pełni automatyczne przetwarzanie wsadowe — w wydzielonym bloku).
- Podział pamięci na bloki dokonywany jest w czasie generacji systemu (podobnie przydział pamięci dla zadań rezydujących). Niewykorzystane obszary tworzą pulę pamięci operacyjnej, przydzielonej zadaniom nierezydującym na okres ich działania.
- Działającym zadaniom może być dynamicznie przydzielona pamięć operacyjna.
- Systemowa procedura ALLOCATOR obsługuje żądania uruchomienia zadań nierezydujących; działa z priorytetem obsługiwanego zadania. Przydziela pamięć operacyjną, ładuje zadania ze zbioru dyskowego i przydziela wyspecyfikowane zasoby z puli systemowej.
- Możliwe jest wprowadzenie zadań czasowo—rezydujących, które po załadowaniu i przydzieleniu zasobów zachowują aż do odwołania wszystkie cechy zadań rezydujących.

2. TRANSLATORY I PROCESORY SYSTEMOWE

Procesory systemowe minikomputera MERA 400 działają pod nadzorem systemu operacyjnego SOM-3. Służą do przygotowywania programów. Procesorami są kompilatory, assembly, interpretry, loadery jak również programy do obsługi źródłowych i wynikowych zbiorów informacji oraz manipulowania strukturami danych.

Procesory systemowe są wywoływane przez JOB CONTROL jako overlay e zadania przetwarzania wsadowego (zadania BACKGROUND). Przydziały strumieni oraz sterowanie kolejnymi krokami procesu przetwarzania określa się w języku JOB CONTROL a. System umożliwia wielostrumieniowe przetwarzanie wsadowe.

Minikomputer MERA 400 ma obszerne oprogramowanie, wspomagające tworzenie systemów i programów użytkowych.

Oprogramowanie zawiera:

- Fortran IV z rozszerzeniami czasu rzeczywistego,
- interakcyjny, wielostanowiskowy BASIC,
- Macroassembler,
- procesory obsługi zbiorów sekwencyjnych i o odstępie bezpośrednim,
- Konsolidator (Link Editor) łączący w jeden program moduły wynikowe, umożliwiającą tworzenie nakładek (overlay ów),
- Bibliotekarz (Cataloger) umożliwiający tworzenie zbiorów słownikowych modułów wynikowych,
- Aktualizator Biblioteki Sekwencyjnej (Library Update),
- Aktualizator Tekstowy (Source Update) pozwalający przetwarzać zbiory źródłowe,
- Debugger ułatwiający uruchomienie programów,
- Biblioteki programów inżynierskich w językach BASIC i FORTRAN.

JOB CONTROL

Procesor JOB CONTROL jest wykonywany w zadaniu BACKGROUND. JOB CONTROL współdziała z procesorami systemowymi i programami użytkownika; steruje i organizuje ich pracę na podstawie pliku dyrektyw sterujących. Plik dyrektyw sterujących jest przetwarzany warunkowo, w zależności od błędów sygnalizowanych przez procesory.

Dyrektywy sterujące mogą być wprowadzane przez użytkownika, zarówno bezpośrednio z urządzenia typu terminal, jak i innych standardowych urządzeń wejściowych systemu.

Kolejne pliki dyrektyw są wykonywane w porządku sekwencyjnym ustalonym przez operatora.

Umożliwiają one:

- przydziały logicznych strumieni i urządzeń,
- pozycjonowanie zbiorów i rekordów,
- odwijanie nośników sekwencyjnych,
- wybranie, załadowanie i wykonanie programu (zgodnie z opcjami),
- zawiadomienie operatora o żądanej akcji,
- definiowanie i wykonywanie procedur JOB CONTROL a,
- inicjowanie i zakończenie pliku prac.

2.1. Translatory

Translatory są wykonywane w zadaniu BACKGROUND. Wykorzystywane są do tworzenia programów użytkowych.

FORTRAN IV

FORTRAN IV MERA 400 jest opracowywany w oparciu o dokument międzynarodowej organizacji ISO pt. "ISO Recommendation R1539, Programming Language Fortran" (Wydanie I z lipca 72 r.).

Zrealizowano rozszerzenie FORTRANU w zakresie czasu rzeczywistego wg ISA Standard 61.1 i 61.2.

BASIC

Język Basic należy do grupy języków konwersacyjnych, ma bardzo prostą konstrukcję i jest łatwy do nauczenia. Użytkownik komunikuje się z BASIC-iem poprzez zwrotne urządzenie wejścia/wyjścia (dalekopis, monitor ekranowy, itp.), służące zarówno do wprowadzenia i poprawienia programu, jak i do wprowadzenia danych i wyprowadzenia wyników.

Język BASIC, podobnie jak inne języki konwersacyjne, nadaje się przede wszystkim do szybkiego rozwiązywania problemów numerycznych.

Translator BASIC MERA 400 jest wielodostępny; umożliwia jednoczesną pracę wielu użytkowników. Wersja dyskowa pozwala korzystać z dyskowych bibliotek programów oraz dyskowych zbiorów danych.

MACROASSEMBLER

Umożliwia generowanie gniazdowych Macro odwołań, rekursywne odwołanie do Macro, translację warunkową, rozgałęzienia podczas translacji.

Macroassembler jest procesorem dwuprzebiegowym, generującym moduły wynikowe w postaci przemieszczalnej jak i nieprzemieszczalnej. W wersji dyskowej Macroassembler zajmuje 8 k słów pamięci operacyjnej. Akceptuje następujące formaty stałych:

- całkowite pojedynczej precyzji (16 bitów),
- całkowite podwójnej precyzji (32 bity)
- zmiennoprzecinkowe pojedynczej precyzji (48 bitów),
- heksadecymalne,
- łańcuchy znakowe,
- adresowe,
- łańcuchy znakowe skompresowane (CAN—kod).

