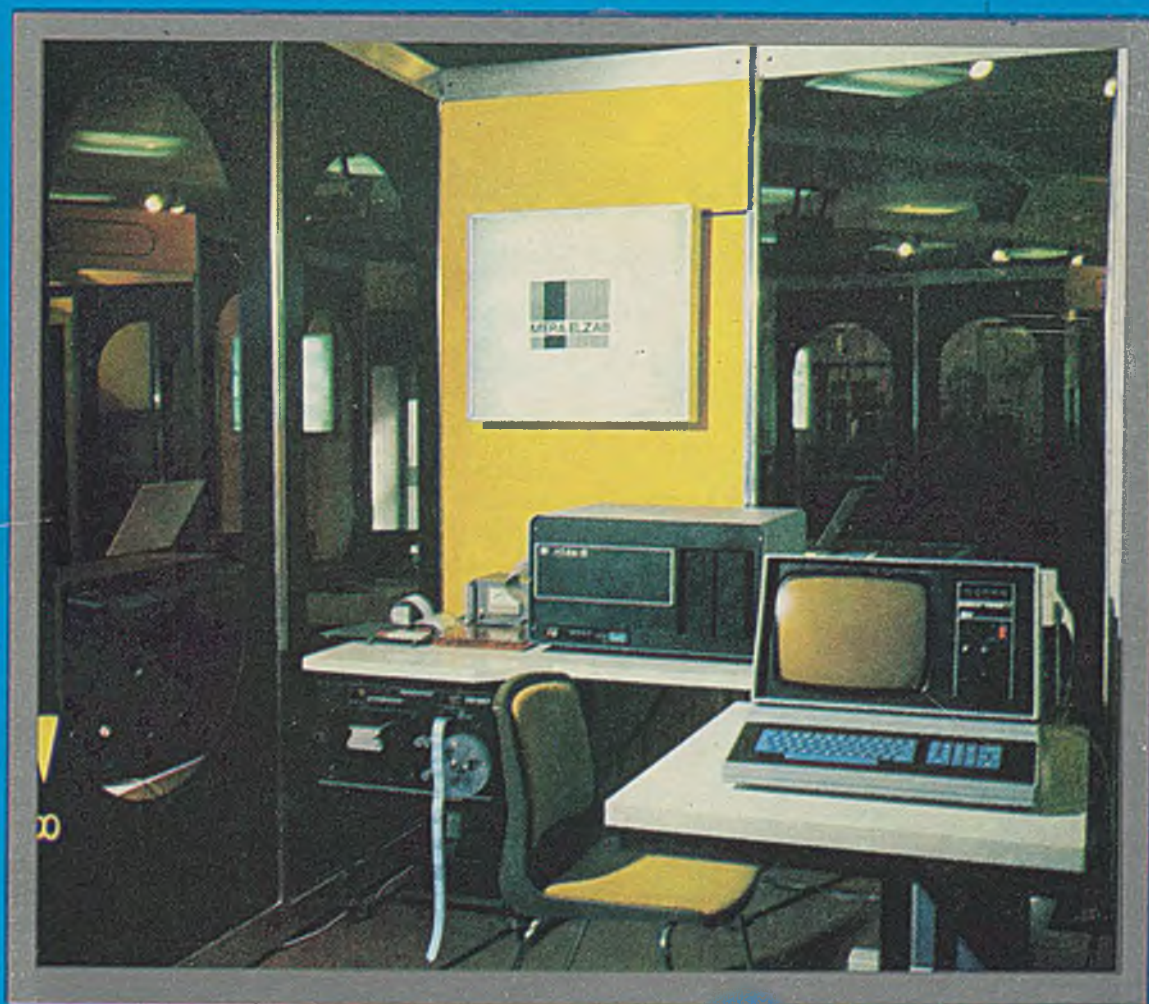


CHARAKTERYSTYKA SYSTEMU

RTDS-8



ZAKŁADY
URZĄDZEŃ
KOMPUTEROWYCH



SPIS TREŚCI:

1. Wstęp
2. Konfiguracja urzędziowa systemu
3. Uniwersalny emulator mikroprocesorów 8 – bitowych
4. Oprogramowanie systemu
5. Charakterystyka funkcjonalna emulatora
6. Specyfikacja systemu

