System uruchomieniowy ROM68K płyty prototypowej MC68EC0x0IDP

Dr inż. Witold Marańda

13 października 2000

Spis treści

1 Konfiguracja systemu

Płyta prototypowa `MC68EC0x0IDP` jest wyposażona w program ROM68K umieszczony w pamięci ROM i uruchamiany po włączeniu zasilania. Program ten realizuje następujące funkcje:

- zapewnia podstawową komunikację użytkownika z systemem poprzez port szeregowy, tj. pozwala na przeglądanie i modyfikowanie zawartości pamięci i rejestrów wewnętrznych mikroprocesora oraz uruchamiania programów;
- pozwala na przesyłanie plików do pamięci systemu poprzez port szeregowy lub interfejs sieciowy;
- umożliwia automatyczne załadowanie innego systemu operacyjnego z opcjonalnej pamięci ROM lub z sieci oraz uruchamianie go zawsze po restarcie systemu.

Najprostsza konfiguracja systemu składa się z płyty prototypowej oraz komputera macierzystego połączonych łączem szeregowym z protokółem RS-232. Na komputerze macierzystym, wykorzystywanym jako konsola systemu `MC68EC0x0IDP`, musi być uruchomiony dowolny program terminala, z możliwością przesyłania danych poprzez port szeregowy.

Podczas pracy z programem ROM68K polecenia wydawane są z klawiatury terminala (tj. komputera macierzystego), a ich efekty wyprowadzane są na ekran monitora terminala. Po włączeniu zasilania, program ROM68K zgłasza się komunikatem i znakiem zachęty:

```
Copyright 1993 Motorola Inc. All rights reserved
Copyright 1993 Software Components Group. All rights reserved
IDP ROM Version 3.0
Processor Type is: M68030
DRAM Size: 2 Megs

ROM68K :->
```

2 Składnia poleceń

Wszystkie polecenia programu ROM68K są dwuliterowe po których podaje się wymagane argumenty (program nie rozróżnia małych i wielkich liter) np.: `GO 1B`, `pr PC 40000`, `db 50ac6 8`.

Wszystkie numeryczne argumenty poleceń tj. dane lub adresy, są liczbami szestnastkowymi bez żadnego znaku prefixu.

Odwołanie do rejestrów wewnętrzych mikroprocesora odbywa się przez podanie jego nazwy tj.:

`D0-D7` — *Data Registers*
`A0-A7` — *Address Registers*
`PC` — *Program Counter*
`SR` — *Status Register*
`USP`, `MSP`, `ISP` — *User, Master, Interrupt Stack Pointer*
`VBR` — *Vector Base Register*
`DFC`, `SFC` — *Destination and Source Function Code registers*
`CACR`, `CAAR` — *Cache Control and Address Registers*

Domyślnie wszystkie polecenia operują na danych o rozmiarze bajtu, ale użytkownik może zmienić to zachowanie dodając symbol rozmiaru danych do polecenia. Dopuszczalne są następujące symbole: `.B` (bajt), `.W` (słowo: 2 bajty) oraz `.L` (długie słowo: 4 bajty).

Niektóre polecenia akceptują lub wymagają podania jako argumentu obszaru pamięci. Obszary można zapisywać podając początkowy i końcowy adres rozdzielony dwiema kropkami np. `40000..400FF` lub podając adres początku obszaru i ilość elementów np. `40000 FF`. W ostatnim przypadku, adres końcowy obliczany jest przez dodanie do adresu poczatkowego ilości elementów przeskalowanych względem rozmiaru danych i pomniejszonych o jeden. Komórka o adresie końcowym również należy do danego obszaru.

Adresy mogą być podawane jako absolutne lub względem zawartości rejestrów adresowych. W tym drugim przypadku, adres zbudowany jest z przesunięcia, znaku ukosnika oraz nazwy rejestru np. `2F/A0`, `0/A0` lub `/A0`.

Przykłady:

- `DM 50000` — pokazuje zawartość pamięci w bajtach od adresu szesnastkowego `50000`.
- `DM.L 50000..5000A` — pokazuje zawartość pamięci w długich słowach w zakresie od `50000` do `5000A` włącznie.
- `DM.W 50000 400` — pokazuje zawartość 400 słów pamięci tj. w zakresie od `50000` do `507FF` włącznie.
- `DM 4B/A1` — pokazuje zawartość pamięci w bajtach poczynając od adresu zawartego w rejestrze `A1`, powiększonego o liczbę 4B.