Język zawiera dyrektywy umożliwiające definiowanie macro-prototypów, translację warunkową, definicję symboli, przetwarzanie etykiet, tworzenie bloków COMMON do komunikacji między programami i podprogramami pisanymi w językach FORTRAN IV i MACROASSEMBLER.

2.2. Procesory systemowe

Procesory działają w zadaniu wsadowym. Dostarczają możliwości przetwarzania programów źródłowych, modułów wynikowych i ładowalnych, tworzenia i zarządzania bibliotekami.

Aktualizator tekstowy (Source Update)

Pozwala manipulować informacjami tekstowymi na zbiorach sekwencyjnych. Sekwencyjne przetwarzanie zbiorów źródłowych odbywa się funkcjami wymiany, dodawania, usuwania, kopiowania i listowania.

Zbiory źródłowe mogą być na dowolnych nośnikach standardowych, jak: taśma papierowa, karty, taśma magnetyczna lub dysk. Przygotowany zbiór źródłowy może być dalej przetwarzany przez translatory.

Aktualizator biblioteki sekwencyjnej (Library Update)

Aktualizator biblioteki sekwencyjnej pozwala na manipulację modułami wynikowymi i ładowalnymi na dowolnych standardowych nośnikach informacji.

Dyrektywy aktualizujące, to: dodawanie, usuwanie, listowanie, kompresowanie i wyszukiwanie binarnych rekordów.

Konsolidator (Link Editor)

Konsolidator konstruuje moduły ładowalne przez łączenie modułów wynikowych, otrzymanych niezależnie z kompilatorów lub assemblera. Umożliwia tworzenie Globalnych Obszarów Wspólnych (COMMON) i dołączenie globalnych podprogramów. Konsolidator przygotowuje wielopoziomowe, rozgałęzione struktury programów. Po poprawnym skonsolidowaniu można uzyskać wydruk mapy pamięci z nazwami, adresami startowymi i końcowymi każdego połączanego modułu. Również generowane są odpowiednie informacje o błędach i brakujących modułach.

Bibliotekarz (Cataloger)

Bibliotekarz jest procesorem zarządzającym modułami ładowalnymi użytkownika i systemu. Główną funkcją jest katalogowanie połączonych modułów wynikowych lub ładowalnych na uprzednio przydzielonych, słownikowych sekcjach dyskowych.

Zarządza słownikami wszystkich katalogowych modułów na każdym zbiorze. Dodatkowe funkcje, to: tworzenie słowników, listowanie, usuwanie, kompresje.

Program wspomagający uruchamianie (Debugger)

Debugger dostarcza łatwiej możliwości obsługi programów uruchamianych. Jest sterowany dyrektywami podobnymi do dyrektyw JOB CONTROLLER'a.

Wszystkie dyrektywy są wprowadzane podczas wykonywania programu i wywołują takie funkcje, jak: wydruki wskazanych miejsc pamięci i rejestrów oraz zastępowaniu tych wartości przez inne.

Ponadto Debugger umożliwia:

- śledzenie z wydrukiem wybranych fragmentów programu,
- śledzenie tylko w obszarach zadeklarowanych,
- zatrzymywanie wykonywania we wskazanych miejscach programu,
- wznowienie wykonywania programu.

5. ROZWIĄZANIA KONSTRUKCYJNE

5.1. Konstrukcja mechaniczna

Standardowa szafa zestawu stanowi konstrukcję metalową, zgrzewaną typu szufladowego z odejmowanymi osłonami bocznymi. Wysokość użytkowa szafy wynosi 25U, gdzie $U = 44,45$ mm. Szuflada pojedyncza posiada wysokość 5U, zaś szuflada modułu jednostki centralnej – złożona z dwóch trwale skręconych szuflad pojedynczych – 9U. Wysokość szuflady modułu zasilania MZS-400 wynosi 3U. Maksymalna pojemność szafy wynosi więc: 2 szuflady pojedyncze (uniwersalne), jedna szuflada podwójna (jednostki centralnej) oraz dwie szuflady modułów MZS-400.

Prowadnice typu wysuwanego umożliwiają wysuwanie z szafy szuflad zawierających moduły logiczne. Po wysunięciu szuflady zapewniony jest łatwy dostęp do znajdujących się wewnątrz pakietów. Pakiety wsuwane są w płaszczyźnie poziomej po prowadnicy zamontowanych wewnątrz szuflad. Wysuw pakietów odbywa się równoległe do ściany czołowej szafy.

Z tyłu każdej szuflady, z wyjątkiem szuflady modułu MZS-400, znajduje się standardowy zasilacz ZLI-SM3 umieszczony na prowadnicach umożliwiających łatwe wyjęcie zasilacza.

Szuflady modułów MZS-400 umieszczone są na prowadnicach ślizgowych.

Z tyłu szafy znajdują się uchwyty przeznaczone do zamocowania kabli. Szafa nie posiada tylnej osłony w celu umożliwienia przepływu powietrza.

5.2. Technika realizacji modułów logicznych

Wszystkie moduły logiczne systemu zrealizowano przy użyciu układów logicznych typu TTL małej i średniej skali integracji. Mikroukłady montowane są na dwustronnie drukowanych pakietach typu SM2 (lub SM22) o wymiarach 295x300 mm i SM11 (lub SM12) o wymiarach 140x300 mm. Pakiety typu SM11 posiadają jedno zaś pakietu typu SM2 dwa złącza pośrednie 84-stykowe typu 831-084-01-1-2-000-1. Pakiety typu SM12 zaopatrzone są w jeden, a pakietu typu SM22 w dwa dodatkowe tylne grzebienie przeznaczone do złącz pośrednich 2x24 stykowych. Grzebienie te umożliwiają bezpośrednie dołączanie do pakietów kabli urządzeń peryferyjnych, testujących i kabli interfejsu. Przedstawione rozwiązanie zwiększa niezawodność systemu oraz pozwala na jego łatwą rekonfigurację lub rozbudowę bez konieczności dokonywania jakichkolwiek przeróbek mechanicznych i montażowych.