Konstruktorzy systemu RTDS – 8

POLSKA AKADEMIA NAUK
Zakład Systemów Automatyki
Komputerowej,
Zespół Mikroprogramowania
44 – 100 GLIWICE,
ul. Bałtycka 5
tel. 31 – 08 – 11

Zakłady Urządzeń Komputerowych
„ Mera – Elzab ”

41 – 813 ZABRZE
ul. Kruczkowskiego 19

tel. 72 – 20 – 21

Oferują:

- projektowanie sond emulacyjnych i oprogramowania sterującego emulacją dla zadanych mikroprocesorów 8 – bitowych;
- dołączenie RTDS – 8 jako inteligentnego terminala systemów komputerowych ze skrośnym oprogramowaniem rozwojowym;
- opracowanie modułów specjalnych RTDS – 8: pamięci kasetowej, wyświetlacza graficznego, drukarki graficznej, analizatora stanów logicznych itp.;
- konsultacje z zakresu projektowania systemów mikrokomputerowych wspomaganego systemu RTDS – 8.

1. WSTĘP.

Wprowadzenie układów dużej skali integracji, a głównie mikroprocesorów i współpracujących z nimi układów programowych stworzyło nowe możliwości i nowe problemy. Między innymi pojawiły się problemy z uruchomieniem i testowaniem systemów aplikacyjnych zawierających tak złożone układy. Efektywne projektowanie i uruchomienie układów i systemów mikroprocesorowych wymaga stosowania specjalizowanych środków sprzętowych i programowych, zwanych systemami wspomagającymi, ułatwiający procesy projektowania i uruchamiania. Przed środkami wspomagającymi budowę systemów stawia się wymagania wynikające z:

- projektowania, uruchamiania i testowania części urządzeniowej,
- projektowania, uruchamiania, testowania i weryfikacji oprogramowania,
- integracji urządzeniowo – programowej uruchomionego systemu.

Stopień spełnienia tych wymagań może stanowić kryterium klasyfikacji systemów wspomagających. Systemy te podzielimy na trzy podstawowe klasy:

- proste zestawy mikrokomputerowe,
- system dla rozwoju oprogramowania i prowadzenia prac modelowych w konfiguracji systemu wspomagającego,
- systemy wspomagające konstrukcję oprogramowania i integrację urządzeniowo – programową systemu uruchamianego w oparciu o emulację układową.

Prezentowany system RTDS – 8 / Real Time Development System for 8 – bit mikroprocesors / należy do trzeciej klasy systemów wspomagających. Jest on modularnym systemem wspomagającym projektowanie i uruchamianie wyposażonym w uniwersalny emulator układowy mikroprocesorów 8 – bitowych. Idea emulacji układowej polega na symulowaniu mikroprocesora w uruchamianym systemie przez układ, który zawiera analogiczny mikroprocesor, otoczony logiką umożliwiającą jego sterowanie i monitorowanie. W programie integracji systemu budowanego element mikroprocesora zastępuje się sondą, układu emulatora. Umożliwia to uruchamianie systemu mikroprocesorowego w:

- docelowej konfiguracji,
- rzeczywistych warunkach pracy,
- bez konieczności wprowadzania do struktury uruchamianej specjalnych środków potrzebnych jedynie w procesie uruchamiania.

2. KONFIGURACJA URZĄDZENIOWA SYSTEMU.

System RTDS – 8 składa się z dwóch podstawowych bloków funkcjonalnych:

- mikrokomputera bazowego
- uniwersalnego emulatora mikroprocesorów 8 – bitowych / Rys. 1 /.

Mikrokomputer bazowy to dwupakietowy / pakiet procesora i pakiet interfejsów / uniwersalny system minikomputerowy wyposażony w procesor 8085 i zawierający w podstawowej wersji 16 kB pamięci typu RAM oraz 8 kB pamięci EPROM.

