

Debugging in Mainframe using Tracemaster

Author(s): (1) [divya_manjari](#)

Written On: 05/24/04

Declaration

I/We hereby declare that this document is based on our personal experiences and / or experiences of my/our project members. To the best of our knowledge, this document does not contain any material that infringes the copyrights of any other individual or organization including the customers of Infosys.

divya_manjari

Project Details :

Projects Involved : LIONSYCH
Hardware WPlatform : IBM OS/390
Software Environment : MVS,CICS,Cobol,VSAM,JCL
Project Type : Maintenance

***Target readers :* All mainframe users**

Keywords:

mainframe,COBOL,CICS,debugging,Tracemaster,tools

Introduction:

TRACEMASTER is a tool for debugging and testing COBOL and assembler sources in mainframe. This document explains the step by step procedure for debugging the CICS program in CICS and the cobol program in foreground mode.

How to Debug a COBOL/CICS program using Tracemaster

Debugging a COBOL/CICS program in CICS

1. [Compile and link the program for trace master](#)
2. [Load the programs and symbol files through CICS Trace master menu](#)
3. [Set the breakpoints](#)
4. [Run the transaction and Debug](#)
5. [Log off](#)

Debugging a COBOL batch program (in Foreground)

1. [Compile and link the program for trace master](#)
2. [Set Default libraries](#)
3. [Set symbol library and Convert the JCL](#)
4. [Run in debug mode](#)

Debugging a COBOL/CICS program in CICS

STEP 1

Compile and link the program for trace master – First of all we need to do some preparation for testing the program in Tracemaster. We need to create a symbol file for Tracemaster. Hence, for running a program using Tracemaster, the program needs

to be compiled in a modified manner. The JCL ,Procedure or macros for compilation needs to undergo following changes:

- Change the program for compilation
 - IGYCRCTL → XT00VSC2 (Cobol II)
 - IKFCBL00 → XT000SCB (OS/V S Cobol)
- Add symbol library
 - *prefix.symbol.library(program name)*
- Add trace master load library
 - *prefix.XTRESLIB* – It should be trace master load library name specified during installation.

Example: The existing macro for compiling a cobol program has following step:

```
//COB EXEC PGM=IGYCRCTL,  
//STEPLIB DD DSN=SYS1.SIGYCOMP,DISP=SHR  
// DD DSN=SYSP.LL.CICSTS22.CICS.SDFHLOAD,DISP=SHR
```