Połączenia między pakietami danego modułu logicznego w większości zrealizowano przy użyciu techniki miniowijania.

Platery zestawu wykonane są z dwustronnie metalizowanego laminatu przy czym strona miniowijania stanowi płaszczyznę masy. Drugą stronę platerów wykorzystano została do rozprowadzenia, przy pomocy połączeń drukowanych, napięć zasilających i pewnych sygnałów logicznych.

W pewnych przypadkach np. dla interfejsu znakowego, tj. łączącego kanał znakowy z jednostkami sterującymi połączenia owijane i drukowane wykonane są standardowo. Umożliwia to przyporządkowanie urządzeniom zewnętrznym dowolnych numerów fizycznych z zakresu $0 \div 7$, przez zwykłą zmianę miejsc pakietów jednostek sterujących.

Analogiczne rozwiązania zastosowane w przypadku modułów pamięci operacyjnej pozwala na zwiększenie pojemności pamięci z 8 k słów do 32 k słów (z przyrostem 8 k słów) bez konieczności dokonywania dodatkowych połączeń.

5.3. Zasilanie zestawu

Zasilanie zestawu rozwiązano w sposób modułowy, tzn. każda szuflada posiada własny, standardowy zasilacz ZLI-400 dołączony wyłącznie do plateru danej szuflady.

Kable sieciowe wszystkich zasilaczy znajdujących się w szafie dołączone są do zespołu gniazd położonego na tylnej płycie modułu zasilania MZS-400. Do gniazd tego modułu dołączone są również kable sieciowe urządzeń peryferyjnych związanych z daną szafą.

Moduł MZS-400, po załączeniu do sieci, podaje napięcie sieciowe na tylne gniazda sieciowe w określonej kolejności. Podstawowe zalety takiego rozwiązania to:

- wyeliminowanie udarowego obciążenia sieci w chwili włączania zestawu;
- wyeliminowanie konieczności włączania każdej szuflady oddzielnie;

– możliwość automatycznego włączania poszczególnych modułów w sekwencji pożądanej ze względów systemowych.

Moduł zasilania MZS-400 dołączony jest do sieci zdalnie, poprzez stacyjkę znajdującą się na pulpicie operatora (w szufladzie jednostki centralnej). Dodatkowo istnieje możliwość lokalnego włączenia modułu zasilania przy użyciu klucza na płycie czołowej MZS-400.

Dla ułatwienia czynności serwisowych zasilacze ZLI-400 zaopatrzone są w przełączniki pozwalające na odłączenie danego zasilacza od sieci. Umożliwia to wyłączenie pożądanej szuflady bez konieczności wyłączenia całego zestawu.

system MERA 400

DOKUMENTACJA TECHNICZNO RUCHOWA
TOM I

część IV
OGÓLNA INSTRUKCJA EKSPLOATACJI
MINIKOMPUTERA MERA - 400

Nowości w opisie są zaznaczane na marginesie. Znak przy numerze strony dotyczy numeracji stron.

Z A P I S Z M I A N

Z M I A N A	O P I S
"A" 22-09-78 "B" 18-10-79	Oryginał Karta zmian nr 486/79. Zmiana na str. iii

23-004040-01-2

Uwagi kierować do : Centrum Naukowo Produkcyjne Technik Komputerowych i Pomiarów
im.J.Krasickiego , ul. Łopuszańska 117/123 02-232 Warszawa

Producent zastrzega sobie prawo wprowadzania zmian w dokumentacji.

SPIS TREŚCI

	Str.
1. INSTALACJA MINIKOMPUTERA	1-1
2. PRZYSPÓBNIENIE MINIKOMPUTERA DO PRACY	1-1
3. WYŁĄCZENIE MINIKOMPUTERA	1-1
4. KONSERWACJA MINIKOMPUTERA	1-1
5. PRZYKŁADY WYKORZYSTANIA PULPITU TECHNICZNEGO	1-1

1. INSTALACJA MINIKOMPUTERA MERA-400

- Rozstawić szafy i urządzenia zewnętrzne wchodzące w skład instalowanego zestawu zgodnie z rysunkiem rozmieszczenia przestrzennego, podanym w liście kompletności 20-004181-01-1.
- Rozmieścić w szafach i szufladach oraz połączyć między sobą moduły konstrukcyjne wchodzące w skład instalowanego zestawu zgodnie z rysunkiem połączeń modułów, podanym w liście kompletności zestawu 20-004181-01-0.
Połączenia kablowe należy wykonać z uwzględnieniem uwagi na stronie 1-4 pkt.6 listy kompletności.
- Przewód uziemiający 11-004596-05-9 modułu MPS-400 łączyć wyłącznie w przypadku sieci niezero-
wanej.

2. PRZYSPOSOBIENIE MINIKOMPUTERA DO PRACY

- Ustawić kluczyk w stacyjce pulpitu technicznego modułu MJC-400 w położenie ON.
- Ustawić gotowość urządzeń zewnętrznych zgodnie z instrukcjami eksploatacji poszczególnych urządzeń.
- Nacisnąć klucz CLEAR na pulpicie technicznym MJC-400
- Wczytać system operacyjny do pamięci operacyjnej minikomputera zgodnie z opisem eksploatacyjnym systemu SOM-3 zamieszczonym w tomie III niniejszej DTR.

3. WYŁĄCZENIE MINIKOMPUTERA

- Po zakończeniu uruchamiania lub wykonywania programów użytkowych ustawić kluczyk stacyjki modułu MJC-400 w położenie OFF.