Polecenia dotyczące pamięci systemu	
`AS`	Asembluj do pamięci
`CM`	Porównaj pamięć
`DI`	Disasembluj pamięć
`DM`	Wyświetl pamięć
`FM`	Wypełnij pamięć
`MM`	Kopiuj pamięć
`MP`	Mapuj pamięć
`PM`	Zapisz do pamięci
`SM`	Przeszukaj pamięć
Polecenia dotyczące rejestrów mikroprocesora	
`DO`	Wyświetl rejestry dodatkowe
`DR`	Wyświetl zawartość rejestrów
`PR`	Zapisz do rejestrów
Polecenia związane z uruchamianiem programów	
`CB`	Usuń punkt przerwań
`DB`	Ustaw punkt przerwań
`GO`	Uruchom program
`LB`	Wyświetl listę punktów przerwań
`ST`	Wykonaj instrukcję
Polecenia związane z przesyłaniem danych	
`BI`	Uruchom program monitora MON68
`DC`	Załaduj plik w postaci S-rekordów
Inne polecenia	
`EC`	Oblicz wartość dziesiętną
`HC`	Konfiguracja sprzętu
`HE`	Wyświetl opis poleceń

Tablica 1: Lista poleceń programu ROM68K

3 Opis poleceń

Pełna lista poleceń programu ROM68K znajduje się w tablicy 1 na stronie 3. Polecenia można podzielić na następujące grupy:

- polecenia dotyczące pamięci systemu, obejmujące czynności inspekcji, wyszukiwania, porównywania oraz modyfikacji zawartości pojedynczych komórek lub bloków pamięci;
- polecenia dotyczące rejestrów mikroprocesora, pozwalające na inspekcję lub modyfikacje wewnętrznych rejestrów mikroprocesora;

- polecenia związane z uruchamianiem programów znajdujących się w pamięci operacyjnej systemu; programy można również uruchamiać w trybie krokowym oraz definiować, wyświetlać i usuwać punkty przerwań w celu ułatwienia śledzenia pracy programu i wykrywania błędów;
- polecenia związane z przesyłaniem danych, służące do ładowania danych w postaci S–rekordów do pamięci systemu z portu szeregowego lub sieci (jeśli zainstalowany jest interfejs sieciowy);
- inne polecenia, służące do zmiany konfiguracji systemu, konwersji liczb szestnastowych na dziesiętne oraz do uzyskania prostego opisu wszystkich poleceń.

W opisie poleceń, argumenty podane w nawiasach kwadratowych '`[]`' są opcjonalne, natomiast rozdzielenie argumentów znakiem '`|`' oznacza konieczność wyboru jednego z nich.

3.1 `AS` — asembluj do pamięci

Składnia: `AS[.rozmiar] adres`

`adres` — adres komórki pamięci

`rozmiar` — symbol rozmiaru danych: `B`, `W` lub `L`

Polecenie `AS` wyświetla i pozwala zmienić zawartość komórki pamięci o podanym adresie używając standardowych mnemoników asemblera 680x0 oraz szestnastkowo. W odpowiedzi na wyświetloną informację i znak zachęty '=', użytkownik może wprowadzić nową instrukcję, zakończyć wykonywanie polecenia przez wpisanie znaku kropki '.' lub po naciśnięciu klawisza *Enter* przejść do asemblacji następnej instrukcji.

Podczas wprowadzania nowej instrukcji, polecenie akceptuje standardowe mnemoniki asemblera 680x0, przy czym polecenia nieodpowiednie dla danego typu procesora będą odrzucone. Adresy względne np. dla instrukcji skoków, muszą być podawane w postaci absolutnej — adres względny będzie obliczony automatycznie.

! W odróżnieniu od innych poleceń, liczby w argumentach polecenia `AS` traktowane są domyślnie jako dziesiętne, natomiast szestnastkowe wymagają prefixu '$'.

3.2 `BI` — uruchom program monitora MON68

Składnia: `BI`

Polecenie służy do przełączenia z programu ROM68 do MON68. Konfiguracja systemu nie ulega zmianie, tj. nie zmienia się aktualne mapowanie pamięci oraz konfiguracje urządzeń systemowych dokonane podczas pracy w programie ROM68.