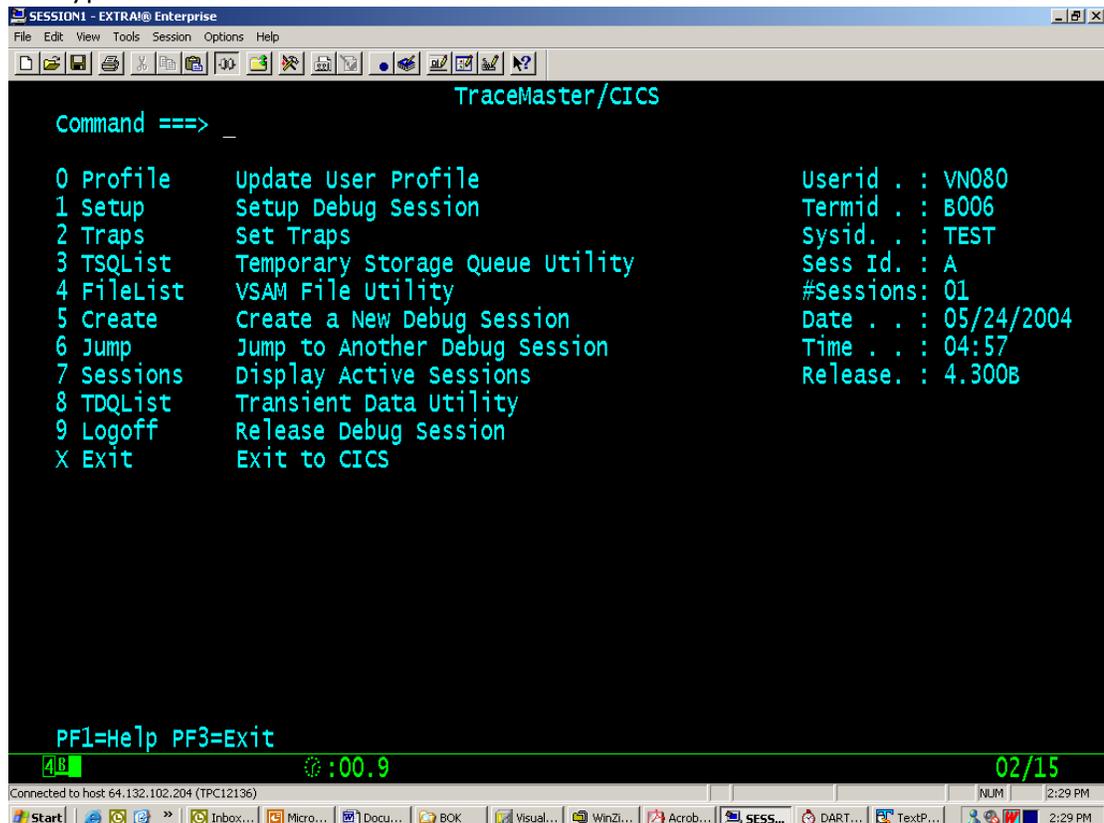
The above step is changed as follows for compiling a program to be tested in Tracemaster.

```
//COB EXEC PGM=XT00VSC2  
//STEPLIB DD DSN=SYST.PD.TRACEMST.V4100M.XTRESLIB,DISP=SHR  
// DD DSN=SYS1.SIGYCOMP,DISP=SHR  
//SYMOUT DD DSN=TST1.FOODLION.SYMBOL.LIBRARY("MEM"),  
// DISP=SHR
```

STEP 2

Load the programs and symbol files through CICS Trace master menu – After compiling and linking the program, following steps needs to be followed to load the program :

- do a new copy of the program in CICS screen.
- Type the transaction ID for Tracemaster on CICS Screen.Default is XXT0.



- When you are entering the Tracemaster menu for the first time, go to 'PROFILE' option and set the symbol library(*prefix.symbol.library* ->same

as in compilation procedure). Also set the Display format as 'C'(character). Save and exit to Tracemaster main menu.

```
SESSION1 - EXTRAI@ Enterprise
File Edit View Tools Session Options Help
PROFILE Update User Profile
Command ==> _

Enter fully qualified dataset names, without quotes
Macro Library . . .
Symbol Library. . . TST1.FOODLION.SYMBOL.LIBRARY
or Scheme Name. . . (Enter a / for a scheme list)
Scheme Description.

Display format requested on file and TSQ utilities
Display Format. . . C (D or Blank=Dump; C=Character)

PF1=Help PF3=Exit
4.8 :00.4 02/15
Connected to host 64.132.102.204 (TPC12136)
Start Inbox... Micro... Docu... BOK Visual... WinZi... Acrob... SESS... DART... TextP... NUM 2:32 PM
```

- o Now go to 'SETUP' option and enter "COBOL" command

```
SESSION1 - EXTRAI@ Enterprise
File Edit View Tools Session Options Help
*-DETAIL(D)-*-
+==TRACE(A)==+
Loc Object Code source Statement Addr1,Addr2

0-7 00000000 00000000 00000000 00000000 00000000 00000000 00000000
8-15 00000000 00000000 00000000 00000000 00000000 00000000 00000000
+====+

==> cobo1_
4.8 :00.3 24/11
Connected to host 64.132.102.204 (TPC12136)
Start Inbox... Micro... Docu... BOK Visual... WinZi... Acrob... SESS... DART... TextP... NUM 2:34 PM
```

- o Load the program by entering LOAD command followed by program name to be tested. E.g. LOAD INVO701.After successful load following msg would appear .