Pakiet interfejsów pozwala na współpracę systemu z:

- monitorem ekranowym TTY – CRT / MERA – 7952 / – konsolą systemową,
- jednostką pamięci na dyskach elastycznych / PL x 45D / – pamięć masowa systemu
- drukarką mozaikową / DZM – 180 /,
- czytnikiem taśmy papierowej / CT – 2200 /,
- dziurkarką taśmy papierowej / DT – 105S /, / czytnik i dziurkarkę można zastąpić np. stacją SPTP – 3 /,
- programatorem pamięci EPROM typu 2716,
- systemem komputerowym ze skrótnym oprogramowaniem rozwojowym; poprzez łącze szeregowe.

Uniwersalny emulator układowy mikroprocesorów 8 – bitowych, to układ oparty na oryginalnej metodzie forsowanego wykonania instrukcji. Forsowane wykonanie instrukcji jest podstawą realizacji funkcji oddziaływania na stan zasobów, sterowania stanem mikroprocesora i monitorowania stanu uruchamianego systemu. Dla mikrokomputera bazowego pakiety modułu emulatora stanowią układy peryferyjne o określonej strukturze portów wyjścia i pamięci danych. Uniwersalność emulatora systemu RTDS – 8 wynika z programowego sterowania przebiegiem emulacji na poziomie cykli rozkazowych oraz skonfigurowania elementów składowych emulatora wokół wyodrębnionej magistrali / mag. P /. Magistrala emulatora posiada cechy szerokiej klasy magistral systemów mikroprocesorowych, a równocześnie zawiera specjalne sygnały statusów i sterowania, wymagane przy sterowaniu emulacji mikroprocesora, oraz współpracy elementów składowych emulatora. W wyniku tego emulator systemu

RTDS – 8 posiada stałą strukturę urządzeniową, a emulacja żądanego mikroprocesora wymaga przyłączenia właściwej sondy emulującej oraz zainicjowania w systemie RTDS – 8 odpowiedniego programu sterującego.

3. UNIWERSALNY EMULATOR MIKROPROCESORÓW 8 – BITOWYCH.

Ogólną strukturę emulatora przedstawiono także na rys. 1. Podstawowymi modułami emulatora są: sonda emulowanego mikroprocesora i adapter.

Moduł sondy jest układem o konstrukcji zależnej od typu mikroprocesora uruchamianego systemu i zawiera sam mikroprocesor wraz z logiką realizującą jego forsowane sterowanie. Moduł ten emuluje sygnały mikroprocesora wyprowadzając je bezpośrednio do podstawki mikroprocesora w uruchamianym systemie. Sonda wypracowuje również sygnały magistrali S, której organizacja umożliwia realizację elementarnych zadań procesu uruchamiania. **Moduł adaptera** jest układem realizującym przejście z magistrali S na buforową magistralę systemu wspomagającego / magistralę P / o organizacji umożliwiającej współpracę z pozostałymi modułami emulatora. Do magistrali P przyłączone są pozostałe moduły emulatora. Są to moduły emulacji pamięci, komparacji stanów i śladowania. **Moduł emulacji pamięci** zawiera 16 kB pamięci RAM zorganizowanej w 4 bloki po 4 kB. Bloki te mogą być dołączone do systemu uruchamianego po zadeklarowaniu adresu początkowego i typu dostępu do bloku / read – only, read – write /. Pamięć emulowana może być umieszczona w obszarze adresowym, w którym istnieje już układowo pamięć uruchamianego systemu. Operacje w takim obszarze adresowym będą wykonywane tylko na pamięci emulowanej. Mechanizm ten pozwala np. na szybkie testowanie poprawek programu umieszczonego już w pamięci typu ROM budowanego systemu. Można wtedy przepisać program z ROM –u do pamięci emulowanej, dokonać poprawek i przystąpić do testowania programu. Z punktu widzenia uruchamianego systemu jego pamięć będzie zawierała poprawiony program. W systemie RTDS – 8 możliwe jest również wykorzystanie pamięci emulowanej jako rozszerzenie PaO systemu. Rozszerzenie PaO o pamięć emulowaną i powrót do konfiguracji podstawowej dokonuje się poprzez elementarne zlecenia programu MONITORA systemowego.

Moduł komparatora stanów zawiera dwa analogiczne układy rozpoznające żądany stan magistrali adresowej w określonym cyklu maszynowym. Układy te pozwalają na realizację braków programu wykonywanego w czasie rzeczywistym w trybie emulacji oraz na warunkowe określenie trybu śladowania przebiegu programu.