3.3 `CB` — Usuń punkt przerwań

Składnia: `CB indeks`

`indeks` — numer punktu przerwania

Polecenie `CB` usuwa zdefiniowany poleceniem `DB` punkt przerwania o podanym numerze. Jeśli jako argument podany zostanie znak '*', usuniete zostaną wszystkie punkty przerwań.

3.4 `CM` — porównaj pamięć

Składnia: `CM[.rozmiar] obszar adres`

`rozmiar` — symbol rozmiaru danych: `B`, `W` lub `L`

`obszar` — obszar pamięci, tj. adres poczatkowy i końcowy

`adres` — adres początkowy obszaru do porównania

Polecenie `CM` dokonuje porównania dwóch bloków pamięci podanych jako argumenty polecenia. Sygnalizowane są wszystkie przypadki niezgodności występujące w porównywanym obszarze.

3.5 `DB` — ustaw punkt przerwań

Składnia: `DB adres [licznik]`

`adres` — adres instrukcji

`licznik` — licznik przejść punktu przerwań

Polecenie ustawia nowy punkt przerwnia dla wskazanej instrukcji. Wykonywanie programu będzie przerwane przed wykonaniem instrukcji dla której zdefiniowano punkt przerwania.

Działanie punktu przerwania może być zmienione przez podanie wartości licznika przerwań. Podczas każdego przejścia przez punkt przerwania, jego licznik jest zmniejszany o jeden. Zatrzymanie programu w danym punkcie nastąpi więc dopiero na ostatnim, z podanej liczby, wykonaniu instrukcji. Jeśli nie podano wartości licznika, przyjmowana jest domyślnie liczba 1.

Polecenie `LB` pokazuje aktualną wartość licznika przerwań, związanego z każdym punktem przerwania.

3.6 `DC` — Załaduj plik w postaci S-rekordów

Składnia: `DC[.rozmiar] [przesunięcie] [PREV]`

`rozmiar` — symbol rozmiaru danych: `B`, `W` lub `L`

`przesunięcie` — przesunięcie względem oryginalnego adresu ładowanych danych

`PREV` — żądanie drukowania końcowego komunikatu z poprzedniego wykonania polecenia `DC`

Polecenie ładuje dane w postaci S-rekoru napływające z portu szeregowego, tj. z konsoli systemu, do pamięci komputera. Po wydaniu polecenia `DC` pojawia się komunikat: `Waiting for S-records from the host` , po którym wszelkie dane napływające z konsoli są traktowane jako dane do załadowania. Transmisja kończy się po otrzymaniu S-rekordu kończącego blok danych, po ręcznym przerwaniu transmisji lub resecie systemu. Po pomyślnym zakończeniu transmisji wyświetlany jest komunikat z informacją o liczbie załadowanych danych.

! Transmisja 1kB danych z domyślną prędkością 9600baud trwa około 1s, a więc w przypadku transferu większych plików należy zachować cierpliwość.

! Dane w pamięci systemu mogą być niszczone podczas restartu komputera,jeśli włączona jest opcja testowania pamięci systemu (zobacz polecenie `HC`).

3.7 `DI` — disasembluj pamięć

Składnia: `DI[.rozmiar] [adres] [licznik]`

`rozmiar` — symbol rozmiaru danych: `B`, `W` lub `L`

`adres` — adres pierwszej disasemblowanej instrukcji

`licznik` — liczba disasemblowanych instrukcji

Polecenie wyświtla zawartość pamięci od podanego adresu w postaci standardowych mnemoników asemblera 680x0. Jesli nie podano adresu, disasemblowana jest pamięć od adresu znajdującego się w rejestrze PC. Dodatkowo można podać liczbę instrukcji, które mają być disasemblowane, domyślna wartość wynosi 8.

Disasemblacja może być powtórzone przez naciśnięcie klawisza *Escape*, z tym, że wówczas rozpoczyna się od następnej instrukcji za blokiem, ktory został wyświetlony przez poprzednie polecenie.

Zawartość pamięci, której nie odpowiada żaden kod instrukcji maszynowej, wyświetlany jest w postaci szestnastkowej jako `DC.W hex`.

! Argumenty disasemblowanych instrukcji są wyświetlane jako liczby dziesiętne, ale ich wartości szestnastkowe mogą być odczytane z kodu instrukcji wyświetlanego równolegle.