SESSION1 - EXTRA@ Enterprise

File Edit View Tools Session Options Help

+SOURCE(C)=

M400: INVO701 loaded from TST1.FOODLION.LOADLIB

4B :00.7 24/06

Connected to host 64.132.102.204 (TPC12136)

Start | Inbox... | Micro... | Docu... | BOK | Visual... | WinZi... | Acrob... | SESS... | DART... | TextP... | NUM | 2:38 PM

- Load the symbol files using the command XSYMBOLS followed by the program name. For e.g. XSYMBOLS INVO701. On successful load the program would be displayed on the screen.

```

===== INVO701 =====
000589      PROCEDURE DIVISION using dfheiblk dfhcommarea.
000590      *
000591      *****
000592      *                MAIN PROCESSING PARAGRAPH.
000593      *****
000594      *
000595      0000-MAIN-PARA.
000596      *
000597      PERFORM 0100-INITIALIZATION
000598      IF EIBCALEN          =      ZEROES
000599      PERFORM 9005-INVALID-ENTRY-PARA
000600      ELSE
000601      PERFORM 1000-PROCESS
000602      PERFORM 9000-OUTPUT-SCREEN
000603      END-IF
000604      .
000605      *
000606      *****
000607      *                INITIALIZATION ROUTINE
000608      *****
=====
M076: symbol information successfully loaded
===>

```

STEP 3

Set the breakpoints – After loading the program, set the breakpoints in the program, wherever required.

It can be done in following ways

- Place the cursor on line and press PF14
- BP *line number*
- BP *Paragraph name*

In the following screen shots two breakpoints are set.

```

===== INVO701 =====
000589      PROCEDURE DIVISION using dfheiblk dfhcommarea.
000590      *
000591      *****
000592      *                MAIN PROCESSING PARAGRAPH.
000593      *****
000594      *
000595      0000-MAIN-PARA.
000596      *
000597      PERFORM 0100-INITIALIZATION
000598      IF EIBCALEN = ZEROES
000599      PERFORM 9005-INVALID-ENTRY-PARA
000600      ELSE
000601      PERFORM 1000-PROCESS
000602      PERFORM 9000-OUTPUT-SCREEN
000603      END-IF
000604      *
000605      *****
000606      *                INITIALIZATION ROUTINE
000607      *****
000608
=====
M038: Breakpoint has been set at INVO701+0D12
==>
4.B          :00.3          15/06

```

STEP 4

Run the transaction and Debug – Now the program is ready for debugging. Come out of Tracemaster menu (use exit, not Log-off) and type the transaction id to run the program to be tested. After the encounter of first break in the program the Tracemaster screen would be displayed. From here you can execute the program in debugging mode. You can press PF1 and use the help for PF keys and commands to be used. Some of the basic functionalities:

- For executing line by line use PF12.
- Run/conditional run by PF10
- To see the value of a variable, place the cursor on it and press PF6

STEP 5

Log off – Enter 'Abend' command to terminate the program

- Abend x(any arbitrary character)
- 'End' command would only return back to the trace master menu.
- Remember to log off from trace master to free the resources

Debugging a COBOL batch program (in Foreground)

STEP 1

Compile and link the program for trace master – It would be same as that for CICS program. You need to make same changes in the compilation JCL procedure.

STEP 2

Set Default libraries – Open trace master menu by typing 'TSO TRMAST' command or using ISPF options(For example:option A.16 from ISPF panel brings the Tracemaster menu)

```

SESSION2 - EXTRA!® Enterprise
File Edit View Tools Session Options Help
TraceMaster - Primary Menu

Option ==>

0 Options          Set user options          User ID . : VN080
1 Foreground      Trace Batch Jobs in TSO Foreground Time. . . : 05:44
2 Background      Trace Batch Jobs in Background Terminal. : 3278
3 IMS/DC          Trace IMS/MPPs           Screen. . : 1
X Exit            Exit TraceMaster/ISPF    Language. : ENGLISH
                                     Appl ID . : M4TM
                                     TSO logon : TSOUSER
                                     TSO prefix: VN080
                                     System ID : CPU1
                                     MVS acct. : NONE
                                     Release . : ISPF 5.2

Enter X to terminate this menu

TraceMaster - Version 4.300b

Software Excellence from Macro 4
http://www.macro4.com
All trademarks acknowledged
(c) copyright 2003, Macro 4. All rights reserved.

PF 1=HELP      2=SPLIT      3=END      4=RETURN      5=RFIND      6=RCHANGE
PF 7=UP        8=DOWN       9=SWAP     10=LEFT      11=RIGHT     12=RETRIEVE

4B  :05.0 02/14