Moduł śladowania przebiegu programu w czasie rzeczywistym pozwala na warunkowe zapamiętanie stanu 1024 cykli maszynowych. Informacje w ramach pojedynczego cyklu zapamiętane są na 32 bitach obejmujących: adres / 16 /, dane / 8 /, rodzaj cyklu / 4 / oraz cztery warunki wybrane przez użytkownika w dowolnym punkcie systemu uruchomionego. Deklaracja warunków śladowania i startu pozwala na selektywne śladowanie cykli.

4. OPROGRAMOWANIE SYSTEMU.

Oprogramowanie podstawowe systemu RTDS – 8 stanowi:

- MONITOR systemowy,
- DYSKOWY SYSTEM OPERACYJNY z biblioteką programów.

MONITOR systemowy jest umieszczony w pamięci typu ROM. Posiada standardowe dla tego typu programu możliwości, rozszerzone o oprogramowanie sterujące pamięcią dyskową oraz zlecenia: testowania pamięci operacyjnej, ładowania systemu operacyjnego, formatowanie dyskietek, programowania pamięci EPROM / 2716 /, ekspansji PaO o pamięć emulowaną.

DYSKOWY SYSTEM OPERACYJNY jest jednoprogramowym systemem kompatybilnym z CP / M wersja 1.4. W podstawowej konfiguracji systemu RTDS – 8 może pracować CP / M w wersji 16 kB, oraz przy pracy bez emulatora – systemy do 32 kB. Ekspansję pamięci osiąga się przez dołączenie, jako rozszerzenia PaO, modułu pamięci emulowanej z bloku emulatora / 16 kB /. Biblioteka programów użytkowych systemu zawiera:

- edytor tekstu / ED /,
- assembler dla mikroprocesorów 8080 / 8085 / ASM /,
- debugger / DDT /,
- systemowe programy organizacyjne / STAT, PIP, LOAD, DUMP, COPY, MOVCPM, SYSGEN, SUBMIT / oraz:

Interpreter języka BASIC / TBASIC – zajmuje 4 kB pamięci i współpracuje z CP / M – 16 kB. Jest przeznaczony do prostego przetwarzania nienumerycznego / arytmetyka stałoprzecinkowa / z możliwością sterowania sprzętem na niskim poziomie / bezpośrednio adresowany dostęp do pamięci i portów wejścia / wyjścia, możliwość wywoływania procedur napisanych w assemblerze /. TBASIC przystosowany jest do pracy w minimalnej konfiguracji pamięci / 16 kB / i pozwala na szybkie przygotowanie i uruchomienie stosunkowo krótkich programów z możliwością przechowywania ich w pamięci dyskowej.

Interpreter języka BASIC / MBASIC / – zajmuje 18 kB pamięci i współpracuje z CP / M 28 kB. MBASIC to szeroko rozbudowana implementacja tego języka, charakteryzująca się m. in. czterema typami zmiennych / całkowite, łańcuchowe, zmiennoprzecinkowe pojedynczej i podwójnej precyzji /, dynamiczną deklaracją rozmiarów tablic, szeroko rozbudowanymi funkcjami edycji tekstu programu, współpracy z pamięcią dyskową i systemem operacyjnym. MBASIC pozwala również na sterowanie sprzętem na niskim poziomie.

Interpreter języka FORTH / FORTH / – zajmuje ok 8 kB pamięci, ale efektywna praca, ze względu na rozmiary obszarów roboczych wymaga systemu CP M – 24 kB. Język FORTH jest uniwersalnym narzędziem dla przygotowania oprogramowania mikroprocesorowych systemów aplikacyjnych. Język, a dokładniej system FORTH pozwala na definiowanie własnego, specjalizowanego dla danej aplikacji słownika. Programy napisane w FORTH są zwarte i efektywne czasowo.

Moduł komparatora stanów – zawiera dwa analogiczne układy rozpoznające żądany stan magistrali adresowej w określonym cyklu maszynowym. Układy te pozwalają na realizację braków programu wykonywanego w czasie rzeczywistym w trybie emulacji oraz na warunkowe określenie trybu śladowania programu.