3.8 `DM` — wyświetl pamięć

Składnia: `DM[.rozmiar] adres|obszar [licznik]`

`rozmiar` — symbol rozmiaru danych: `B`, `W` lub `L`

`adres` — adres początkowy obszaru pamięci `obszar` — obszar pamięci, tj. adres poczatkowy i końcowy

`licznik` — liczba elementów danych do wyświetlenia

Polecenie wyświetla zawartość pamięci w postaci szestnastkowej i ASCII razem z jej adresem i adresem względnym. Położenie bloku pamięci może być podane w postaci kompletnego zakresu lub adresu i liczby elementów do wyświetlenia. Domyślna wartość elementów wynosi 80. Polecenie zawsze wyświetla pamięć w porcjach będących całkowitą wielokrotnością 16 bajtów.

Polcenie może być powtórzone przez naciśnięcie klawisza *Escape* i wówczas wyświetlony zostanie następny blok pamięci, położony za wyświetlonym poprzednio.

3.9 DO — wyświetl rejestry monitora

Składnia: `DO`

Polecenie wyświetla zawartość dodatkowych rejestrów programu monitora `F0`–`F7`.

3.10 DR — wyświetl zawartość rejestrów

Składnia: `DR`

Polecenie wyświetla zawartość wewnętrznych rejestrów mikroprocesora:

```
Current registers.
---------------------------------------------------------------------------
SR=2708-tfSm.111...xNzvc USP=00000000 MSP=00000500 ISP=00000500
VBR=00000000 DFC=7 SFC=7 CACR=0000 CAAR=F7FEFFFE
DR=901C0080 00000000 00000000 00000000  00000000 00000000 00000000 00000000
AR=00000001 00000000 00000000 00000000  00000000 00000000 00000000
PC=00000026-00000026  900C              DC.W    $900C
```

Zawartość rejestru statusowego `SR` wyświetlona jest szesnastkowo oraz symbolicznie, tj. litery oznaczają bity (małe =0, wielkie =1), a cyfry maskę przerwań np. t → *Trace*=0, S → *Supervisor*=1, x → *Extended*=0, N → *Negatative*=1, itd.

Po skrócie `DR` wyświetlona jest zawartość ośmiu rejestrów danych, a po `AR` – zawartość siedmiu rejestrów adresowych (rejestrem `A7` jest `USP` lub `ISP`, w zależności od kontekstu).

Na końcu, po rejestrze `PC` wyświetlana jest zawartość słowa pamięci pod adresem wskazywanym przez `PC` oraz jej mnemomik asemblera, a jeśli taki nie istnieje sygnalizowane jest to kodem `DC.W`.

Dodatkowo, powyżej może być wyświetlana informacja o ostatniej przyczynie wstrzymania wykonywania programu, np. `Ill Ins` — *Illegal Instruction*

```
RS Exception.  Level=7  F/O=0010 (Ill Ins)
---------------------------------------------------------------------------
SR=2708-tfSm.111...xNzvc USP=00000000 MSP=00000500 ISP=00000500
VBR=00000000 DFC=7 SFC=7 CACR=0000 CAAR=F7FEFFFE
DR=DBDE00A8 00809014 00000000 00000000  00000000 00000000 00000000 00000000
AR=00000001 00000000 00000000 00000000  00000004 00000000 00000000
PC=00000412-00000412  9D3C              DC.W    $9D3C
```

3.11 EC — oblicz wartość dziesiętną

Składnia: `EC wyrażenie`

`wyrażenie` — liczba szestnastkowa lub wyrażenie

Polecenie wyświetla podaną liczbę szesnastkową w postaci dziesiętnej. Możliwe jest również obliczanie wartości wyrażeń z użyciem nazw rejestrów (poprzedzonych znakiem '/') oraz operatorów dodawania i odejmowania.

3.12 FM — wypełnij pamięć

Składnia: `FM[.rozmiar] obszar wartość`

`rozmiar` — symbol rozmiaru danych: `B`, `W` lub `L` `obszar` — obszar pamięci, tj. adres poczatkowy i końcowy

`wartość` — wartość, która będzie wpisywana do pamięci

Polecenie wypełnia podany blok pamięci podaną wartością. Podana liczba powinna być zgodna z zadeklarowanym typem danych.

