

PRZEMYSŁOWY INSTYTUT AUTOMATYKI I POMIARÓW  
„MERA-PIAP”

Al. Jerozolimskie 202 02-222 Warszawa Telefon 23-70-81

074

A

OSRODEK AUTOMATYKI ELEKTRYCZNEJ

ZESPÓŁ BUDOWY CYFROWYCH URZĄDZEN SYSTEMOWYCH

Główny wykonawca mgr inż. M. Słodczyk

*MSDe*

Wykonawcy mgr inż. T. Kacprowski , mgr inż. M. Partyka,  
mgr inż. M. Słodczyk :

Konsultant

Nr zlecenia

U-23.01.01.D

Opracowanie urządzeń mikroprocesorowych  
systemu MIR-PROWAY.

ETAP 11: Opracowanie, uruchomienie i badania  
systemu operacyjnego jednostki cent-  
tralnej z mikroprocesorem 8-bitowym.

Instrukcja użytkowania systemu RTMI.

Zleceniodawca problem węzłowy 06.1

Pracę rozpoczęto dnia 05.83r.

KIEROWNIK ZESPOŁU

*A. Syrczyński*

dr inż. A. Syrczyński

zakończono dnia 15.12.83r.

KIEROWNIK OSRODKA

*T. Missala*

prof. dr inż. T. Missala

Praca zawiera:

stron 49

rysunków -

fotografii -

tabel -

tablic -

Rozdzielnik:

Egz. 1 BOINTE

Egz. 2 OAK

Egz. 3 OAE 83

Egz. 4 OAE 8

Egz. 5 ZAP

Egz. 6 ZAP

Nr. rejestr. 5167

1

## Analiza deskryptorowa

URZADZENIA AUTOMATYCZNEJ REGULACJI ISTEROWANIA KSAP + MIR-PROWAY +  
+ MIKROPROCESOR + OPROGRAMOWANIE

## Analiza dokumentacyjna

Instrukcja użytkowania systemu RTMP jednostki centralnej z mikroprocesorem 8-bitowym zawiera opis eksploatacji systemu operacyjnego czasu rzeczywistego, opis ekstrakodów systemowych, opis programu monitora operatorskiego, procedur sprzężenia z pamięcią kasetową, opisy podprogramów systemowych, konfigurację sprzętową oraz dane potrzebne do generacji systemu dla konkretnych aplikacji.

## Tytuły poprzednich sprawozdań

"Instrukcja użytkowania"oprogramowania podstawowego pakietu jednostki centralnej z mikroprocesorem 8-bitowym / wersja do badań /. Nr. rej. 4807.

62-50 Teoria i praktyka techn. - regulacji i sterowania

UKD

Druk „MERA-PIAP”/TW-62/5000 egz.

## Spis treści

1. Wstęp
2. Eksploatacja systemu RTMT
  - 2.1. Zarządzanie zadaniem
  - 2.2. Budowa zadania
  - 2.3. Włączanie oprogramowania użytkownika
  - 2.4. Obsługa przerw
  - 2.5. Obsługa urządzeń zewnętrznych
  - 2.6. Warunkowe zawieszanie zadań
3. Ekstrakody systemowe
4. Eksploatacja programu monitor operatorski
  - 4.1. Opis ogólny monitora operatorskiego
  - 4.2. Opis dyrektyw monitora operatorskiego
5. Procedury sprzężenia z pamięcią kasetową
6. Podprogramy systemowe
7. Konfiguracja sprzętowa
8. Generacja systemu

## 1. Wstęp

System RTMT jest wielozadaniowym systemem operacyjnym czasu rzeczywistego, przeznaczonym dla mikrokomputera MM 80. System umożliwia sterowanie programami - zadaniami w zależności od czasu, wzajemnych uwarunkowań oraz zdarzeń zewnętrznych. Ponadto system zapewnia komunikację oprogramowania użytkowego z monitorem ekranowym, drukarką znakowo-mozaikową, czytnikiem i dziurkarką taśmy papierowej oraz pamięcią kasetową. System posiada ekstrakody umożliwiające sterowanie zadaniami i urządzeniami zewnętrznymi.

W skład systemu wchodzi program inicjacji i restartu automatycznego oraz monitor przerw. Pod kontrolą systemu może pracować jako zadanie program monitora operatorskiego.

Dzięki modułowej budowie możliwe jest dla minimalizowania zajętości pamięci i czasu działania, generowanie wersji systemu odpowiedniej dla konkretnej aplikacji.

## 2. Eksploatacja systemu RTMT

### 2.1. Zarządzanie zadaniami

Zasadniczą częścią systemu jest koordynator zarządzający pracą zadań. Koordynator uruchamia zadanie zgodnie z zasadą priorytetów względnych. Zgłoszenie do wykonania zadania o wyższym priorytecie w trakcie wykonywania się zadania o priorytecie niższym, nie powoduje przerwania wykonywania się tego zadania. Następuje jedynie ustawienie odpowiednich wskaźników i zadanie zostanie uruchomione po zakończeniu wykonywania się zadania aktualnie pracującego. Istnieje możliwość zmiany zasady uruchamiania zadań przez koordynator z priorytetów względnych na priorytety bezwzględne przez odpowiednie napisanie programów obsługi przerwania użytkownika.

Koordynator może obsługiwać do 31 zadań o numerach od 1 do 31, przy czym zadanie nr 1 ma priorytet najwyższy, a zadanie 31 priorytet najniższy. Ilość zadań obsługiwanych przez koordynator może być ustalona na etapie generacji systemu.

W trakcie pracy systemu koordynator wyszukuje zadanie o najwyższym priorytecie, które jest zgłoszone do wykonania /lub do kontynuacji po zawieszeniu/ i uruchamia je. Po zakończeniu pracy tego zadania koordynator ponownie wyszukuje do wykonania zadanie o najwyższym priorytecie.

### 2.2. Budowa zadania

Aby dowolny program mógł pracować pod kontrolą systemu operacyjnego, konieczne jest nadanie mu formatu programu-zadania.

Zadanie musi zaczynać się ustawieniem wskaźnika stosu na niezastreżony obszar RAM-u wewnętrznego lub RAM użytkownika.

## 2.5. Obsługa urządzeń zewnętrznych

Komunikacja zadań użytkownika ze wszystkimi urządzeniami zewnętrznymi dokonywana jest za pomocą ekstrakodów i podprogramów systemowych.

Zadanie może korzystać z urządzenia zewnętrznego pod warunkiem, że urządzenie to nie jest zajęte przez inne zadanie. Jeżeli urządzenie jest wolne, to pierwsze użycie ekstrakodu obsługującego to urządzenie przydziela je do zadania wywołującego ekstrakod. Gdy w chwili użycia ekstrakodu, urządzenie jest przydzielone do innego zadania, zadanie wywołujące ekstrakod zawiesza się w oczekiwaniu na zwolnienie urządzenia. Z chwilą zajęcia urządzenia przez dane zadanie, urządzenie jest przydzielone do tego zadania, aż do momentu użycia ekstrakodu zwalniającego to urządzenie. Możliwe jest wtedy zajęcie tego urządzenia przez zadanie o najwyższym priorytecie spośród czekających na to urządzenie. Należy pamiętać, że nie zwolnienie urządzenia uniemożliwia dostęp do niego dla innych zadań. Jedynym wyjątkiem jest podstawowe urządzenie operatorskie /urządzenie nr 0/, na które można wyprowadzać komunikaty alarmowe /przy użyciu ekstrakodu ALARM/ niezależnie od faktu zajęcia tego urządzenia przez inne zadanie.

## 2.6. Warunkowe zawieszanie zadań

Użytkownik może przy użyciu ekstrakodów systemowych zawiesić wykonywanie bieżącego zadania w oczekiwaniu na spełnienie określonego warunku zewnętrznego. Sprawdzenie, czy dany warunek jest spełniony, może być realizowane dwojako: przez system lub za pomocą odrębnego programu użytkownika uruchamianego przez koordynator. Ilość warunków obu typów jest ustalana na etapie generacji systemu. Maksymalna ilość warunków każdego typu wynosi 127.

Długość stosu ustala użytkownik na podstawie analizy swojego oprogramowania pamiętając o tym, że zadanie może być przerwane przez przerwania zewnętrzne /do 8 przerwania/.

Zadanie musi kończyć się skokiem do adresu określonego w systemie etykietą KOZAD.

### 2.3. Włączanie oprogramowania użytkownika

Oprogramowanie systemowe umieszczone jest w pamięci typu EPROM, pamięci jednostki centralnej od adresu 0 do adresu 1FFFH.

Po włączeniu zasilania, system startuje automatycznie i po wykonaniu niezbędnych działań inicjujących, uruchamia przy nie włączonych przerwaniach zadanie Nr 1, o ile znajduje się ono na pamięci stałej typu EPROM. Zadanie to powinno być wykorzystane przez użytkownika do zainicjowania własnego oprogramowania, to znaczy wykonania takich czynności jak: zainstalowanie zadań, podanie adresów programów obsługi przerwania, ustawienie własnych obszarów roboczych itp. Zadanie nr 1 inicjacji użytkownika musi mieć budowę typowego zadania użytkownika, to znaczy musi ustawiać wskaźnik stosu i kończyć się skokiem do systemowej etykiety KOZAD. Zakończenie zadania nr 1 powoduje włączenie przerwania instrukcją EI i rozpoczęcie pracy koordynatora.

Zadanie nr 1 inicjujące oprogramowanie użytkownika zainstalowane jest przez system od stałego adresu określonego na etapie generacji systemu etykietą INICU.