4. KONSERWACJA MINIKOMPUTERA

- Do codziennych czynności konserwatorskich należy oczyszczenie minikomputera z kurzu i usunięcie "confetti" ze zbiornika znajdującego się przy dziurkarce oraz testowanie minikomputera przy pomocy testów przypisanych danemu zestawowi zgodnie z listą kompletności 20-004181-01-0 według opisów podanych w II tomie niniejszej DTR.
- Wszystkie urządzenia zewnętrzne wchodzące w skład zestawu winny być konserwowane zgodnie z ich instrukcjami obsługi i eksploatacji.
- We wszystkich silnikach wentylatorów zasilaczy ZLI-400, wchodzących w moduły szufladowe MERA-400 należy uzupełniać zapas oleju w pierścieniach filcowych. Należy używać wyłącznie oleju "MWP". Czynność tą należy dokonywać corocznie lub po każdych 2500 godzinach eksploatacji zestawu.

5. PRZYKŁADY WYKORZYSTANIA PULPITU TECHNICZNEGO

5.1. Wpisywanie informacji do rejestrów

1. Wybrać rejestr przy pomocy przełącznika obrotowego.

2. Ustawić na kluczach KB żadaną informację.
3. Nacisnąć przełącznik LOAD

— na lampkach $0 \div 15$ wyświetlana jest informacja ustawiona na kluczach KB.

5.2. Wpisywanie informacji do pamięci operacyjnej

1. Na kluczach KB ustawić adres komórki pamięci (dostęp z klawiatury istnieje tylko do pamięci w bloku systemowym tzn. adres nie powinien przekraczać 8 k słów maszynowych).
2. Przełącznikiem obrotowym wybrać rejestr AR.
3. Nacisnąć przełącznik LOAD — na lampkach $0 \div 15$ wyświetlony jest adres komórki pamięci operacyjnej ustawiony na kluczach KB.
4. Przełącznikiem obrotowym wybrać rejestr, z którego ma być wpisana informacja lub wybrać pozycję KB, jeśli informacja ma być pobrana z kluczy.
5. Nacisnąć przełącznik STORE
 - informacja z wybranego rejestru lub ustawiona na kluczach KB zostaje zapamiętana w komórce pamięci o adresie wskazanym zawartością rejestru AR;
 - zawartość rejestru AR (wyświetlona na lampkach $0 \div 15$ po wybraniu rejestru AR przełącznikiem) zwiększona jest o 1.

5.3. Odczytanie informacji z pamięci operacyjnej

1. Na kluczach KB ustawić adres komórki pamięci operacyjnej.
2. Przełącznikiem obrotowym wybrać rejestr AR.
3. Nacisnąć przełącznik LOAD — na lampkach $0 \div 15$ wyświetlany jest adres komórki PAO, ustawiony na kluczach KB.
4. Przełącznikiem obrotowym wybrać rejestr, do którego ma zostać wprowadzona informacja z pamięci.
5. Nacisnąć przełącznik FETCH — na lampkach $0 \div 15$ wyświetlana jest informacja odczytana z komórki pamięci o adresie ustawionym na kluczach KB. Zawartość rejestru AR (wyświetlana na lampkach $0 \div 15$ po wybraniu rejestru AR przełącznikiem) zwiększona jest o 1.

5.4. Przykład wprowadzania i wykonywania programu

Od adresu $/00400/g^*$ wprowadzić program na dodanie 1 do rejestru R7 i porównanie zawartości rejestrów R6 i R7. $R6 := /01500/g$ $R7 := /01477/g$ Postać programu:

$/00400/g$ AWT, 7, 1
 CW, 6, 7

Wartość początkowa rejestrów:

$R6 := /01500/g$

$R7 := /01477/g$

1. Przełącznikiem wybrać rejestr R6.
2. Ustawić na kluczach KB $/01500/g$
3. Nacisnąć przełącznik LOAD — na lampkach $0 \div 15$ wyświetlona jest informacja ustawiona na kluczach.
4. Przełącznikiem wybrać rejestr R7.
5. Ustawić na kluczach KB $/01477/g$
6. Nacisnąć przełącznik LOAD — na lampkach $0 \div 15$ wyświetlona jest wartość $/01477/g$
7. Przełącznikiem obrotowym wybrać rejestr AR.
8. Ustawić na kluczach KB $/00400/g$.
9. Nacisnąć przełącznik LOAD — na lampkach $0 \div 15$ wyświetlona jest $/0400/g$.
10. Ustawić przełącznik obrotowy w położeniu KB.
11. Ustawić na kluczach $/140001/g$ (rozkaz AWT, 7, 1)

* — zapis oktalny

12. Nacisnąć przełącznik STORE.
13. Ustawić na kluczach / 106607/g (rozkaz CW, 6, 7)
14. Nacisnąć przełącznik STRORE.
15. Wybrać rejestr IC.
16. Ustawić na kluczach /00400/g
17. Nacisnąć przełącznik LOAD.
18. Nacisnąć przełącznik CYCLE – zawartość rejestru IC wyświetlona na lampkach $0 \div 15$ zwiększa się o 1. Zawartość rejestru R6 wyświetlona na lampkach po wybraniu rejestru przełącznikiem zwiększona jest o 1 i wynosi /01500/g
19. Nacisnąć przełącznik CYCLE – zawartość rejestru IC zwiększa się o 1 i wynosi /00402/g.
W rejestrze RO zapalona jest lampka 5 (wskaźnik E) oznaczająca, że zawartość rejestrów R6 i R7 są jednakowe.

UWAGA: Opis eksploatacji systemu pod nadzorem systemu operacyjnego jest zawarty w tomie III DTR.

system MERA 400

DOKUMENTACJA TECHNICZNO RUCHOWA

TOM I

Część 5

INSRUKCJA PAKOWANIA MODUŁÓW
SYSTEMU MERA-400

Nowości w opisie są zaznaczane na marginesie. Znak przy numerze strony dotyczy numeracji stron.

Z A P I S Z M I A N

Z M I A N A

O P I S

"A" 30.12.78

Oryginał

21-006581-01-9

Uwagi kierować do : Centrum Naukowo Produkcyjne Technik Komputerowych i Pomiarów
im. J. Krasickiego , ul. Łopuszańska 117/123 02-232 Warszawa

Producent zastrzega sobie prawo wprowadzania zmian w dokumentacji.