Moduł śladowania przebiegu programu w czasie rzeczywistym – pozwala na warunkowe zapamiętanie stanu 1024 cykli maszynowych. Informacje w ramach pojedynczego cyklu zapamiętane są na 32 bitach obejmujących: adres / 16 /, dane / 8 /, rodzaj cyklu / 4 / oraz cztery warunki wybrane przez użytkownika w dowolnym punkcie systemu uruchamianego. Deklaracja warunków śladowania i startu śladowania pozwala na selektywne śladowanie cykli.

Program sterujący emulacją – EMULATOR / EM80, EM85 – wersja programu przeznaczone do emulacji mikroprocesorów odpowiednio 8080 i 8085 / zajmuje 9, 5 kB PaO i współpracuje z systemem CP / M – 16 kB. Program ten koordynuje pracę wszystkich funkcjonalnych modułów urządzeńowych emulatora. Dostarcza on użytkownikowi możliwość operowania na zasobach SU, zasobach emulowanych, korzystania z funkcji systemu realizowanych sprzętowo jak np. śladowanie w czasach rzeczywistych, punkty zatrzymań oraz zastępuje oprogramowanie rezydujące typu loader i debugger dla uruchamianego systemu. Program ten umożliwi organizację procesu uruchamiania na drodze interaktywnego dialogu z użytkownikiem. Program EMULATOR wyposażony jest również w test modułu emulatora.

Program ładowający / ALOD / – jest przeznaczony głównie do współpracy z programatorem. Służy on do przepisanie zbioru dyskowego typu „COM” / obraz pamięci / np. utworzonego przy pomocy programu EMULATOR i zawierającego przetestowany program do pamięci operacyjnej. Początkowy adres ładowania podawany jest przez użytkownika lub przyjmowany standardowo – bufor programatora.

Test pamięci dyskowej / DTA / – jest programem przygotowanym w assemblerze i pozwala na testowanie całej dyskietki lub pojedynczego sektora. Test wykorzystuje odczyt jednosektorowy.

Test pamięci dyskowej / DTB / – jest programem napisanym w języku BASIC i wykonywany jest przy użyciu interpretera TBASIC. DTB wykorzystuje odczyt i zapis wielosektorowy.

5. CHARAKTERYSTYKA FUNKCJONALNA EMULATORA.

Oprogramowanie systemu wymienione w p. 4 umożliwia prowadzenie procesu konstrukcji oprogramowania dla budowanego systemu w ramach RTDS – 8. Użycie edytora i assemblera pozwala na przygotowanie tekstu programu, kompilację i uzyskanie w pamięci dyskowej zbioru z wynikową postacią programu. Użycie EMULATORA pozwala na dialogowe prowadzenie dalszego procesu uruchamiania w konfiguracji tworzonego systemu. Zlecenia EMULATORA można podzielić funkcjonalnie na trzy podstawowe grupy:

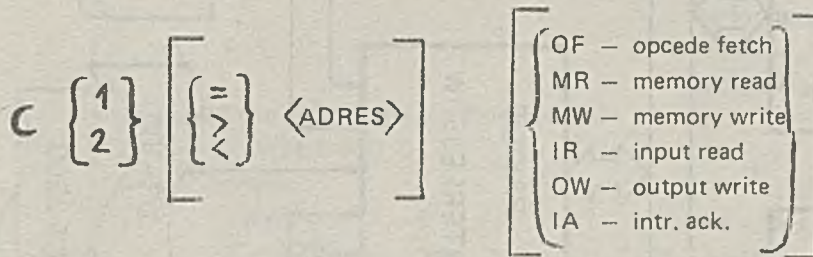
- zlecenia pozwalające na przygotowanie odpowiedniej konfiguracji pamięci i załadowanie programu,
- zlecenia pozwalające na obserwowanie i modyfikowanie stanu zasobów oraz na kontrolowany przebieg programu,
- zlecenia pozwalające na składowanie zawartości pamięci.