3.13 `GO` — uruchom program

Składnia: `GO [adres] [,punkt1] [,punkt2]`

`adres` — adres, od którego ma się rozpocząć wykonywania programu

`punkt1, punkt2` — chwilowe punkty przerwań

Polecenie uruchamia program znajdujący się w pamięci komputera od adresu zawartego w rejestrze `PC` lub od adresu podanego jako argument polecenia. Dodatkowo można podać dwa chwilowe punkty przerwań. Ich działanie ogranicza się jedynie do aktualnego uchuchomienia programu, tj. po zakończeniu działania programu, punkty te są usuwane z listy punktów przerwań. Punkty przerwań podaje się jako poprawny adres do instrucji programu.

Poprawne zakończenie wykonywania programu, tj. powrót do programu ROM68 powinno odbywać się poprzez wywołanie sytuacji wyjątkowej (np. *illegal instruction* (kod `$4AFC`) lub `trap #0`).

3.14 `HC` — konfiguracja sprzętu

Składnia: `HC obiekt dane`

`obiekt` — obiekt podlegający konfiguracji: `T`,`B` lub `M`

`dane` — dane konfiguracyjne

Polecenie służy do konfiguracji następujących elementów systemu:

`T` — ustawianie czasu i daty systemowego zagara czasu rzeczywistego

`HC T yr mo dt dy hr mi` (*Time*) — ustawienia daty i czasu, gdzie:

`yr` — rok, od 00 do 99
`mo` — miesiąc, od 01 do 12
`dt` — dzień, od 01 do 31
`dy` — dzień tygodnia, od 01 (poniedziałek) do 07 (niedziela)
`hr` — godzina, od 00 do 23
`mi` — minuta, od 00 do 59

`B` — ustawianie prędkości transmisji danych portu szeregowego

`HC B bps` (*Baud rate*) — ustawienie prędkości na `bps`, która może być jedną z trzech wartości: 9600, 19200 lub 38400. Nowe ustawienie będzie aktywne po restarcie systemu.

Podczas restartu, na wyświetlaczu LED płyty systemu wyświetlana jest na okres 1s cyfra informująca o aktualnej szybkość transmisji: cyfra 9 odpowiada szybkości 9600, 1 — 19200, a 3 — 38400. Uwaga: naciśnięcie mikroprzełącznika ABORT podczas wyświetlania cyfry powoduje, że system startuje z domyslną prędkością 9600bps.

`M` — Włączanie/wyłączanie testowania pamięci podczas restartu systemu

`HC M D` (*Memory Disable*) — wyłączenie opcji testowania; programy istniejące w pamięci nie są niszczone.

`HC M E` (*Memory Enable*) — włączenie opcji testowania; programy istniejące w pamięci są niszczone.

Podczas częstego uruchamiania i ładowania plików zaleca się wyłączenie testowania, aby programy nie były usuwane z pamięci po każdym restarcie systemu.

3.15 `HE` — wyświetl opis poleceń

Składnia: `HE [polecenie]`

`polecenie` — nazwa polecenia

Polecenie wyświetla pełną listę dostępnych poleceń programu ROM68 wraz z ich skrótowym opisem. Podanie argumentu powoduje wyświetlenie informacji jednynie o wybranym poleceniu. Odpowiednikiem tego polecenia są także `HELP` oraz `?`.

3.16 LB — wyświetl listę punktów przerwań

Składnia: `LB`

Polecenie wyświetla aktualną listę wszystkich punktów przerwań oraz ich liczniki przejść.

3.17 MM — kopiuj pamięć

Składnia: `MM[.rozmiar] obszar adres`

`rozmiar` — symbol rozmiaru danych: `B`, `W` lub `L`
`obszar` — kopiowany obszar pamięci , tj. adres poczatkowy i końcowy
`adres` — adres początkowy obszaru przeznaczenia

Polecenie wypełnia kopiuje dane z podanego obszaru pamięci do obszaru wskazanego poprzez adres poczatkowy.

3.18 MP — ustaw mapowanie adresów logicznych systemu

Składnia: `MP`

Polecenie to umożliwia zmianę przyporządkowania adresów logicznych systemu do fizycznych obszarów pamięci systemu. Stosowane jest głównie w sytuacji dołączania nowych urządzeń lub pamięci do głownej magistrali systemu mikroprocesorowego. Szczegóły użycia polecenia zostały opisane w dokumentacji płyty prototypowej `MC68EC0x0IDP`.