SPIS TREŚCI

1. OPIS OGÓLNY.....	1-1
2. WYKAZ MODUŁÓW I OPAKOWAŃ.....	1-2
3. INSTRUKCJA PAKOWANIA.....	13

1. OPIS OGÓLNY

Moduły systemu MERA-400 dostarczane są:

- w opakowaniach transportowych wielokrotnego użytku, w formie skrzyń o konstrukcji ramowej i o wielowarstwowym poszyciu, których poszczególne elementy (czoła, boki, wieko, dno) łączone są na śruby;
- w opakowaniach bezpośrednich jednorazowego użytku, w formie pudeł kartonowych o ściankach łączonych drutem intrrolligatorskim płaskim.

Części zapasowe oraz wyposażenie dodatkowe zapakowane w opakowania bezpośrednie umieszczone są w opakowaniach transportowych modułów.

W przypadku modułów, w których skład wchodzi urządzenia z zakupu, części zapasowe oraz wyposażenie dodatkowe do tych urządzeń są zapakowane w opakowania bezpośrednie tych urządzeń.

2. WYKAZ MODUŁÓW I OPAKOWAŃ

Oznaczenie modułu	Rodzaj opakowania	Wymiary zewnętrzne	Ilość sztuk	Punkt Instrukcji pakowania
MJC-400	skrzynia	975x660x620	1	3.1.
MPU-400	skrzynia	975x660x440	1	3.2.
MCT1001A-400	skrzynia skrzynia	280x555x440 705x605x800	1	3.3.
MDT105S-400	skrzynia skrzynia	280x555x440 705x605x800	1	3.4.
MKZ-400	skrzynia	695x650x420	1	3.5.
MAZ-400	skrzynia	695x650x420	1	3.6.
MDZ180-400	Opakowanie handlowe z Z-du MERA-BŁONIE		1	3.7.
MKSR-400	Opakowanie handlowe z Z-du MERA-BŁONIE		1	3.8.
MZS-400	skrzynia	975x660x355	1	3.9.
MPD9425-400	skrzynia	1030x745x1206	1	3.10.
MPZ-400	skrzynia	975x660x440	1	3.11.
MPOF-400	skrzynia	975x660x440	1	3.12.
MRC-400	skrzynia	1090x780x1505	1	3.13.
MPS-400	skrzynia skrzynia skrzynia	1090x780x1505 975x660x620 975x660x355	1 1 1	3.14.
MCT2100-400	skrzynia skrzynia	1185x630x380 705x1015x800	1 1	3.15.
MMV340-400	skrzynia opakowanie handlowe monitora VT-340	525x685x720	1 1	3.16.
MDE-400	skrzynia	700x805x946	1	3.17.
MPI-400	skrzynia	695x650x420	1	3.18.
BA-400	puddo tekturowe	620x155x145	1	3.19.
MPT305-400	skrzynia	645x580x605	1	3.20.
Akcesoria zestawu	torebka z folii puddo tekturowe		2 3	3.21

3. INSTRUKCJA PAKOWANIA

3.1. Pakowanie Modułu Jednostki Centralnej MJC-400 i MJC-400d

Opakowanie MJC-400 stanowią:

- skrzynia wraz z wyposażeniem wewnętrznym
- opakowanie z folii
- opakowanie bezpośrednio pakietów
- opakowanie bezpośrednio kabli
- opakowanie fabryczne pakietu pamięci.

Pakiety wyjęte z bloku JC zapakować w opakowania bezpośrednio, a pakiety pamięci w opakowania fabryczne.

Kabel i terminator włożyć do worka z folii.

Blok JC umieścić w worku z folii wraz z dwoma woreczkami silikażelowego pochłaniacza wilgoci po 200 g każdy.

Bok worka zgrzewać szczelnie szwem podwójnym. Otworzyć skrzynię, umieścić na jej dnie 4 narożniki z gumy porowatej, włożyć ramę dolną i ustawić w niej urządzenie.

Na ustawione w skrzyni urządzenie nałożyć ramę górną, a następnie między ramę a poszycie skrzyni umieścić 4 narożniki z gumy porowatej. W wolne przestrzenie między bokami urządzenia a poszyciem skrzyni wsunąć kabel KSA. Na ramie górnej ułożyć tabelę połączeń.

Skrzynię zamknąć wiekiem, w otwory wprowadzić śruby, nałożyć podkładki i skrócić nakrętkami. W wersji MJC-400d bloki pamięci drutowej pozostają w bloku JC.

3.2. Pakowanie Modułu Pudła Uniwersalnego MPU-400

Opakowanie MPU-400 stanowią:

- skrzynia
- opakowanie z folii
- opakowanie bezpośrednio pakietów
- opakowanie bezpośrednio kabli

Pakiety wyjęte z bloku PU zapakować w opakowania bezpośrednio.

Kabel włożyć do worka z folii.

Blok PU umieścić w worku z folii wraz z dwoma woreczkami silikażelowego pochłaniacza wilgoci po 200 g każdy. Bok worka zgrać szczelnie szwem podwójnym. Otworzyć skrzynię, umieścić na jej dnie 4 narożniki z gumy porowatej, ułożyć ramę dolną i ustawić w niej urządzenie. Na ustawione w skrzyni urządzenie nałożyć ramę górną, następnie między ramę a elementy skrzyni wsunąć 4 narożniki z gumy porowatej. W wolną przestrzeń między bokami urządzenia a poszyciem skrzyni wsunąć kabel Interfejsu MERA-400. Na ramie górnej ułożyć kopertę z instrukcją instalacji i uruchamiania wraz z tabelą połączeń. Skrzynię zamknąć wiekiem, w otwory wprowadzić śruby, nałożyć podkładki i skrócić nakrętkami.