Pierwsza grupa zleceń pozwala, w przypadku braku wymaganej pamięci w tworzonym systemie na dołączenie bloków pamięci emulowanej poprzez podanie adresu i trybu dostępu do bloku. Po tych operacjach możliwe jest załadowanie programu do pamięci uruchamianego systemu / pamięć uruchamianego systemu i / lub pamięć emulowana /. Odpowiednie zlecenia pozwalają na ładowanie programu z pamięci dyskowej / zbioru typu „COM” i „HEX” /, z łącza komputerowego, z czytnika taśmy papierowej.

Przy pomocy zleceń drugiej grupy możliwe są następujące operacje:

- wyświetlanie i modyfikacja stanu mikroprocesora, pamięci i portów wejścia / wyjścia,
- wykonywanie programu po instrukcji, z wyświetlania mnemoniki instrukcji i pełnego stanu mikroprocesora,
- deklaracja warunków dla modułu komputera,
- wykonywanie programu w czasie rzeczywistym od zadanego adresu z zatrzymaniem po spełnieniu zadanego warunku zatrzymania,
- sterowanie śladowaniem przebiegu programu w czasie rzeczywistym i wyświetlanie śladu,
- pomiar czasu wykonywania programu.

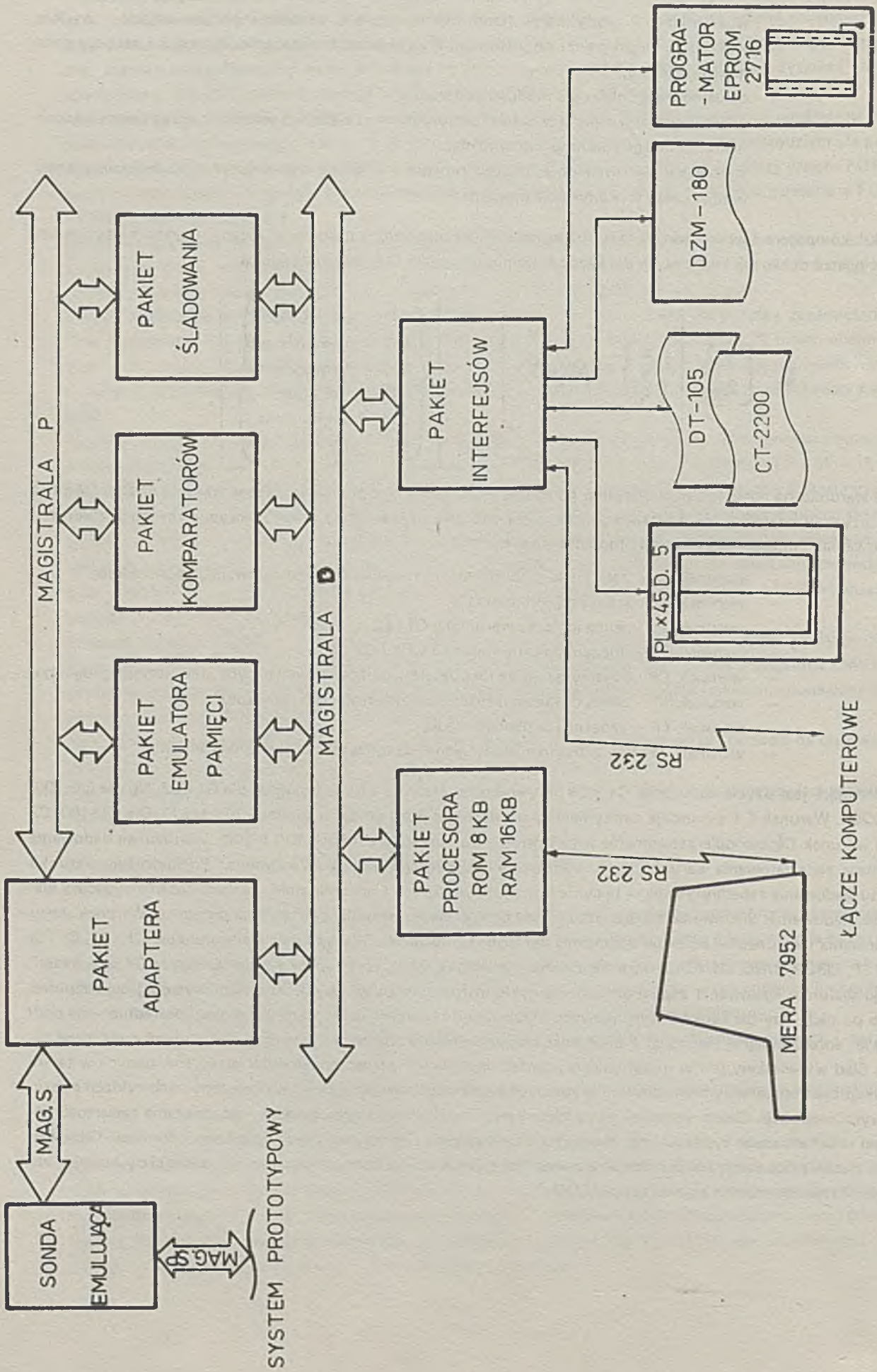
Moduł komputera zawiera dwa układy rozpoznające stan magistrali adresowej w zadanym cyklu maszynowym, toteż postać deklaracji i warunków dla każdego komputera / C1 i C2 / jest następująca:



Pole warunku na adres lub pole warunku na rodzaj cyklu może być pomijane. Zadane warunki na C1 i C2 mogą być użyte w deklaracji punktu zatrzymania programu wykonywanego w czasie rzeczywistym lub w deklaracji śladowania. Punktem zatrzymania programu może być:

- warunek C1 – zatrzymanie po spełnieniu warunku C1 zadeklarowanego poprzednio,
- warunek C2 – jak wyżej, tylko dla C2
- warunek C+ – suma logiczna warunków C1 i C2,
- warunek C \times – iloczyn logiczny warunków C1 i C2,
- warunek CP – zatrzymanie na instrukcjach skoków zmieniających stan licznika programu,
- warunek SP – zatrzymanie na instrukcjach zmieniających stan stosu,
- warunek TF – zapełnienie pamięci śladu,
- warunek USER – zatrzymanie gdy sygnał użytkownika w stanie aktywnym.

Interesujące jest użycie warunków C+ i C \times przy wyborze relacji na adresie typu $\langle \text{ADRES} \rangle$ dla C1 i C2. Np dla C1 $\langle 100, C\mathbf{\times} \rangle 200$. Warunek C+ powoduje zatrzymanie przy dostępie poza obszar o adresach 100 \mp 200. Dla C1 $\langle 100, C2 \rangle 200$ warunek C \times powoduje zatrzymanie przy dostępie do obszaru o adresach 100 \mp 200. Warunkowe śladowanie wymaga zadeklarowania warunku startu śladowania i właściwego warunku śladowania. Po spełnieniu warunku startu śladowania zapamiętywane są te cykle instrukcji, w których stan cyklu maszynowego spełnia właściwy warunek śladowania. Śladowanie zostaje wstrzymane po spełnieniu warunku zatrzymania programu. Warunek startu śladowania i warunek właściwego śladowania deklaruje się spośród opisanych powyżej warunków C1 i C2, C+, C \times , CP, SP, USER, UNC. /UNC oznacza śladowanie bezwarunkowe /. W przypadku zadeklarowania CP jako właściwego warunku śladowania zapamiętywane są cykle instrukcji skoków i cykle instrukcji występującej bezpośrednio po nich, aby dla skoków warunkowych możliwa była ocena wartości warunku skoku / pośrednio – na podstawie adresu kolejnej instrukcji /. Ślad może być wyświetlany od początku pamięci śladu lub od dowolnego cyklu. Ślad wyświetlany jest w mnemonicznej postaci instrukcji z zaznaczeniem cykli przesyłów danych w tej instrukcji oraz opcjonalnym stanem linii wybranych przez użytkownika / p. 4 / w poszczególnych cyklach maszynowych instrukcji. Celem ostatniej grupy zleceń jest umożliwienie użytkownikowi zapamiętania zawartości pamięci uruchamianego systemu / np. programu z poprawkami / na zewnętrznych nośnikach informacji. Odpowiednie zlecenia pozwalają na składowanie zawartości tej pamięci na taśmie papierowej lub pamięci dyskowej w autonomicznie tworzonym zbiorze typu „COM”.