Jeżeli pod adresem określonym etykietą INICU nie ma pamięci stałej, to znaczy jest pamięć typu RAM lub brak jakiegokolwiek pamięci, system nie uruchamia zadania inicjacji użytkownika lecz po wykonaniu swojej inicjacji przechodzi przy włączonych przerwaniach do koordynatora. Wówczas zadaniem inicjującym oprogramowanie

użytkownika /jeśli jest niezbędne/ może być dowolne zadanie uruchamiane dyrektywą monitora operatorskiego

#### 2.4. Obsługa przerwania

Użytkownik może wykorzystywać w swoich zadaniach wszystkie przerwania nie używane przez zastosowaną przez niego wersję systemu.

Po inicjacji systemu włączone są tylko przerwania wykorzystywane przez system. Użytkownik może włączać i wyłączać wszystkie pozostałe przerwania w swoim programie inicjacji lub w dowolnym miejscu swojego oprogramowania wyłącznie za pomocą maskowania słowa priorytetu przerwania w układzie programowalnego sterownika przerwania 8259. Nie może jednak, w żadnym wypadku, zmieniać bitów maski przerwania dotyczących przerwania używanych przez system. Operowanie na poszczególnych bitach słowa priorytetu przerwania możliwe jest jedynie przy pomocy ekstrakodów systemowych.

Program obsługi przerwania użytkownika musi kończyć się skokiem do adresu określonego w systemie etykietą WYODT, co zapewnia kontynuację programu przerwanej przez to przerwanie. W ten sposób jest zrealizowana zasada priorytetów względnych uruchamiania zadań.

Kończąc program obsługi przerwania skokiem do adresu określonego etykietą systemową WYKOO można zapewnić realizację zasady priorytetów bezwzględnych uruchamiania zadań. Oznacza to, że w momencie zakończenia obsługi tego przerwania, koordynator sprawdza, czy jest zgłoszone do wykonania zadanie o priorytecie wyższym, niż zadanie przerwane i ewentualnie uruchamia je.



### 3. Ekstrakody systemowe

Ekstrakody systemowe stanowią jedyne źródło komunikacji oprogramowania użytkowego z systemem operacyjnym i urządzeniami zewnętrznymi. Zrealizowane są one w oparciu o instrukcję mikroprocesora RST 3 o kodzie heksadecymalnym ODFH. W następnej komórce na pamięci po kodzie instrukcji RST 3 znajduje się parametr określający typ ekstrakodu, a w kolejnych ewentualne dalsze parametry. Wszystkie ekstrakody zachowują rejestry i słowo stanu procesora niezmienione i działają przy włączonych przerwaniach.

W oprogramowaniu użytkowym wygodnie jest zdefiniować wszystkie używane ekstrakody systemowe jako makrorozkazy. W punktach opisujących poszczególne ekstrakody systemowe podano ich przykładowe makrodefinicje. Nazwy ekstrakodów nie są obligatoryjne i użytkownik definiując makroinstrukcje, może wprowadzić inne nazwy.

#### 3.1. Ekstrakod - uruchom zadanie - START

Definicja makroinstrukcji:

```
START  MACRO  NRZAD
        RST    3
        DB     0,NRZAD
        ENDM
```

Opis:

Ekstrakod START powoduje zgłoszenie zadania o numerze określonym przez parametr NRZAD do uruchomienia. Zadanie zostanie uruchomione przez koordynator, gdy przestanie działać /skończy się lub zawiesi/ wykonywane zadanie, a zadanie zgłoszone do wykonania będzie miało najwyższy priorytet spośród zadań oczekujących na uruchomienie. Zgłoszenie do uruchomienia ekstrakodem START zadania już zgłoszonego, wykonującego się lub zawieszzonego z dowolnego powodu jest ignorowane. Ekstrakod może być użyty w dowolnym zadaniu użytkownika i w programie obsługi przerwania.

Przed użyciem ekstrakodu START należy zadanie zainstalować w systemie przy pomocy ekstrakodu INSTZ lub dyrektywą programu monitor operatorski.

Przykład użycia:

START 5 - zgłoszenie do uruchomienia zadania nr 5

### 3.2. Ekstrakod - czytaj dane - CZYTD

Definicja makroinstrukcji :

CZYTD	MACRO	NRURZ, BUFWE
	RST	3
	DB	1, NRURZ
	DW	BUFWE
	ENDM	

Opis:

Ekstrakod CZYTD powoduje zawieszenie wykonywania zadania go używającego do czasu wprowadzenia z urządzenia podłączonego do interfejsu V-24 o numerze NRURZ ciągu znaków zakończonego znakiem powrotu karetki <CR>. Znaki te, łącznie ze znakiem CR zostają wprowadzone do bufora zdefiniowanego przez użytkownika od adresu BUFWE. Na etapie generacji systemu definiuje się parametr DLUGT określający maksymalną długość bufora dla wprowadzanego tekstu. Zapełnienie bufora tekstu znakami wprowadzonymi z urządzenia zewnętrznego jest równoznaczne z zakończeniem tekstu znakiem <CR> to znaczy powoduje kontynuację działania zadania od miejsca bezpośrednio za ekstrakodem CZYTD. Pierwsze użycie ekstrakodu CZYTD lub ekstrakodu pisanania na to samo urządzenie PISZD i PISZZ powoduje przydzielenie urządzenia do tego zadania aż do czasu użycia ekstrakodu ZWOLN. Jeżeli urządzenie jest przydzielone do innego zadania, to zadanie bieżące zawiesza się do czasu zwolnienia urządzenia.

wpisywanej do komórek użyta zostanie cyfra nie hex, lub adres początku będzie większy od adresu końca obszaru pamięci, nastąpi sygnalizacja błędu znakiem " ? ", po czym zgłosi się program MONITOR.

Dyrektywa sprawdza przed napisaniem każdej komórki, czy jest ona w pamięci typu RAM. Gdy tak nie jest, sygnalizuje błąd znakiem "!" i adresem pierwszej wykrytej niepoprawnej komórki, po czym przechodzi do programu MONITOR.

#### 4.2.4. DYREKTYWA Z - Zapis obszaru pamięci

Wywołanie dyrektywy:

Zaaaa <CR>

gdzie aaaa - adres pierwszej zapisywanej komórki. Jeżeli w adresie zostanie użyty znak nie będący cyfrą heksadecymalną wydrukowany będzie znak "?" i zgłosi się program MONITOR.

Gdy adres jest poprawny, następuje sprawdzenie, czy komórka o podanym w wywołaniu adresie należy do pamięci RAM. Gdy nie, to następuje sygnalizacja błędu znakiem "!" i powrót do MONITORA.

Jeżeli nie zostanie stwierdzony błąd, drukowany jest nagłówek podobnie jak w dyrektywie M. Dane należy wpisywać w kolejnych wierszach pod nagłówkiem. Każdy wiersz jest poprzedzony adresem pierwszej występującej w nim komórki. Poszczególne dane wpisywane do kolejnych komórek należy oddzielać spacją. Analiza wprowadzonych danych i zapis do komórek dokonywane są dopiero po naciśnięciu znaku <CR>. Dlatego zaleca się, po wypełnieniu całego wiersza, kończyć go znakiem <CR>, mimo że dalsze wpisywanie poprawnych danych /tzn. grup po dwie cyfry hex rozdzielonych spacją/ nie będzie uważane za błąd.

Dane w buforze BUFWE są przedstawione w postaci kodów ASCII.

Przykład użycia:

CZYTD 1, POLE1 - czytaj dane z urządzenia nr 1 do pamięci RAM, począwszy od adresu określonego etykietą POLE1,

### 3.3. Ekstrakod - pisz dane - PISZD

Definicja makroinstrukcji:

PISZD	MACRO	NRURZ,	BUFWE
	RST	3	
	DB	2, NRURZ	
	DW	BUFWE	
	ENDM		

Opis:

Ekstrakod PISZD powoduje dopisanie do kolejki wyjściowej urządzenia NRURZ podłączonego do interfejsu V-24 adresu tekstu rozpoczynającego się od etykiety BUFWE określonej przez użytkownika. Na urządzenie wyprowadzane są kolejne znaki bufora BUFWE aż do znaku NULL /o kodzie 0/ kończącego każdy wyprowadzany tekst. Ekstrakod nie zawiesza wykonywania zadania używającego go, to znaczy tekst jest wyprowadzany równocześnie z biegnącym dalej zadaniem.

Pierwsze użycie w bieżącym zadaniu ekstrakodu PISZD lub ekstrakodów pisania z zawieszeniem PISZZ i czytania CZYTD powoduje przydzielenie urządzenia do tego zadania, aż do czasu użycia w tym zadaniu ekstrakodu ZWOLN. Jeżeli urządzenie jest przydzielone do innego zadania, to zadanie bieżące zawiesza się na ekstrakodzie PISZD do czasu zwolnienia urządzenia.

Dane w buforze BUFWE muszą być przedstawione w postaci kodów ASCII.

Przykład użycia:

PISZD 2, POLE2 - pisz na urządzenie nr 2 dane umieszczone w pamięci począwszy od adresu określonego etykietą POLE2.

### 3.4. Ekstrakod - zwolnij urządzenie - ZWOLN

Definicja makroinstrukcji:

ZWOLN	MACRO	NRURZ
	RST	3
	DB	3, NRURZ
	ENDM	

Opis:

Ekstrakod ZWOLN powoduje odłączenie /zwolnienie/ urządzenia NRURZ od zadania, które ten ekstrakod użyło. Jeżeli zadanie nie używało poprzednio ekstrakodów PISZD, PISZZ lub CZYTD dotyczących urządzenia NRURZ, do ekstrakod ZWOLN nie powoduje żadnej reakcji. Jeżeli urządzenie NRURZ było używane w zadaniu przy pomocy ekstrakodów PISZD, PISZZ lub CZYTD to przed zakończeniem zadania /skokiem do systemowej dyrektywy KOZAD/ należy koniecznie zastosować ekstrakod ZWOLN. W przeciwnym wypadku nie będzie można korzystać z urządzenia w innych zadaniach.