3.3. Pakowanie Modułu Czytnika MCT1001A-400

Opakowanie transportowe Modułu MCT1001A-400 stanowią dwie skrzynie:

- opakowanie transportowe CT i DT
- opakowanie transportowe stolika uniwersalnego

Pakowanie bezpośrednio bloku czytnika CT-1001A

Blok czytnika umieścić w opakowaniu bezpośrednim na pasach amortyzacyjnych w ten sposób, aby prowadnica taśmy znalazła się przy lewym boku skrzynki z dolnym poziomym pasem amortyzacyjnym. Na skrzynce położyć wieko i przymocować je przy pomocy czterech śrub.

Pakowanie bezpośrednie pakietu SM, kabli KB-CT1, instrukcji instalacji i uruchomienia

Pakiet zapakować w pudełko tekturowe, kable włożyć do worka z folii, instrukcję instalacji i uruchomienia umieścić w kopercie. Tak zapakowane elementy ułożyć w skrzynce. Skrzynkę zamknąć przez przymocowanie wieka 4-ma śrubami.

Opakowanie transportowe CT i w/w elementów

Opakowane w opakowania bezpośrednie blok czytnika i elementy umieścić w skrzyni na narożnikach amortyzacyjnych. Włożyć pozostałe narożniki, na skrzynię położyć wieko i przymocować je za pomocą 10 śrub.

Opakowanie transportowe stolika uniwersalnego i kosza

Połączyć śrubami bok, czoło i dno skrzyni. Ustawić stolik na dnie i umieścić pod nim kosz. Dołączyć pozostały bok i czoło oraz wprowadzić wkładki z tektury falistej. Na skrzynię położyć wieko i przymocować śrubami.

3.4. Pakowanie Modułu Dziurkarki MDT105S-400

Opakowanie transportowe Modułu MDT105S-400 stanowią dwie skrzynie:

- opakowanie transportowe CT i DT
- opakowanie transportowe stolika uniwersalnego

Pakowanie bezpośrednie bloku dziurkarki DT-105S

Przed pakowaniem odłączyć od dziurkarki szpulę oraz pojemnik z tworzywa. Dziurkarkę umieścić w opakowaniu bezpośrednim na pasach amortyzacyjnych w normalnej pozycji pracy. Do przestrzeni wydzielonej przegrodą włożyć szpulę i pojemnik owinięte w papier. Na skrzynkę położyć wieko i przymocować je przy pomocy 4 śrub.

Pakowanie bezpośrednie pakietu SM, kabla KB-DT, instrukcji instalacji i uruchomienia

Pakiet zapakować w pudełko tekturowe, kabel włożyć do worka z folii, instrukcję instalacji i uruchomienia umieścić w kopercie. Tak zapakowane elementy włożyć w skrzynkę. Skrzynkę zamknąć przez przymocowanie wieka 4 śrubami.

Opakowanie transportowe DT i w/w elementów

Opakowanie w opakowania bezpośrednie blok dziurkarki i elementy umieścić w skrzyni na narożnikach amortyzacyjnych. Włożyć pozostałe narożniki, na skrzynię położyć wieko i przymocować je za pomocą 10 śrub.

Opakowanie transportowe stolika uniwersalnego i kosza

wg. p. 3.3.

3.5. Pakowanie Modułu Kanału Znakowego MKZ-400

Opakowanie bezpośrednie Modułu MKZ-400 stanowi opakowanie bezpośrednie pakietu.

Pakiety SM-KZB11 i SM-KZA11 zapakować w jedno pudło.

Kopertę z instrukcją instalacji i uruchomienia przymocować do pudła taśmą samoprzylepną.

O ile tak uformowanej jednostki nie można dołączyć do opakowania transportowego (skrzyni) innego modułu, należy ją umieścić w oddzielnej skrzyni. Luzy wypełnić tekturą falistą, ścinkami gąbki lub gumy porowatej.

3.6. Pakowanie Modułu Arytmometru Wielokrotnej Precyzji MAZ-400

Opakowanie bezpośrednie Modułu MAZ-400 stanowi opakowanie bezpośrednie pakietu.

Pakiety F-PSZ-1, F-PMZ-1, F-PAZ-1 zapakować do dwóch pudeł (w jednym pudle 2 pakiety, w drugim pudle 1 pakiet).

Kopertę z instrukcją instalacji i uruchomienia przymocować do pudła taśmą samoprzylepną. O ile tak uformowanej jednostki nie można dołączyć do opakowania transportowego (skrzyni) innego modułu, należy ją umieścić w oddzielnej skrzyni. Luzy wypełnić tekturą falistą, ścinkami gąbki lub gumy porowatej.

3.7. Pakowanie Modułu Drukarki MDZ180-400

Opakowanie transportowe drukarki DZM-180 stanowi opakowanie handlowe stosowane przez zakład MERA-BŁONIE.

Pozostałe elementy modułu tj. pakiet SM-UZM11 umieścić w pudle, kabel KB-DZM włożyć do worka z folii, instrukcję instalacji i uruchomienia umieścić w kopercie.

O ile tak zapakowanych pozostałych elementów modułu nie można dołączyć do opakowania drukarki lub do skrzyni innego modułu, należy je umieścić w oddzielnej skrzyni. Luzy wypełnić tekturą falistą, ścinkami gąbki lub gumy porowatej.

3.8. Pakowanie Modułu Drukarki MKSR-400

wg p. 3.7.

3.9. Pakowanie Modułu Zasilania Sekwencyjnego MZS-400

Opakowanie MZS-400 stanowi skrzynia wraz z wyposażeniem wewnętrznym w postaci 8 narożników dwóch ram, oraz opakowania z folii.

Otworzyć skrzynię, umieścić na jej dnie 4 narożniki z gumy porowatej, włożyć ramę i ustawić w niej urządzenie (obudowa kpl MZS-400). Urządzenie włożyć do worka z folii wraz z 200 gramowym woreczkiem silikażelowego pochłaniacza wilgoci, a następnie bok worka zgrzewać szczelnie szwem podwójnym. Na ustawione w skrzyni urządzenie nałożyć ramę, następnie między ramę a elementy skrzyni wsunąć 4 narożniki z gumy porowatej.