Konfiguracja systemu RTDS - 8

6. SPECYFIKACJA SYSTEMU RTDS – 8.

SYSTEM BAZOWY

Funkcja: Wspomaganie konstrukcji oprogramowania układów mikrokomputerowych 8080/8085

Wyposażenie:

Zespół elektroniki:

~ ~ Kasety dziesięciopakietowa

~ ~ Pakiet procesora

 | ~ ~ ~ pakiet z procesorem 8085 i pamięcią 16 kB

 | ~ ~ ~ pakiet z procesorem 8080 i pamięcią 64 kB

Pakiet interfejsów peryferii:

— dysków elastycznych

— konsoli systemowych

— czytnika taśmy papierowej

— perforatora taśmy papierowej

— drukarki

— łącze szeregowe RS 232 C

~ ~ ~ Jednostka pamięci na dyskach elastycznych

 | ~ ~ ~ PL x 45D. 5

Zespół zasilacza

 | ~ ~ ~ SPS – 1B – 5.

 | / + 5V. 40A /

 | EZW – 02 – 00

 | / + 5V, ± 12V, + 48V /

Oprogramowanie.

~ ~ Monitor systemowy

~ ~ Dyskowy system operacyjny kompatybilny z CP / M

~ ~ ~ Testy

~ ~ Biblioteka programów pomocniczych

~ ~ Edytor

~ ~ Asembler 80 80

~ ~ Debugger 80 80

~ ~ TABASIC – prosty interpreter

~ ~ FORTH – system programowania

UNIERSALNY EMULATOR MIKROPRPCESORÓW 8 – bitowych.

Wyposarzenie:

Zespół elektroniki

~ ~ Pakiet adaptera

~ ~ Pakiet pamięci emulowanej 16 kB

~ ~ Pakiet komponentów adresu i cyklu z programatorem pamięci EPROM 2716

~ ~ Pakiet selektywnego śladowania 32 bitowej informacji / 16 bitowy adres 8 bitowa dana, rodzaj cyklu – 4 bity, 4 bity analizatora

Środowisko sprzętowe i programowe:

SONDY

System bazowy

Sonda procesora 8085A – emulacja w czasie rzeczywistym mikroprocesora 8085A

~ ~ ~ Zespół elektroniki sondy 8085A

~ ~ ~ Program sterujący emulacją dla procesora 8085A

Sonda procesora 8085A – emulacja w czasie rzeczywistym mikroprocesora 8080A

~ ~ ~ Zespół elektroniki sondy 8080A

~ ~ ~ Program sterujący emulacją dla procesora 8080A

Środowisko sprzętowe i programowe:

PERYFERIA

Uniwersalny emulator procesorów 8 – bitowych.

~ ~ Monitor-ekranowy z interfejsem RS232C

 | ~ ~ MERA 7952

 | ~ ~ MERA 7952 VGD

~ ~ Drukarka znakowa – interfejs równoległy

 | ~ ~ DZM – 180

✧ Perforator i / lub czytnik taśmy papierowej – interfejs równoległy

| ✧ SPTP – 3

^ Stolik urządzeniowy

OPROGRAMOWANIE

✧ M BASIC – interpreter BASIC / 18 kB /

| Środowisko programowe: CP / M

✧ System operacyjny kompatybilny z ISIS II

✧ ✧ biblioteka programów użytkowych

✧ ✧ Makroassembler

✧ ✧ Linker

✧ ✧ Loader

✧ ✧ PL / M – kompilator języka PL / M

✧ ✧ FORTRAN – kompilator języka FORTRAN

Środowisko sprzętowe – System bazowy

✧ ✧ Uwaga: Dostępne w 1984r.



**ZAKŁADY
URZĄDZEŃ KOMPUTEROWYCH
MERA-ELZAB
41-808 ZĄBRZE
ul. Kruczkowskiego 39
POLAND
Telex: 036411, 036412
Phone: 72 20 21-29**



**POLISH FOREIGN
TRADE ENTERPRISE
Warszawa 1, ul. Mysia 2
Poland
Telephone: 210 17
Cable address:
METRONEX - WARSZAWA
Telex: 814471 P.O.Box 198**