Przykład użycia:

ZWOLN 0 - zwolnij /odłącz/ urządzenie nr 0 od zadania

### 3.5. Ekstrakod - drukuj tekst w trybie alarmowym - ALARM

Definicja makroinstrukcji:

ALARM	MACRO	BUFWY
	RST	3
	DB	4
	DW	BUFWY
	ENOM	

Opis:

Ekstrakod ALARM powoduje dopisanie do kolejki wydruków alarmowych adresu tekstu rozpoczynającego się od adresu BUFWY określonego przez użytkownika. Ekstrakod ALARM dotyczy zawsze druku na urządzeniu podstawowym systemu o numerze 0. Na urzą-

dzenie wyprowadzane są kolejne znaki bufora BUFWY aż do znaku NULL /kod ASCII 0/ kończącego tekst. Ekstrakod nie zawieszania wykonywania zadania używającego go, to znaczy tekst jest wyprowadzany równocześnie z wykonującym się zadaniem. Ekstrakod ALARM nie bada zajętości urządzenia O przez inne zadanie. Oznacza to, że tekst drukowany tym ekstrakodem jest wyprowadzany natychmiast po zakończeniu aktualnie drukowanego tekstu i może rozdzielić wydruki pisane przez inne zadanie na urządzeniu O przy pomocy ekstrakodów PISZD i PISZZ.

Dane w buforze BUFWY muszą być przedstawione w postaci kodów ASCII.

Przykład użycia:

ALARM POLE3 - pisz na urządzenie O poza kolejką tekstów dane umieszczone w pamięci od adresu określonego etykietą POLE3.

### 3.6. Ekstrakod - uruchom zadanie okresowo - OKRES

Definicja makroinstrukcji:

OKRES	MACRO	NRZAD, KWANT
	RST	3
	DB	5, NRZAD
	DW	KWANT
	ENDM	

Opis:

Ekstrakod OKRES powoduje takie ustawienie programowanych liczników systemu, że koordynator będzie zgłaszał do uruchomienia zadanie NRZAD co okres KWANT sekund. Ekstrakod może być użyty w dowolnym zadaniu /także NRZAD/ oraz w programie obsługi przerwania.

Użycie ekstrakodu OKRES w stosunku do zadania, które było już uruchomione okresowo powoduje jedynie zmianę częstotliwości

wywoływania tego zadania. Zadanie będzie uruchamiane cyklicznie aż do momentu użycia w dowolnym miejscu oprogramowania użytkownika ekstrakodu WYOKR wyłączającego zadanie okresowe. Po użyciu ekstrakodu OKRES zadanie NRZAD będzie po raz pierwszy zgłoszone do uruchomienia po czasie KWANT sekund.

Przykład użycia:

OKRES 6,3600 - uruchom zadanie nr 6 , co 3600 sekund

### 3.7. Ekstrakod - wyłącz zadanie okresowe - WYOKR

Definicja makroinstrukcji:

```
WYOKR  MACRO  NRZAD
        RST    3
        DB     6, NRZAD
        ENDM
```

Opis:

Ekstrakod WYOKR powoduje zakończenie cyklicznego uruchomienia zadania NRZAD włączonego uprzednio ekstrakodem OKRES. Próba zastosowania ekstrakodu WYOKR do zadania, które nie jest włączone jako zadanie okresowe, nie powoduje żadnych reakcji. Ekstrakod WYOKR może być użyty w dowolnym zadaniu /także NRZAD/ lub w programie obsługi przerwania .

Przykład użycia:

WYOKR 2 - wyłącz zadanie nr 2 - jeśli było uprzednio włączone jako okresowe.

### 3.8. Ekstrakod - zawieś zadanie pod warunkiem - ZAWWA

Definicja makroinstrukcji:

```
ZAWWA  MACRO  NRWAR
        RST    3
        DB     7, NRWAR
        ENDM
```

Opis:

Ekstrakod ZAWWA powoduje zawieszenie wykonywania

zadania go wywołującego do czasu stwierdzenia przez koordynator systemu, że warunek nr NRWAR jest ustawiony /spełniony/. Warunek można ustawiać i zerować przy pomocy ekstrakodów USTAW i ZERUJ w każdym innym zadaniu i w programie obsługi przerwania.

Przykład użycia:

ZAWWA 2 - zawieś bieżące zadanie w oczekiwaniu na spełnienie warunku nr 2

3.9. Ekstrakod - zawieś zadanie pod warunkiem wykonania innego programu - ZAWPR

Definicja makroinstrukcji :

ZAWPR	MACRO	NRPRO
	RST	3
	DB	8, NRPRO
	ENDM	

Opis:

Ekstrakod ZAWPR powoduje zawieszenie wykonywania zadania go wywołującego do czasu stwierdzenia przez dodatkowy program użytkownika, że można zadanie kontynuować. Program dodatkowy sprawdzający możliwość kontynuacji zadania musi być uprzednio zainstalowany przy użyciu ekstrakodu INSTW. Musi on kończyć się skokiem do systemowej etykiety KOWAR z odpowiednio ustawioną flagą CY : CY = 0, gdy warunek nie jest spełniony lub CY = 1, gdy zadanie można kontynuować.

Przykład użycia:

ZAWPR 5 - zawieś bieżące zadanie w oczekiwaniu na spełnienie warunku sprawdzanego przez program zainstalowany ekstrakodem INSTW pod nr 5.



3.10. Ekstrakod - warunek spełniony - USTAW

Definicja makroinstrukcji:

```
USTAW  MACRO  NRWAR
        RST    3
        DB     9, NRWAR
        ENDM
```

Opis:

Ekstrakod USTAW powoduje ustawienie warunku nr NRWAR. Jeśli warunek był wcześniej ustawiony, działanie ekstrakodu jest ignorowane. Użycie ekstrakodu USTAW powoduje zgłoszenie do kontynuacji zadania zawieszonoego pod tym warunkiem.

Przykład użycia:

```
USTAW  8      - ustaw warunek nr 8
```

3.11. Ekstrakod - zeruj warunek - ZERUJ

Definicja makroinstrukcji:

```
ZERUJ  MACRO  NRWAR
        RST    3
        DB     10, NRWAR
        ENDM
```

Opis:

Ekstrakod ZERUJ powoduje wyzerowanie warunku nr NRWAR

Przykład użycia:

```
ZERUJ  2      - zeruj warunek nr 2
```

3.12. Ekstrakod - ZAWIEŚ zadanie na czas - ZAWIE

Definicja makroinstrukcji :

```
ZAWIE  MACRO  KWANT
        RTS    3
        DB     11, KWANT
        ENDM
```

Opis:

Ekstrakod ZAWIE powoduje takie ustawienie programowych liczników systemu, że zadanie używające ten ekstrakod będzie kontynuowane od miejsca bezpośrednio za ekstrakodem ZAWIE po czasie KWANT sekund. Użycie ekstrakodu ZAWIE w programie obsługi przerwania jest zakazane.

Przykład użycia:

ZAWIE 10 - zawieś bieżące zadanie na 10 sekund.

3.13. Ekstrakod - uruchom zadanie o godzinie - URUCH

Definicja makroinstrukcji:

```
URUCH  MACRO  NRZAD, GODZ
        RST    3
        DB    12, NRZAD
        DW    GODZ
        ENDM
```

Opis:

Ekstrakod powoduje takie ustawienie programowych liczników systemowych, że zadanie NRZAD zostanie zgłoszone do wykonania o godzinie GODZ. Ekstrakod może być użyty w dowolnym zadaniu użytkownika /także NRZAD/ oraz w programie obsługi przerwania. Jeżeli o godzinie GODZ zadanie NRZAD jest zawieszona lub w trakcie wykonywania ekstrakod jest ignorowany. Parametr GODZ musi być podany w formacie BCD na dwóch kolejnych bajtach, przy czym pierwszy określa minuty, a drugi godzinę. żadanego czasu uruchomienia zadania. Zadanie zostanie zgłoszone do wykonania, gdy godziny i minuty czasu ustawionego w systemie osiągną parametr GODZ , a sekundy przyjmą wartość 0.

Przykład użycia:

URUCH 9,1235H - uruchom zadanie nr 9 o godzinie  
12 minut 35 sekund 00.

3.14. Ekstrakod - czytaj dane z czytnika - CZYTC

Definicja makroinstrukcji:

```
CZYTC  MACRO  ILZNA, BUFWE
        RST    3
        DB    13, ILZNA
        DW    BUFWE
        ENDM
```

Opis:

Ekstrakod CZYTC powoduje zawieszenie aktualnie wykonywanego zadania do czasu wczytania przez czytnik określonej przez parametr ILZNA ilości znaków lub wykrycia we wczytywanych znakach znaku <LF> o kodzie OAH. Znaki te są wprowadzane do bufora określonego przez użytkownika parametrem BUFWE. Po zakończeniu ekstrakodu CZYTC użytkownik może sprawdzić, czy dane zostały prawidłowo wczytane do bufora. W komórce określonej systemową etykietą STANC znajduje się informacja zwrotna ekstrakodu CZYTC w formacie:

- 0 - poprawne wczytanie danych
- 40H - czytnik nie zgłasza gotowości
- 80H - time-out czytnika - czytnik nie otrzymał nowego znaku w okresie 2 sekund

Użycie ekstrakodu powoduje przyłączenie czytnika do zadania. Przed zakończeniem zadania /skokiem do systemowej etykiety KOZAD/ należy odłączyć czytnik od zadania ekstrakodem ODLCT.