W wolną przestrzeń między bokami urządzenia a poszyciem skrzyni wsunąć kable w opakowaniu bezpośrednim.

Na ramie górnej ułożyć kopertę z tabelą połączeń, schematem ideowym, instrukcją kontroli i opisem technicznym. Skrzynię zamknąć wlekiem, w otwory wprowadzić śruby, nałożyć podkładki i skrócić nakrętkami.

O ile kable nie zmieszczą się w w/w skrzyni i nie będą mogły być dołączone do opakowania transportowego, innego modułu, należy je włożyć do oddzielnej skrzyni.

3.10. Pakowanie Modułu Pamięci Dyskowych MPD9425-400

Opakowanie MPD9425-400 stanowi:

- opakowanie transportowe pamięci dyskowej MERA-9425
- opakowanie bezpośrednio pamięci dyskowej MERA-9425
- opakowanie z folii
- opakowanie bezpośrednio pakietu.

Pakiety opakowane w opakowaniu bezpośrednim wraz ze sznurami, tabelą połączeń oraz instrukcją instalacji i uruchomienia umieścić wewnątrz dolnej części pamięci i zabezpieczyć przed przemieszczaniem taśmą samoprzylepną.

Z obu boków osłony górnej ułożyć płyty czołowe. Na osłonie górnej położyć 4 woreczki po 200 g każdy silikażelowego pochłaniacza wilgoci. Tak przygotowaną pamięć umieścić w opakowaniu z folii, którego brzeg zgrzać szczelnie szwem podwójnym ciągłym.

Następnie całość ustawić na listwach tacy wewnętrznej opakowania bezpośredniego, klapki odchylić ku górze i nasunąć nasuwę (klapki tacy powinny być wewnątrz nasuwy).

Na dnie skrzyni ułożyć symetrycznie cztery narożniki, a następnie ustawić na nich urządzenie w opakowaniu bezpośrednim. Do dna skrzyni dołączyć kolejno czoła i bok i oraz połączyć czoła z bokami za pomocą śrub.

Od góry włożyć cztery narożniki, skrzynię zamknąć wiekiem i połączyć wieko z czołami i bokami za pomocą śrub.

3.11. Pakowanie Modułu Pamięci Zewnętrznych MPZ-400

wg p. 3.2.

3.12. Pakowanie Modułu Pamięci Operacyjnej Ferrytowej MPOF-400

— wg p. 3.2.

Pakiety pamięci ferrytowej pakować w opakowanie fabryczne (ELWRO).

3.13. Pakowanie Modułu Szafy Standardowej MRC-400

Opakowanie MRC-400 stanowi skrzynia:

— opakowanie transportowe szafy MRC-400

Skrzynię rozmontować.

Na listwach amortyzacji dna ustawić szafę przodem w kierunku czoła. Nóżki szafy powinny wypaść na zewnątrz pasów amortyzacji.

Zmontować skrzynię łącząc jej elementy śrubami w następującej kolejności: czoła i boki z dnem, czoła z bokami, wieko z czołami i bokami.

3.14. Pakowanie Modułu Podstawowego Systemu MPS-400

Opakowanie MPS-400 stanowią skrzynie:

— opakowanie transportowe szafy MRC-400

— opakowanie transportowe MJC-400

— opakowanie transportowe MZS-400

Pakowanie Modułu MRC-400

wg p. 3.13.

Pakowanie Modułu MJC-400

wg p. 3.1.

Pakowanie Modułu MZS-400

wg p. 3.9.

3.15. Pakowanie Modułu Czytnika MCT2100-400

Opakowanie transportowe Modułu Czytnika MCT2100-400 stanowią dwie skrzynie:

— opakowanie transportowe CT-2100, RT-2200; ZT-2200

— opakowanie transportowe stolika uniwersalnego

Pakowanie bezpośrednio pakietu SM, kabla KB-CT2, sznura uziemień, przedłużacza sieciowego oraz instrukcji instalacji i uruchomienia

Pakiet owinać folią pęcherzykową a następnie jednokrotnie tekturą falistą 3-warstwową wg PN-88/P-50527 i zakleić taśmą samoprzylepną.

Kable włożyć do trzech worków z folii.
Instrukcję instalacji i uruchomienia umieścić w kopercie.

Pakowanie transportowe CT, RT, ZT oraz w/w elementów

Opakowane w opakowania bezpośrednie CT-2100 (wraz z częściami zamiennymi i wyposażeniem) RT-2200 oraz ZT-2200 (opakowania bezpośredniego dostarcza producent CT, RT, ZT) umieścić w skrzyni na narożnikach amortyzacyjnych.

Z góry włożyć pozostałe narożniki. Wzdłuż ścian bocznych między skrzynią a opakowaniem bezpośrednim ZT umieścić worki z kablami. Pakiet w opakowaniu bezpośrednim oraz w opakowaniu z folii zgrzanym szczelnie szwem podwójnym oraz kopertę z instrukcją umieścić na górnej powierzchni opakowania bezpośredniego RT i zabezpieczyć je przed przemieszczaniem przez przyklejenie do opakowania bezpośredniego taśmą samoprzylepną.

Następnie na skrzynię położyć wieko i przymocować je za pomocą 10-clu śrub.

Opakowanie transportowe stolika uniwersalnego i kosza

Połączyć śrubami bok, czoło i dno skrzyni. Ustawić stolik na dnie skrzyni i umieścić pod nim kosz dnem do góry.

Między dnem skrzyni a koszem umieścić jedną warstwę tektury falistej 3-warstwowej.

Przestrzeń między dnem kosza oraz blatem stolika wypełnić tekturą falistą 3-warstwową tak, aby zabezpieczyć kosz przed przemieszczaniem.

Dołączyć pozostały bok i czoło. Na skrzynię położyć wieko i przymocować śrubami.