3.19 PM — zapisz do pamięci

Składnia: `PM[.rozmiar] adres [SKIP] [wartość]`

`rozmiar` — symbol rozmiaru danych: `B`, `W` lub `L`
`adres` — adres komórki pamięci
`SKIP` — omiń komórki
`wartość` — wartość, która będzie wpisywana do pamięci

Polecenie wyświetla i pozwala wpisać nową wartość do komórki pamięci o podanym adresie. Polecenie może pracować w trybie natychmiastowym lub interaktywnym. W trybie natychmiastowym, tj. gdy w linii polecenia podano adres i wartość, wykonanie dotyczy tylko bieżącej komórki i kończy się po wpisaniu nowej wartości.

W trybie interaktywnym, tj. po podaniu tylko adresu komórki, użytkownik może przeglądać i wpisywać wartości do serii komórek pamięci. Na ekranie wyświetlany jest adres oraz aktualna wartość, a program czeka na podanie rozkazu:

. — (kropka) zakończenia pracy interaktywnej
\ — powrótu do poprzedniego elementu
= — powtórnego wyświetlenia bieżącego elementu
(null) — przejścia do następnego elementu

Opcja `SKIP` (tylko w trybie interaktywnym) pozwala ominąć nieistniejące fizycznie komórki, np. podczas pracy z rejestrami urządzeń zewnętrznych, których 8–bitowe rejestry mapowane są tylko pod parzystymi lub nieparzystymi adresami.

3.20 PR — zapisz do rejestrów

Składnia: `PR [rejestr] [wartość]`

`rejestr` — nazwa rejestru wewetrznego mikroprocesora
`wartość` — wartość, która będzie wpisywana do rejestru

Polecenie wyświetla i pozwala wpisać nową wartość do rejestrów mikroprocesora oraz dodatkowych rejestrów monitora. Polecenie może pracować w trybie natychmiastowym lub interaktywnym. W trybie natychmiastowym, tj. gdy w linii polecenia podano nazwę rejestru i wartość, wykonanie dotyczy tylko danego rejestru i kończy się po wpisaniu nowej wartości.

W trybie interaktywnym, tj. gdy podano tylko nazwę rejestru, użytkownik może przeglądać i wpisywać wartości rejestrów mikroprocesora. Rejestry podzielono na cztery grupy: 1) sterujące (PC, SR, VBR), 2) danych (D0-D7), 3) adresowe (A0-A7) oraz 4) monitora (F0-F7). Polecenie będzie wyświetlać tylko rejestry z grupy, do której należy nazwa podana w linii polecenia. Na ekranie wyświetlana jest nazwa rejestru oraz jego aktualna wartość, a program czeka na podanie rozkazu:

. — (kropka) zakończenia pracy interaktywnej
\ — powrótu do poprzedniego elementu
= — powtórnego wyświetlenia bieżącego elementu
(null) — przejścia do następnego elementu

3.21 `SM` — przeszukaj pamięć

Składnia: `SM[.rozmiar] obszar [NOT] wartość`

`rozmiar` — symbol rozmiaru danych: `B`, `W` lub `L`
`obszar` — obszar pamięci, tj. adres poczatkowy i końcowy
`NOT` — wyszukuj elementy nie równe
`wartość` — wartość do porównania

Polecenie przeszukuje podany obszar pamięci i wyświetla wszystkie komórki, których zawartość jest różna od podanej wartości. W przypadku typu danych innych niż `.B`, przeszukiwane elementy muszą zaczynać się pod parzystym adresem.

Opcja `NOT` odwraca sens przeszukiwania, tj powoduje, że polecnie wyszukuje elementy, których wartość nie równa się wartości podanej w linii polecenia.

3.22 `ST` — wykonaj instrukcję

Składnia: `ST [licznik]`

`licznik` — liczba instrukcji do wykonania

Polecenie wykonuje jedną instrukcję wskazywaną przez aktualną wartość rejestru `PC`. Wykonanie większej liczby instrukcji wymaga podania parametru w linii poleceń.

Polecenie, przez wykonaniem, wyświetla adres, kod i mnemonik wykonywanej instrukcji, a po jej wykonaniu (lub ostatniej z serii) zawartość wszystkich rejestrów mikroprocesora oraz następną intrukcję do wykonania.

! Jeśli wartość maski przerwań w `SR` pozwala na przyjęcie przerwania, to procedura jego obsługi nie będzie śledzona. Przerwania nadchodzące w trakcie wykonywania polecenia `ST` będą normalnie obsługiwane i nie będzie to sygnalizowane użytkownikowi.