Przykład użycia:

```
CZYTC  100, POLE4 - czytaj 100 znaków z czytnika do
                  pamięci RAM począwszy od adresu
                  określonego etykietą POLE4.
```

3.15. Ekstrakod - odłącz czytnik - ODLCT

Definicja makroinstrukcji:

```
ODLCT MACRO
      RST      3
      DB       14
      ENDM
```

Opis:

Ekstrakod ODLCT powoduje odłączenie /zwolnienie/ czytnika taśmy papierowej od zadania, które ten ekstrakod użyło. Jeżeli zadanie nie używało poprzednio ekstrakodu czytania danych z czytnika CZYTC, to ekstrakod ODLCT nie powoduje żadnych reakcji. Jeżeli zadanie czytało dane z czytnika ekstrakodem CZYTC, to przed zakończeniem zadania /skokiem do systemowej etykiety KOZAD/ należy koniecznie użyć ekstrakodu ODLCT. W przeciwnym wypadku nie będzie można korzystać z czytnika w innych zadaniach /także w zadaniu monitora operatorskiego/.

Przykład użycia:

ODLCT - odłącz czytnik od aktualnie wykonującego się zadania.

3.16. Ekstrakod - dziurkuj - dane na dziurkarce - DZIUR

Definicja makroinstrukcji:

```
DZIUR  MACRO  ILZNA, BUFWY
      RST      3
      DB       15, ILZNA
      DW       BUFWY
      ENDM
```

Opis:

Ekstrakod powoduje zawieszenie wykonywania zadania go używającego do czasu wydziurkowania na dziurkarce ilości ILZNA znaków z bufora określonego przez użytkownika parametrem BUFWY. Po zakończeniu ekstrakodu DZIUR użytkownik może spraw-

dzić, czy dane zostały prawidłowo wydziurkowane. W komórce określonej systemową etykietą STAND znajduje się informacja zwrotna ekstrakodu DZIUR w formacie:

- O - poprawne wydziurkowanie danych
- 40H - dziurkarka nie zgłasza gotowości
- 80H - time-out dziurkarki - dziurkarka nie wydziurkowała nowego znaku w okresie 4 sekund

Użycie ekstrakodu powoduje przyłączenie dziurkarki do zadania. Przed zakończeniem zadania /skokiem do systemowej etykiety KOZAD/ należy odłączyć dziurkarkę od zadania ekstrakodem ODLDT.

Przykład użycia:

DZIUR 50, POLE5 - dziurkuj 50 znaków z pamięci począwszy od adresu określonego etykietą POLE5.

### 3.17. Ekstrakod - pisz dane z zawieszeniem - PISZZ

Definicja makroinstrukcji:

PISZZ	MACRO	NRURZ, BUFWY
	RST	3
	DB	16, NRURZ
	DW	BUFWY
	ENDM	

Opis:

Ekstrakod PISZZ powoduje dopisanie do kolejki wyjściowej urządzenia nr NRURZ podłączonego do interfejsu V-24 adresu tekstu <sup>rozpoczyn</sup> ~~wspomagającego~~ się etykietą BUFWY określoną przez użytkownika. Na urządzenie wyprowadzane są kolejne znaki bufora BUFWY aż do znaku NULL /o kodzie 0/ kończącego tekst. W buforze muszą znajdować się kody ASCII wyprowadzanych znaków. Ekstrakod zawiesza wykonywanie zadania używającego go aż do czasu wyprowadzenia na urządzenie ostatniego znaku bufora BUFWY.

Ekstrakod przydziela urządzenie do zadania do czasu użycia ekstrakodu ZWOLN.

Przykład użycia:

PISZZ 2, POLE6 - pisz na urządzenie 2 dane umieszczone w pamięci począwszy od adresu określonego etykietą POLE6.

### 3.18. Ekstrakod - odłącz dziurkarkę - ODLDT

Definicja makroinstrukcji:

ODLDT	MACRO	
	RST	3
	DB	17
	ENDM	

Opis:

Ekstrakod ODLDT powoduje odłączenie /zwolnienie/ dziurkarki taśmy papierowej od zadania, które ten ekstrakod użyło. Jeżeli zadanie nie używało poprzednio ekstrakodu perforowania danych na dziurkarce DZIUR, to ekstrakod ODLDT nie powoduje żadnych reakcji. Jeżeli jakiegokolwiek zadanie perforowało dane na dziurkarce ekstrakodem DZIUR, to przed zakończeniem zadania /skokiem do systemowej etykiety KOZAD/ należy koniecznie użyć ekstrakodu ODLDT. W przeciwnym wypadku nie będzie można korzystać z dziurkarki w żadnym innym zadaniu /także w zadaniu monitora operatorskiego /.

Przykład użycia:

ODLDT - odłącz dziurkarkę od aktualnie wykonującego się zadania.

### 3.19. Ekstrakod - zainstaluj zadanie - INSTZ

Definicja makroinstrukcji:

INSTZ	MACRO	NRZAD, ADRES
	RST	3
	DB	1E, NRZAD
	DW	ADRES
	ENDM	

Opis:

Ekstrakod INSTZ powoduje zainstalowanie w systemie programu, umieszczonego w pamięci począwszy od adresu określonego etykietą użytkownika ADRES, jako systemowego zadania o numerze NRZAD.

Przykład użycia:

INSTZ 5,ZAD05 - zainstaluj program zaczynający się etykietą ZAD05 jako zadanie nr 5

3.20. Ekstrakod - zainstaluj przerwanie - INSTP

Definicja makroinstrukcji:

INSTP	MACRO	NRPRZ, ADRES
	RST	3
	DB	19, NRPRZ
	DW	ADRES
	ENDM	

Opis:

Ekstrakod INSTP powoduje zainstalowanie w systemie programu umieszczonego w pamięci począwszy od adresu określonego etykietą użytkownika ADRES jako programu obsługi przerwania nr NRPRZ. Parametr NRPRZ = 0 oznacza obsługę przerwania o najwyższym priorytecie, a odpowiednio NRPRZ=7 o najniższym priorytecie.

Przykład użycia:

INSTP 5, PRZER - zainstaluj program zaczynający się etykietą PRZER jako programu obsługi przerwania nr 5

3.20. Ekstrakod - zainstaluj przerwanie - INSTP

Definicja makroinstrukcji:

INSTP	MACRO	NRPRZ, ADRES
	RST	3
	DB	19, NRPRZ
	DW	ADRES
	ENDM	

Opis:

Ekstrakod INSTP powoduje zainstalowanie w systemie programu umieszczonego w pamięci począwszy od adresu określonego etykietą użytkownika ADRES jako programu obsługi przerwania nr NRPRZ. Parametr NRPRZ = 0 oznacza obsługę przerwania o najwyższym priorytecie, a odpowiednio NRPRZ = 7 o najniższym priorytecie.

Przykład użycia:

INSTP 5, PRZER - zainstaluj program zaczynający się etykietą PRZER jako programu obsługi przerwania nr 5

3.21. Ekstrakod - zainstaluj program sprawdzania warunku -INSTW

Definicja makroinstrukcji:

INSTW	MACRO	NRWAR, ADRES
	RST	3
	DB	20, NRWAR
	DW	ADRES
	ENDM	

Opis:

Ekstrakod INSTW powoduje zainstalowanie w systemie programu umieszczonego w pamięci począwszy od adresu określonego etykietą użytkownika ADRES jako programu nr NRWAR sprawdzania spełnienia warunku.

Przykład użycia:



INSTW 2,SPRAW - zainstaluj program zaczynający się etykietą SPRAW jako program nr 2 sprawdzający speżnienie warunku.

#### 4. Eksploatacja programu monitor operatorski

##### 4.1. Opis ogólny monitora operatorskiego

Program MONITOR MM-80 jest to monitor operatorski opracowany dla mikrokomputera MM-80 przeznaczony do pracy pod systemem operacyjnym RTMP. Program ten jest przeznaczony do przygotowywania i uruchamiania zadań użytkowych. MONITOR MM-80 wykonuje typowe dla takich programów polecenia operatora dotyczące pamięci operacyjnej, dziurkarki, monitora ekranowego, taśmy perforowanej.

MONITOR MM-80 jest podłączony do systemu RTMP jako jedno z zadań o numerze określonym podczas operacji danej wersji systemu. Sterowanie programem monitora możliwe jest tylko z podstawowego urządzenia operatorskiego /o numerze  $\emptyset$  / i może odbywać się w czasie rzeczywistym podczas pracy na obiekcie. Aby program MONITOR MM-80 mógł rozpocząć pracę, urządzenie nr  $\emptyset$  nie może być zajęte przez żadne z innych zadań. Gdy urządzenie to jest wolne, program monitor można wywołać znakiem BTX /CTL,C/. Należy pamiętać, że przez cały czas pracy zadania MONITOR urządzenie to jest zajęte i żadne inne zadanie nie może korzystać z tego urządzenia. Dopuszczone będą tylko wydruki alarmowe /wywoływane ekstrakodem ALARM/.

Po wywołaniu monitor zgłasza się drukując na urządzeniu nr  $\emptyset$  tekst:

MONITOR MM-80

>

Znak ">" oznacza oczekiwanie programu na podanie przez operatora jednej z dyrektyw wraz z jej parametrami. Zbiór możliwych do wyboru przez operatora dyrektyw stanowią znaki: B, D, F, G, H, J, K, L, M, O, P, R, S, T, V, W, Y, Z

26

Ponadto, gdy była wcześniej użyta dyrektywa B możliwe jest użycie dyrektyw C i U.

Operator wybiera jedną z dyrektyw, podaje /jeśli są wymagane/ parametry, po czym kończy wprowadzanie danych znakiem powrotu karetki <CR>. Użycie samego znaku <CR> powoduje zwolnienie urządzenia nr 0, zakończenie programu MONITOR i powrót do koordynatora systemu.