3.16. Pakowanie Modułu Monitora Ekranowego MMV340-400

Opakowanie transportowe modułu MMV340-400 stanowią

- opakowanie fabryczne monitora VT-340
- opakowanie transportowe stolika "450x660" (wraz z wyposażeniem)
- opakowanie z folii
- opakowanie bezpośrednie pakietu
- opakowanie bezpośrednie kabli.

Monitor ekranowy VT-340 zdjąć ze stolika i zapakować w opakowanie fabryczne.

Pakiet zapakować w pudło tekturowe.

Kable i części zapasowe włożyć do worka z folii, instrukcję instalacji i uruchomienia umieścić w kopercie. Stolik ustawić na dnie skrzyni. Worek z kablami i pudło z pakietem położyć pod stolikiem na dnie i zabezpieczyć przed przemieszczaniem taśmą samoprzylepną.

Połączyć śrubami bok, czoło i dno skrzyni, kopertę z instrukcją położyć na blacie stolika. Dołączyć pozostały bok i czoło oraz wprowadzić wkładki z tektury falistej.

Na skrzynię położyć wieko i przymocować śrubami.

3.17. Pakowanie Modułu Dysku Elastycznego MDE-400

Opakowanie Modułu MDE-400 stanowią:

- opakowanie modułu MDE-400
- opakowanie z folii
- opakowanie bezpośrednie pakietów
- opakowanie bezpośrednie kabli

Pakiety włożyć do pudeł tekturowych.

Kable oraz części zapasowe włożyć do worka z folii.

Szafkę umieścić w worku z folii wraz z dwoma woreczkami silikażelowego pochłaniacza wilgoci po 200 g każdy. Na górnej pokrywie szafki położyć instrukcję instalacji i uruchamiania. Brzeg worka zaspawać szczelnie szwem podwójnym.

Na listwach amortyzacji dna skrzyni ustawić szafkę przodem w kierunku jednego z czoł opakowania. Pakiety umieścić w kasetach boków skrzyni. Kable włożyć pomiędzy tylną ściankę skrzyni, a tylną ściankę szafki.

Zmontować skrzynię łącząc jej elementy śrubami w następującej kolejności: czoła i boki z dnem, czoła z bokami, wieko z czołami i bokami.

3.18. Pakowanie Modułu Kanału Automatyki MPI-400

Opakowanie bezpośrednie modułu MPI-400 stanowi opakowanie pakietu oraz opakowanie kabli.

Pakiet A-PI3-1 zapakować w pudło.

Kable pakować w dwa worki z folii. Kopertę z instrukcją instalacji i uruchomienia przymocować do pudła taśmą samoprzylepną.

O ile tak uformowanej jednostki nie można dołączyć do opakowania transportowego (skrzyni) innego modułu, należy ją umieścić w oddzielnej skrzyni. Luzy wypełnić tekturą falistą, ścinkami gąbki lub gumy porowatej.

3.19. Pakowanie bloku BA-400

Opakowanie bloku BA-400 stanowi pudło tekturowe.

Do nasuwy włożyć jedną ściankę i przyszyć drutem introllgatorskim. Każdy amortyzator oddzielnie owinąć dwoma warstwami papieru pakowego asfaltowego. Następnie w pudło wsunąć amortyzatory śrubami do siebie, umieszczając instrukcję instalacji i montażu między nimi.

Pudło zamknąć drugą ścianką i przyszyć ją drutem introllgatorskim.

3.20. Pakowanie Modułu Pamięci Taśmowej MPT305-400

Opakowanie transportowe modułu MPT305-400 stanowią:

- opakowanie transportowe podstawy
- opakowanie bezpośrednie pakietów
- opakowanie bezpośrednie kabli.

Pakiet włożyć do pudła tekturowego, kable, sznur uziemiń, przedłużacz sieciowy włożyć do worka z folii.

Blok pamięci taśmowej zdjąć z podstawy odkręcając 4 wkręty mocujące nóżki i zapakować w opakowanie fabryczne. Blok formatera odkręcić od podstawy (4 wkręty M5), wyjąć i zapakować w opakowanie fabryczne.

Wkręty mocujące wkręcić z powrotem w odpowiednie otwory.

Podstawę ustawić na dnie skrzyni. Worek z kablami i pudło z pakietem oraz instrukcją położyć pod stolikiem na dnie i zabezpieczyć przed przemieszczaniem taśmą samoprzylepną.

Połączyć śrubami boki, czoła, wieko i dno.

3.21. Pakowanie akcesoriów zestawu systemu MERA-400

Opakowanie akcesoriów stanowią:

- opakowanie z folii pęcherzykowej
- opakowanie bloku BA-400.

3.21.1. Pakowanie przedłużacza PP-400

Przedłużacz włożyć do torebki z folii; brzeg torebki zgrzewać liną ciągną. Na wierzchu torebki nakleić etykietkę samoprzylepną zawierającą nazwę i numer rysunku, nazwę producenta.

3.21.2. Pakowanie przedłużacza SM-PP11

Przedłużacz włożyć do torebki z folii; brzeg torebki zgrzewać liną ciągną. Na wierzchu torebki przykleić etykietkę samoprzylepną zawierającą nazwę, numer rysunku i nazwę producenta.

3.21.3. Pakowanie przedłużacza SM-PP-21 wg punktu 3.21.1.

3.21.4. Pakowanie płytki uniwersalnej P-UNI-1 wg p. 3.21.1.

3.21.5. Pakowanie płytki pakietu SM-PUX-11 wg p. 3.21.2.

3.21.6. Pakowanie płytki pakietu SM-PUX-21 wg p. 3.21.1.

3.21.7. Pakowanie złącza 801

Złącze włożyć do torebki z folii, brzeg torebki zgrzewać liną ciągłą. Na wierzch torebki nakleić etykietką samoprzylepną zawierającą nazwę i numer rysunku oraz nazwę producenta.

3.21.8. Pakowanie złącza w obudowie wg p. 3.21.7.

3.21.9. Pakowanie bloku amortyzatora szafy BA-400 wg p. 3.19.