Wykrycie błędu w dyrektywie /np. podanie pierwszego znaku nie oznaczającego żadnej z dyrektyw lub podanie w adresie cyfry nie hex/ sygnalizowane jest znakiem "?". Natomiast, gdyby operator chciał wpisywać dane lub ustawić pułapkę do obszaru nieistniejącej pamięci albo do pamięci typu PROM następuje wtedy sygnalizacja znakiem "!". Po zasygnalizowaniu błędu program MONITOR ponownie zgłasza swoją gotowość znakiem ">". Operator może zawsze, zanim zakończy wprowadzać dane znakiem powrotu karetki, wycofać ostatnio wprowadzony znak naciskając klawisz DEL /RUBOUT/, Usunięcie znaku sygnalizowane jest drukiem spacji.

znaku DEL powoduje usuwanie następnych,  
Kolejne naciskanie znaków w kolejności odwrotnej do ich wprowadzania.

#### 4.2. Opis dyrektyw monitora operatorskiego

W podanym niżej opisie poszczególnych dyrektyw programu MONITOR przyjęto następujące oznaczenia.

- ␣ - spacja, która koniecznie musi wystąpić między parametrami dyrektywy
- <CR> - znak powrotu karetki
- aaaa - cztery znaki hex określające adres
- pppp - cztery znaki hex określające adres początku obszaru pamięci

kkkk - cztery znaki hex określające adres końca obszaru  
pamięci

ww - dwa znaki hex określające wartość komórki

dd - dwie cyfry dziesiętne

dddd - cztery cyfry dziesiętne

Ponieważ jako podstawowe urządzenie operatorskie może być  
zastosowany zarówno monitor ekranowy jak i drukarka znakowo-  
mozaikowa, dlatego też używane w tekście <sup>terminy</sup> "drukuj" i "wyświetla"  
należy traktować zamiennie.

4.2.1. DYREKTYWA 0 - wyświetlenie i ewentualna zmiana zawarto-  
ści komórki pamięci

Wywołanie dyrektywy:

Oaaaa <CR>

gdzie: aaaa - adres komórki.

W odpowiedzi program MONITOR wyświetla zawartość tej komórki  
poprzedzonej jej adresem i oczekuje /bez sygnalizacji znakiem"/  
na dalsze dyspozycje operatora. Naciśnięcie znaku <CR> spowo-  
duje wyświetlenie zawartości komórki o ad<sup>e</sup>resie o 1 większym  
od poprzedniej, naciśnięcie znaków "-" i <CR> spowoduje wy-  
świetlenie zawartości komórki o <sup>adresie o</sup> 1 mniejszym niż poprzednio.  
Aby zakończyć dyrektywę należy podać znaki "." i <CR>, zgłosi  
się wtedy ponownie program MONITOR znakiem ">".

C h c ą c z m i e n i ć zawartość aktualnie wyświetlanej  
komórki, należy podać, bezpośrednio po jej dotychczasowej za-  
wartości, dwa znaki hex, tworzące jej nową zawartość. Następnie  
należy wybrać dalsze działanie dyrektywy w sposób identyczny  
jak podano wyżej tzn. odczyt następnej lub poprzedniej komórki  
albo zakończenie dyrektywy.

Program sprawdza, czy podana nowa zawartość komórki pamięci została poprawnie wpisana do pamięci i każdą niezgodność sygnalizuje znakiem "!". Użycie znaku nie będącego cyfrą hex w adresie lub w nowej zawartości komórki albo podanie znaku różnego od "-", ".", <CR> powoduje sygnalizację błędu znakiem "?", po czym następuje oczekiwanie na wprowadzenie przez operatora poprawnych danych.

#### 4.2.2. DYREKTYWA M - wyświetlanie zawartości obszaru pamięci.

Wywołanie dyrektywy.

Mpppp    kkkk <CR>

W odpowiedzi program drukuje nagłówek, który stanowią ostatnie cyfry adresu komórek pamięci, a następnie w wierszach po 16 komórek w jednym wierszu ich zawartości. Każdy wiersz jest poprzedzony adresem pierwszej komórki w danym wierszu.

Błędne wywołanie dyrektywy tj. użycie cyfry nie hex w którymkolwiek z adresów lub podanie adresu końca mniejszego od adresu początku powoduje sygnalizację błędu znakiem "?", po czym następuje zgłoszenie MONITORA znakiem ">". Podobnie program MONITOR zgłasza swoją gotowość po wyświetleniu zawartości całego obszaru z komórkami o adresach pppp i kkkk włącznie.

#### 4.2.3. DYREKTYWA V - zapis stałej wartości do obszaru pamięci . Wywołanie dyrektywy:

Vpppp    kkkk    ww <CR>

gdzie ww - wartość wpisywana do wszystkich komórek pamięci od adresu pppp do kkkk włącznie. Po zapełnieniu podaną wartością komórek pamięci dyrektywa sama kończy swoje działanie i powraca do programu MONITOR, który zgłasza swoją gotowość znakiem ">". Gdy w którymkolwiek z adresów lub w wartości

Jeżeli podczas podawania danych popełniony zostanie błąd /zostanie użyta cyfra nie hex/, to podczas analizy wprowadzonego <sup>tekstu</sup> błęd<sup>zignorowane</sup> będą ~~sygnalizowane~~ wszystkie przeznaczone do wpisania wartości, począwszy od pierwszej komórki, dla której wystąpił błąd aż do znaku <CR>. Następnie wydrukowany będzie w nowej linii adres pierwszej komórki wiersza, w którym został popełniony błąd. Jeżeli operator przystąpi do dalszego wprowadzania danych, to będą one zapisywane począwszy od komórki błędnie zapisanej.

Przed wpisaniem nowej wartości do komórki, sprawdzone jest, czy ta komórka należy do pamięci typu RAM. Jeżeli nie, to sygnalizowany jest błąd komunikatem.

aaaa !

gdzie aaaa - adres pierwszej, dla której stwierdzono błąd, po czym zgłasza się program MONITOR znakiem ">".

Wprowadzanie danych kończy się naciskając znak "." przed użyciem <CR>. Nastąpi wtedy powrót do programu MONITOR.

#### 4.2.5. DYREKTYWA R - przesunąć blok danych.

Wywołanie dyrektywy:

Rpppp kkkk aaaa <CR>

Dyrektywa pobiera zawartość komórek z obszarem pamięci określonego adresami pppp i kkkk włącznie i ładuje je do kolejnych komórek pamięci począwszy od adresu aaaa. Zakończenie dyrektywy powoduje powrót do programu MONITOR i zgłoszenie jego gotowości znakiem ">".

Sygnalizacja błędów jest taka sama jak w dyrektywie V.

#### 4.2.6. DYREKTYWA L - czytanie taśmy papierowej..

Wywołanie dyrektywy:

L <CR>

Dyrektywa służy do czytania taśmy papierowej i zapisu odczytanej informacji do pamięci RAM. Dyrektywa bada, czy znaki odczytane z taśmy mają parzystą ilość jedynek, czy znaki są heksadecymalne, czy jest poprawna suma kontrolna w rekordach, czy dane na taśmie papierowej mają właściwy format oraz czy informacja została poprawnie zapisana do pamięci. Gdy cała taśma zostanie poprawnie wczytana, drukowany jest komunikat:  
OK

po czym zgłosi się program MONITOR znakiem ">".

W przypadku, gdy zostanie stwierdzony jakikolwiek błąd nastąpi sygnalizacja komunikatem:

n BLAD CT

gdzie n cyfra od 1 do 6.

Jedynie gdy stwierdzona zostanie niezgodność informacji wpisanej do komórki pamięci z poprawnie odczytaną błąd, sygnalizowany jest komunikatem:

aaaa .. !

gdzie aaaa oznacza adres pierwszej komórki, dla której wystąpił błąd.

Po zasygnalizowaniu błędu dyrektywa kończy swoje działanie i zgłasza się znakiem ">". program MONITOR.

W komunikacie o błędzie

n = 1 oznacza, że wystąpił koniec taśmy zanim nastąpił ostatni rekord informacji lub czytnik został wyłączony z sieci w trakcie czytania informacji.

n = 2 oznacza, że odczytany został znak o nieprzystej liczbie bitów

n = 3 oznacza, że wystąpiła zła suma kontrolna w rekordzie

n = 4 oznacza, że w rekordzie wystąpił znak nie będący cyfrą hex

n = 5 oznacza, że stwierdzony zły typ rekordu/różny od zera dla rekordów informacyjnych/

n = 6 oznacza, że nastąpiło przekroczenie czasu, tzn. czytnik w ciągu 2 sek. nie odczytał żadnego znaku. Może to być spowodowane np. zacięciem się taśmy lub nie włączeniem zasilania czytnika

Dane na wczytywanej taśmie powinny mieć format zgodny z formatem Intellec Hex.

Istnieje możliwość wczytywania danych z taśmy papierowej do innego obszaru pamięci niż podany w rekordach na taśmie. W tym celu należy wywołać dyrektywę za pomocą komunikatu:

Laaaa < CR >

gdzie aaaa oznacza przesunięcie danych, tzn. jest to wartość dodawana do każdego adresu ładowanie informacji odczytanego z taśmy. Komunikaty są identyczne jak w dyrektywie L bez przesunięcia.

#### 4.2.7. DYREKTYWA S - sprawdzanie taśmy papierowej.

Wywołanie dyrektywy:

S < CR >

Dyrektywa porównuje informację zapisaną na taśmie papierowej z zawartością pamięci. Diagnostyka taśmy papierowej jest identyczna jak w dyrektywie L. Jedynie zamiast wpisywania informacji do pamięci porównywana jest wartość odczytana z taśmy z zawartością komórki pamięci i w przypadku niezgodności drukowany jest komunikat:

aaaa ... !

po czym zgłasza się znakiem ">" program MONITOR. aaaa oznacza



adres pierwszej komórki dla, której stwierdzono błąd. Pozostałe komunikaty o błędach są takie same jak w dyrektywie L. Gdy dla całej taśmy nie zostanie stwierdzony żaden błąd drukowany jest komunikat

OK

po czym zgłasza się program MONITOR.

Podobnie jak w dyrektywie L istnieje możliwość porównywania zawartości taśmy z pamięcią o adresach przesuniętych o wartość aaaa w stosunku do adresów ładowania na taśmie papierowej. W tym przypadku dyrektywę należy uruchomić podając tekst:

Saaaa <CR>

4.2.8. DYREKTYWA D - dziurkowanie taśmy papierowej.

Wywołanie dyrektywy:

a/ Dpppp  kkkk. <CR>

b/ lub Dpppp  kkkk <CR>

Dyrektywa służy do wyprowadzania zawartości pamięci na taśmę papierową. Informacja na taśmie jest dziurkowana zgodnie z formatem Intellec Hex.

Używa w wywołaniu dyrektywy, w którymkolwiek z adresów, cyfry nie hex lub podanie adresu początku wyprowadzanego obszaru większego od adresu końca powoduje sygnalizację błędu znakiem "?", i powrót do programu MONITOR, który zgłasza się znakiem ">". Podobne działanie nastąpi, gdy po ostatnim znaku określającym adres końca wyprowadzanego obszaru podany będzie znak różny od "." lub <CR>.

Jeżeli wywołanie dyrektywy nastąpiło poprzez podanie tekstu a/ to po wyprowadzeniu na taśmę zawartości obszaru określonego adresami pppp i kkkk, zostanie wydziurkowany rekord końca, a następnie pewien odcinek pustej taśmy, po czym zgłosi się znakiem ">" program MONITOR.

Jeżeli dyrektywa była wywołana tekstem b/ , to po wydziurkowaniu zawartości zadeklarowanego obszaru dyrektywa przechodzi do oczekiwania na podanie przez operatora w nowej linii dalszych dyspozycji. Operator ma do wyboru następujące możliwości:

c/ pppp kkkk <CR>

d/ pppp kkkk.<CR>

e/ . <CR>

Podanie tekstu c/ spowoduje wyprowadzenie na taśmę papierową kolejnego określonego adresami pppp i kkkk obszaru pamięci i ponowne oczekiwanie na dalsze dyspozycje operatora. Gdy został podany tekst d/, to po wydziurkowaniu zawartości zadeklarowanego adresami pppp i kkkk obszaru pamięci zostanie wyprowadzony rekord kończący dane na taśmie, następnie pewien odcinek pustej taśmy, po czym nastąpi zgłoszenie MONITORA znakiem " ". Podanie tekstu e/ spowoduje działanie jak wyżej, z tym że nie będzie wyprowadzona zawartość komórek pamięci.

Jeżeli, w którymkolwiek z tekstów c/, d/, e/ zostanie stwierdzony błąd, to nastąpi jego sygnalizacja znakiem "?", po czym dyrektywa oczekiwać będzie na poprawne dyspozycje operatora.

Jeżeli w trakcie dziurkowania taśmy zostanie stwierdzona niepoprawna praca dziurkarki, wyświetlany jest komunikat:

n BLAD DT

gdzie n = 1 oznacza, że dziurkarka jest nie włączona do sieci lub, że skończyła się taśma w dziurkarce

n = 2 oznacza przekroczenie czasu tzn. , że dziurkarka nie zgłosiła gotowości przez ok. 3 sek.

Po wyświetleniu powyższego komunikatu następuje zakończenie dyrektywy i zgłasza się program MONITOR.

#### 4.2.9. DYREKTYWA K - kopiowanie taśmy

Wywołanie dyrektywy:

K<CR>

Dyrektywa służy do kopiowania taśmy o dowolnym temacie. Po wywołaniu dyrektywy taśma czytana przez czytnik będzie kopiowana na dziurkarce aż do końca taśmy w czytniku, a następnie zgłosi się znakiem ">" program MONITOR. W razie stwierdzenia niepoprawnej pracy czytnika lub dziurkarki zostanie wydrukowany komunikat:

n BLAD CT      gdzie n = 1 lub 6

lub            n BLAD DT      gdzie n = 1 lub 2

po czym wykonywanie dyrektywy zostanie zakończone i zgłosi się program MONITOR. Znaczenie błędów jest takie samo jak podane w opisie dyrektyw L i D.

#### 4.2.10. DYREKTYWA W - uruchomienie programu

Wywołanie dyrektywy:

Waaaa <CR>

gdzie aaaa adres startu uruchamianego programu. Jeżeli w adresie będzie użyty znak nie będący cyfrą heksadecymalną, to wydrukowany zostanie znak "?" i nastąpi powrót do MONITORA.

Użycie dyrektywy powoduje, że system pracuje w modzie jednozadaniowym tzn. wykonywany jest tylko uruchomiony przez tę dyrektywę program.

Żadne inne zadania nie będą uruchamiane, ani wznowione jeżeli nie zostanie zakończony uruchomiony program. W programie tym mogą być używane wszystkie ekstrakody systemowe i musi się on kończyć skokiem do okresu określonego etykietą systemową KOZAD.

#### 4.2.11. DYREKTYWA I - instalowanie zadania

Wywołanie dyrektywy:

Idd aaaa <CR>

Dyrektywa powoduje przypisanie zadaniu o numerze określonym parametrem dd adresu startu aaaa. Błąd jest sygnalizowany znakiem "?" gdy w numerze instalowanego zadania zostanie użyty znak nie będący cyfrą dziesiętną lub w adresie startu zadania użyty będzie znak nie będący cyfrą heksadecymalną. Po zasygnalizowaniu błędu nastąpi powrót do programu MONITOR. Nieznaczące zero w numerze instalowanego zadania można pominąć.

Po prawidłowym wykonaniu dyrektywy zgłasza się znakiem ">" program MONITOR.

#### 4.2.12. DYREKTYWA F - usunięcie zadania

Wywołanie dyrektywy:

Fdd <CR>

gdzie dd - dwie cyfry dziesiętne określające nr zadania. Nieznaczące zero można pominąć. Użycie znaku różnego od cyfry dziesiętnej powoduje sygnalizację błędu znakiem "?" i powrót do MONITORA.

Dyrektywa powoduje usunięcie zadania ze wszystkich tablic i kolejek systemowych, zwalnia urządzenie zajęte przez to zadanie.

Jeżeli usunięcie zadania aktualnie jest zawieszona, to nie zostanie ono już wznowione. Po wykonaniu dyrektywy zgłasza się <sup>znakiem</sup> ">" program MONITOR.

#### 4.2.13. DYREKTYWA Y - wydruk adresów startów zadań.

Wywołanie dyrektywy:

Y <CR>

Użycie dyrektywy powoduje wyświetlenie w kolejnych wierszach adresów startu wszystkich zadań w systemie. Po zakończeniu działania następuje powrót do programu MONITOR.

#### 4.2.14. DYREKTYWA G - uruchomienie zadania

Wywołanie dyrektywy:

Gdd <CR>

Dyrektywa powoduje zgłoszenie do wykonania zadania o numerze określonym parametrem dd. Muszą to być dwie cyfry dziesiętne. /nieznaczące zero można pominąć/. Jeżeli, któryś ze znaków nie jest cyfrą dziesiętną następuje sygnalizacja błędu znakiem "?" i powrót do programu MONITOR.

W przypadku poprawnego wywołania, po dopisaniu zadania do kolejki zadań oczekujących na uruchomienie następuje zwolnienie urządzenia nr 0 przez zadanie MONITOR i zakończenie tego zadania.

#### 4.2.15. DYREKTYWA P - włączenie zadania okresowego.

Wywołanie dyrektywy:

Pdd □ dddd <CR>

Dyrektywa powoduje zgłoszenie zadania o numerze dd do wykonywania okresowo co dddd Pierwsze uruchomienie sekund. zadania nastąpi po upływie dddd sekund od momentu zakończenia dyrektywy. Jeżeli zadanie było wcześniej zgłoszone do wykonywania okresowo, to użycie dyrektywy P spowoduje zmianę okresu wykonywania zadania.

Użycie w wywołaniu dyrektywy znaku nie będącego cyfrą dziesiętną powoduje sygnalizację błędu znakiem "?" i powrót do programu MONITOR.

Nieznaczące zero w numerze uruchamianego zadania można pominąć.

#### 4.2.16. DYREKTYWA Q - wyłączenie zadania okresowego.

Wywołanie dyrektywy:

Qdd <CR>

Dyrektywa powoduje usunięcie z tablic systemowych zadania o numerze

dd zgłoszonego wcześniej do okresowego wykonywania. Jeżeli zadanie o tym numerze nie było zgłoszone jako okresowe, to wywołanie dyrektywy Q nie powoduje żadnego działania. Gdy w momencie wywołania dyrektywy usuwane zadanie jest w trakcie wykonywania /zawieszona/, to wykona się ono do końca.

Po zakończeniu działania dyrektywy następuje zgłoszenie MONITORA znakiem ">". Użycie w wywołaniu znaku nie reprezentującego cyfry dziesiętnej, powoduje sygnalizację błędu znakiem "?" i powrót do programu MONITOR.

Nieznaczące zero w <sup>numerze</sup> ~~momencie~~ wyłączonego zadania można pominąć.

#### 4.2.17. DYREKTYWA H - włączenie zadania o godzinie.

Wywołanie dyrektywy:

Hdd,dd.dd <CR>

Dyrektywa powoduje zgłoszenie do wykonania zadania o numerze dd, gdy w licznikach systemowych czasu rzeczywistego pojawi się godzina dd.dd /pierwsze dwa znaki określają godzinę, następne minuty/.

Wszystkie cyfry muszą być dziesiętne, a godziny i minuty muszą być rozdzielone znakiem kropki ".". Jeżeli tak nie jest, to następuje sygnalizacja błędu znakiem "?" i powrót do MONITORA.

Sygnalizacja błędu znakiem "?" nastąpi również, gdy podana zostanie godzina większa niż 23 i minuty większe niż 59.

Nieznaczące zero w numerze włączanego zadania można pominąć.

Po wykonaniu dyrektywy nastąpi powrót do MONITORA.

#### 4.2.18. DYREKTYWA T - zmiana czasu astronomicznego.

Wywołanie dyrektywy:

Tdd.dd <CR>

Dyrektywa powoduje wpisanie do systemowych liczników czasu rzeczywistego wartość podanych przez operatora w jej parametrach.

Godziny i minuty muszą być rozdzielone kropką. Mogą być używane tylko cyfry dziesiętne, a wielkość godzin musi zawierać się w przedziale 0-23, a minut 0-59. Każdy błąd sygnalizowany jest znakiem "?" i powrotem do programu MONITOR.

Prawidłowe działanie dyrektywy kończy się zgłoszeniem przez program MONITOR gotowości znakiem ">".

#### 4.2.19. DYREKTYWA B - ustawienie pułapki /break-point/.

Wywołanie dyrektywy:

Baaaa <CR>

Dyrektywa ta służy do ustawiania pułapki w programie użytkownika umieszczonym w pamięci RAM.

Po ustawieniu pułapki należy spowodować wykonywanie programu np. używając dyrektywy G. Jeżeli w trakcie wykonywania programu dojdzie do adresu ustawionego przez dyrektywę B, zostanie on przerwany i na urządzeniu nr 0 będzie wyświetlona zawartość wszystkich rejestrów i wskaźników procesora w następującym formacie:

PC A B C D E H L SP S Z X AC X P X CY

Program wraca do monitora, który zgłasza gotowość znakiem ">".

Operator ma do dyspozycji wszystkie dyrektywy oraz dodatkowo dyrektywy C i U. Użycie teraz ponownie dyrektywy B /wywołanie identyczne jak poprzednio/ z innym adresem umieszczenia pułapki powoduje skasowanie starej pułapki, ustawienie nowej i wznowienie wykonywania programu od przerwanej miejsca.

Użycie przycisku RESET na pulpicie operatora powoduje usunięcie ostatnio ustawionej pułapki.

Podanie w adresie pułapki znaku nie reprezentującego cyfry heksadecymalnej powoduje sygnalizację błędu znakiem "?" i powrót do programu MONITOR.

Próba ustawienia pułapki dyrektywy B w pamięci PROM sygnalizowana jest znakiem "!", po czym ponownie zgłasza się program MONITOR.

4.2.20. DYREKTYWA U - usunięcie pułapki.

Wywołanie dyrektywy:

U <CR>

Użycie tej dyrektywy jest możliwe jedynie w przypadku, gdy wcześniej była użyta dyrektywa B. W przeciwnym przypadku sygnalizowany jest błąd znakiem "?" i następuje powrót do MONITORA. Dyrektywa U usuwa ostatnio ustawioną pułapkę i powoduje kontynuację przerwane programu.

4.2.21. DYREKTYWA C - kontynuacja programu bez przesunięcia pułapki.

Wywołanie dyrektywy:

C <CR>

Użycie tej dyrektywy jest możliwe jedynie w przypadku, gdy wcześniej była użyta dyrektywa B, a program został zatrzymany w miejscu ustawionej pułapki. Gdy tak nie jest, następuje sygnalizacja błędu znakiem "?" i zgłoszenie programu MONITOR.

Dyrektywa C służy do kontynuowania programu zatrzymanego w pułapce bez zmian miejsca ustawienia pułapki.



## 5. Procedury sprzężenia z pamięcią kasetową

Użytkownik ma możliwości wykorzystania w swoim oprogramowaniu następujących systemowych procedur sprzężenia z pamięcią kasetową:

- przewiń taśmę do położenia wyjściowego
- zapisz blok danych
- odczytaj blok danych
- wyszukaj blok danych
- zwolnij pamięć kasetową

Wszystkie te procedury działają w oparciu o systemową obsługę przerwania z pakietu sprzężenia z pamięcią kasetową.

Po zakończeniu działania procedury.

o poprawnym jej wykonaniu użytkownik jest informowany za pomocą flagi CY w słowie stanu procesora.

CY = 0 - oznacza poprawne wykonanie procedury

CY = 1 - oznacza niepoprawne wykonanie procedury

O konkretnej przyczynie niepoprawnego działania procedury

użytkownik może się dowiedzieć na podstawie komórki pamięci o adresie określonym ~~sytuowaną~~ <sup>systemową</sup> etykietą INPKØ. Znaczenie informacji zwrotnej zawartości w tej komórce jest następujące:

- 0 - brak informacji, procedura jeszcze nie wykonana do końca
- 1 - procedura wykonana poprawnie
- 2 - wystąpił koniec taśmy zanim blok został zapisany, odczytany lub wyszukany
- 3 - błąd przy zapisie lub odczycie /np. znak nie hex/
- 4 - blok został zapisany poprawnie i jest ostatnim blokiem na taśmie - koniec taśmy
- 5 - przy czytaniu bloku danych wystąpiło więcej bajtów niż zadeklarowano w wywołaniu procedury.

6 - zły wybór przewijaka

7 - pamięć nie włączono do sieci, nie przydzielono lub brak kasety

8 - w procedurze zapisu danych stwierdzono, że zapis danych na kasecie jest niedozwolony

80 - time-out tzn. dana procedura nie została zahamowana w czasie przewidzianym na jej wykonanie. /np. w wyniku zablokowania taśmy/

Wystąpienie błędu określonego w informacji zwrotnej przez 2 jest połączone z wyprowadzeniem na urządzenie nr 0 wydruku alarmowego:

KONIEC TASMY

Błąd określony w informacji zwrotnej przez 3 jest połączony z wydrukiem alarmowym:

BŁAD PRZY ZAPISIE

lub BŁAD PRZY ODCZYCIE

Błąd określony w informacji zwrotnej przez 6 jest połączony z wydrukiem alarmowym

ZŁY WYBOR PRZEWIJAKA

Błąd określony w informacji zwrotnej przez 7 jest połączony z jednym z poniższych wydruków alarmowych.

BRAK KASETY

PK NIE PRZYDZIELONA

BŁAD PK - gdy pamięci nie włączono do sieci

Błąd określony w informacji zwrotnej przez 8 jest połączony z wydrukiem alarmowym:

ZAPIS NIEDOZWOLONY

Natomiast wystąpienie błędu określonego w informacji ~~bardziej~~ <sup>zwrotnej</sup> przez 80 wiąże się z wydrukiem alarmowym.

PK NIE SPRAWNA

Przed pierwszą komunikacją z danym przewijakiem, należy jego numer /1 lub 2/ umieścić w komórce pamięci określonej systemową etykietą WYBOR. Procedury komunikacji z pamięcią kasetową nie zmieniają zawartości tej komórki.

Wpisanie innej wartości powoduje sygnalizację błędu określonego w informacji zwrotnej przez 6.

### 5.1. Procedura ustawiania taśmy w położeniu początkowym

Wywołanie procedury następuje poprzez użycie instrukcji

CALL REVND

gdzie: REVND - etykieta systemowa

W wyniku działania procedury taśma ustawiona jest w położeniu wyjściowym na pierwszym znaczniku końca taśmy /BET/.

### 5.2. Procedura zapisu bloku danych

Wywołanie procedury następuje poprzez użycie instrukcji:

CALL ZAPKØ

gdzie: ZAPKØ - etykieta systemowa

Przed wywołaniem procedury należy w rejestrach H,L ustawić adres początku, a w D,E adres końca zapisywanego obszaru.

Informacja jest zapisywana na taśmie w postaci bloku danych tzn. z przerwą między-blokową przed i po danych.

W trakcie zapisu informacji prowadzony jest kontrolny odczyt i następuje sprawdzanie, czy informacja jest poprawnie zapisywana. Błąd sygnalizowany jest w informacji zwrotnej wartością 3

### 5.3. Procedura odczytu bloku danych

Wywołanie procedury

CALL ODPKØ

gdzie: ODPKØ - etykieta systemowa

Przed wywołaniem procedury należy w rejestrach H,L ustawić

43

adres początku obszaru, do którego mają być zapisywane dane z pamięci kasetowej, a w rejestrach D,E maksymalną liczbę bajtów, jaką może mieć odczytywany blok.

Blok danych z taśmy dłuższy od zadeklarowanej w D,E wielkości sygnalizowany jest jako błąd i w informacji zwrotnej ustawiona jest wartość 5.

#### 5.4. Procedura wyszukania bloku danych

Wywołanie procedury

CALL BLOKØ

gdzie: BLOKØ - etykieta systemowa

Przed wywołaniem należy ustawić w rejestrach H,L liczbę przerw międzyblokowych zliczonych od aktualnego położenia taśmy przy przewijaniu w przód, po których taśma ma zatrzymać się.

#### 5.5. Procedura zwolnienia pamięci kasetowej

Wywołanie procedury:

CALL ZWPKØ

gdzie: ZWPKØ - etykieta systemowa

Użycie procedury powoduje zwolnienie pamięci kasetowej i umożliwia dostęp do niej innym zadaniom. Procedura powoduje też zwolnienie blokady kasety, co pozwala operatorowi na wyjęcie kasety z PK.

#### 6. Podprogramy systemowe dostępne dla użytkownika

W opisie omawianych podprogramów, jako adresy startów podawane są etykiety systemowe. Wszystkie podprogramy / z wyjątkiem RST 4, RST 5 i RST 6/wywoływane są instrukcją CALL. Podprogramy oznaczone<sup>x</sup> są dostępne dla użytkownika tylko razem z programem MONITOR.

44

E7 RST-4 - podprogram opóźnienia programowego. Przed wejściem do podprogramu należy umieścić w rejestrze B jednostkę opóźnienia /1 dla jednostki 1 ms i wartość różną od 1 dla jednostki 100 ms/, a w rejestrze C ilość jednostek. Wyjście z RST4 następuje po zadanym w rejestrach B,C czasie. Podprogram nie zmienia zawartości rejestrów D,E,H,L.

Przykład : wykonanie ciągu instrukcji:

ØC MVI B,80H ; jednostka 100ms

LXI B → 01H

ØE MVI C,1AH ; 26 jednostek

E7 RST 4

powoduje opóźnienie programowe 2,6 sekundy

EF RST 5 - podprogram czytania pakietu 16-bitowego.

Przed wywołaniem podprogramu w parze rejestrów H,L należy umieścić adres pakietu. Po wyjściu z podprogramu informacja odczytana z pakietu znajduje się w parze rejestrów D,E. Podprogram nie zmienia zawartości rejestrów B,C,H,L.

F7 RST 6 - podprogram pisania do pakietu 16-bitowego

Przed wywołaniem podprogramu w parze rejestrów H,L należy umieścić adres pakietu, a w parze rejestrów D,E - wpisywaną informację. Podprogram nie zmienia zawartości rejestrów B,C,D,E,H,L.

SUMA - dodawanie zawartości akumulatora do pary rejestrów H,L. Podprogram nie zawiera zawartości B,C,D,E.

<sup>191E</sup>  
HXAS1 - zamiana 4 młodszych bitów akumulatora na kod ASCII cyfry heksadecymalnej w akumulatorze. Rejestry niezmienniane B,C,D,E,H,L.

<sup>1928</sup>  
HXAS2 - zamiana zawartości akumulatora na kody ASCII dwóch cyfr heksadecymalnych w pamięci.

Przed wywołaniem podprogramu, w parze rejestrów B,C należy

umieścić adres pamięci RAM, do której mają być wpisane kody ASCII.

Do pierwszej komórki pamięci wpisywany jest kod ASCII 4 starszych bitów akumulatora, a do następnej komórki wpisywany jest kod ASCII 4 młodszych bitów. Na wyjściu z podprogramu zawartość pary rejestrów B,C jest zwiększona o 2. Rejestry niezmienniane D,E, H,L.

<sup>19 3B</sup>  
ASHX - zamiana cyfry heksadecymalnej w kodzie ASCII z akumulatora na wartość binarną w akumulatorze.

Jeżeli na wejściu do podprogramu nie znajduje się w akumulatorze kod ASCII cyfry heksadecymalnej, to sygnalizowany jest błąd poprzez ustawianie flagi CY=1 w słowie stanu procesora. W ~~pierwszym~~ <sup>przeciwnym</sup> przypadku CY=0. Podprogram nie zmienia zawartości rejestrów B,C,D,E,H,L.

<sup>7AB3</sup>  
DWHX - zamiana dwóch cyfr heksadecymalnych w kodach ASCII z kolejnych komórek pamięci na liczbę binarną w akumulatorze.

Na wejściu podprogramu, w parze rejestrów B,C należy ustawić adres pierwszej komórki pamięci. Jeżeli, któryś ze znaków ASCII nie reprezentuje cyfry heksadecymalnej, to CY≠1.

W przeciwnym przypadku CY=0, a zawartość pary rejestrów B,C jest zwiększona o 2. Rejestry niezmienniane: D,H,L.

<sup>7AC6</sup>  
CZHX - zamiana czterech cyfr heksadecymalnych o kodach ASCII z kolejnych komórek pamięci na dwie liczby binarne w rejestrach D,E.

Na wejściu podprogramu, w parze rejestrów B,C należy ustawić adres pierwszej komórki pamięci.

Jeżeli którykolwiek ze znaków ASCII nie reprezentuje cyfry heksadecymalnej, to CY=1. W przeciwnym przypadku CY=0, a zawartość pary rejestrów B,C jest zwiększona o 4. Rejestry niezmienniane : H,L.

<sup>7AF3</sup>  
xDZHEX - zamiana dziesiętnej cyfry w kodzie ASCII z komórki pamięci na wartość binarną w akumulatorze. Przed wywołaniem podprogramu w parze rejestrów B,C należy ustawić adres komórki pamięci, w której znajduje się kod ASCII zamienianej cyfry. Jeżeli zawartość komórki pamięci nie reprezentuje cyfry dziesiętnej, to następuje sygnalizacja poprzez ustawienie flagi CY=1 w słowie stanu procesora. Rejestry niezmienniane: B,C,D,E, H,L.

<sup>7B13</sup>  
xDWDZ - zamiana jednej lub dwóch cyfr dziesiętnych w kodzie ASCII z kolejnych komórek pamięci, na wartość binarną w akumulatorze.

Przed wywołaniem podprogramu w parze rejestrów B,C należy ustawić adres pierwszej komórki pamięci. Jeżeli jej zawartość nie reprezentuje cyfry dziesiętnej, to CY=1, a zawartość B,C jest zwiększona o 1.

Jeżeli zawartość drugiej komórki pamięci nie reprezentuje cyfry dziesiętnej, to CY=0, a B,C jest zwiększona o 1

/w stosunku do wartości na wejściu podprogramu/. Jeżeli w obu komórkach pamięci są cyfry dziesiętne, to CY=0, a B,C jest zwiększane o 2. Rejestry niezmienniane: H,L.

<sup>7B19</sup>  
xDZBCD - zamiana dwóch cyfr dziesiętnych w kodzie ASCII z kolejnych komórek pamięci na liczbę binarną w kodzie BCD w akumulatorze.

Działanie podprogramu jest analogiczne do DWHX. Rejestry niezmienniane: H,L.

<sup>7B2D156F</sup>  
xPOLI - porównanie zawartości pary rejestrów D,E z zawartością pary rejestrów H,L.

Wynik porównania sygnalizowany jest na flagach CY i Z w słowie stanu procesora:

gdy D,E = H,L to CY=0 Z=1

gdy D,E < H,L to CY=1 Z=0

gdy D,E > H,L to CY=0 Z=0

Rejestry niezmienniane: B,C,D,E,H,L.

## 7. Konfiguracja sprzętowa

System RTMT przeznaczony jest do pracy na mikrokomputerze MM80 i znajduje się w wewnętrznej pamięci EPROM pakietu jednostki centralnej. Obszarem roboczym systemu jest część wewnętrznej pamięci RAM pakietu jednostki centralnej.

Do pakietu procesora można dołączyć:

- do złącza C dowolne urządzenie o interfejsie V-24 stanowiące podstawowe urządzenie operatorskie systemu. Urządzeniu temu przyporządkowany jest zawsze numer logiczny 0.

- do złącza D czytnik taśmy papierowej CT2100

- do złącza E dziurkarka taśmy papierowej DT105S.

Dodatkowo system może obsługiwać inne urządzenia podłączone do zestawu mikroprocesorowego poprzez pakiety sprzężenia:

- MI24 obsługującego dwa niezależne urządzenia o interfejsach V-24

- MI50 obsługującego pamięć kasetową PK-1

## 8. Generacja systemu

W celu dostosowania systemu do konkretnej aplikacji oraz w celu minimalizacji zajętości pamięci i przyspieszenia działania możliwe jest generowanie systemu na żądanie użytkownika.

Na tym etapie konieczne jest określenie wartości pewnych parametrów, które stanowią informację wejściową do generacji systemu.



Określić trzeba konfigurację sprzętową oraz:

1. Ilość zadań obsługiwanych przez system wliczając w nią zadania monitora operatorskiego i inicjacji oprogramowania użytkownika.
2. Maksymalną ilość tekstów drukowanych w trybie alarmowym jednocześnie oczekujących na wyprowadzenie na urządzenie O /długość kolejki wydruków alarmowych/.
3. Maksymalną ilość tekstów jednocześnie oczekujących na wyprowadzenie na urządzenie O /długość kolejki wydruków urządzenia O/.
4. Długość kolejek wydruków na kolejne urządzenia o interfejsie V-24 /o ile istnieją w systemie/.
5. Ilość warunków zawieszenia zadań ekstrakodem ZAWWA.
6. Ilość warunków zawieszenia zadań ekstrakodem ZAWPR.
7. Długość bufora dla tekstów wprowadzanych z urządzeń o interfejsie V-24.
8. Numer i adres startowy zadania monitor operatorski.
9. Numery przerwania wykorzystywanych przez system.
10. Adresy pakietów MI24 obsługujących urządzenia o interfejsie V-24 stanowiących urządzenia nr 1,2 itd.
11. Adres pakietu MI50 obsługującego pamięć kasetową.
12. Adres zadania nr 1 inicjującego oprogramowanie użytkowe /o ile istnieje/.

Zajętość pamięci EPROM i RAM pakietu jednostki centralnej MM80 przez system operacyjny i program monitora operatorskiego zależy od wygenerowanej wersji oprogramowania podstawowego.

System obsługujący 31 zadań użytkownika, zawierający obsługę wszystkich ekstrakodów systemowych, procedury sprzężenia z pamięcią kasetową, obsługę 3 urządzeń w interfejsie V-24,

obsługę czytnika i dziurkarki zajmuje wraz z programem monitora operatorskiego nie więcej niż 8k pamięci EPROM i 2k pamięci RAM. Przy zastosowaniu pamięci typu 2716 całość oprogramowania systemowego mieści się na pakiecie jednostki centralnej MM80.