```

- First time users should set the default libraries using option 0(OPTIONS)on the menu.ASR,profile and macros libraries are mandatory. For setting a library enter the corresponding number which would display the following screen.

```

SESSION2 - EXTRA!® Enterprise
File Edit View Tools Session Options Help
TraceMaster - Options Menu

Option ==> 2

1 General Options      Set TraceMaster general options
2 ASR Library          Specify/Allocate ASR library
3 Test Profiles Library Specify/Allocate test profiles library
4 Macro Library        Specify/Allocate default macro library
5 Symbols              Specify symbols options
6 Terminal I/O File    Specify/Allocate default TIO file
7 Select IMS Id        select IMS Id for IMS/MPP testing
                       Specify/Allocate ASR Library

CU
Command ==>
Data Set Name ==> 'VN080.XTASR'

PF 1=HELP      2=SPLIT      3=END      4=RETURN      5=RFIND
PF 6=RCHANGE    7=UP         8=DOWN     9=SWAP        10=LEFT

PF 1=HELP      2=SPLIT      3=END      4=RETURN      5=RFIND      6=RCHANGE
PF 7=UP        8=DOWN       9=SWAP     10=LEFT      11=RIGHT     12=RETRIEVE

4B  :01.8 15/27

```

Here enter the dataset name as '*prefix*.XTASR' or simply XTASR. If the quotes are dropped, the prefix would be the default prefix. The default prefix could be changed using option 1(General Options) which is by default your userid. If the dataset does not exist then a screen for new dataset creation would appear:

```

SESSION2 - EXTRA@ Enterprise
File Edit View Tools Session Options Help
TraceMaster - Allocate New Data Set

Command ==>

Data set name . . . . : 'VN080.MY.XTASR'
Data set description : ASR Library
Management class . . . _____ (Blank for default management class)
Storage class . . . . _____ (Blank for default storage class)
Volume serial . . . . _____ (Blank for system default volume) **
Device type . . . . _____ (Generic unit or device address) **
Data class . . . . _____ (Blank for default data class)
Space units . . . . _____ (BLKS, TRKS, CYLS or RECORDS)
Average record unit . _____ (M, K, or U)
Primary quantity . . _____ (In above units)
Secondary quantity . . _____ (In above units)
Directory blocks . . _____ (Number of directory blocks) *
Record format . . . . : FB
Record length . . . . : 80
Block size . . . . _____ (Maximum = 32720)
Data set name type . . _____ (LIBRARY, PDS, or blank) *

( * Specifying LIBRARY overrides directory blocks)
( ** Only one of these fields may be specified)
PF 1=HELP      2=SPLIT      3=END      4=RETURN      5=RFIND      6=RCHANGE
PF 7=UP        8=DOWN        9=SWAP     10=LEFT       11=RIGHT     12=RETRIEVE

4B  :02.2 06/25

```

You need to fill in the required information. After setting all the libraries, the libraries would be shown in current library definitions.

```

SESSION2 - EXTRA@ Enterprise
File Edit View Tools Session Options Help
TraceMaster - Options Menu Invalid JOB Card

option ==>

1 General Options      Set TraceMaster general options
2 ASR Library          Specify/Allocate ASR library
3 Test Profiles Library Specify/Allocate test profiles library
4 Macro Library        Specify/Allocate default macro library
5 Symbols              Specify symbols options
6 Terminal I/O File    Specify/Allocate default TIO file
7 Select IMS Id        select IMS Id for IMS/MPP testing

current library definitions:
ASR . . . . . 'VN080.XTASR'
Test Profiles . . 'VN080.XTPROF'
Macro . . . . . 'VN080.XTMACLIB'
TIOfile . . . . .

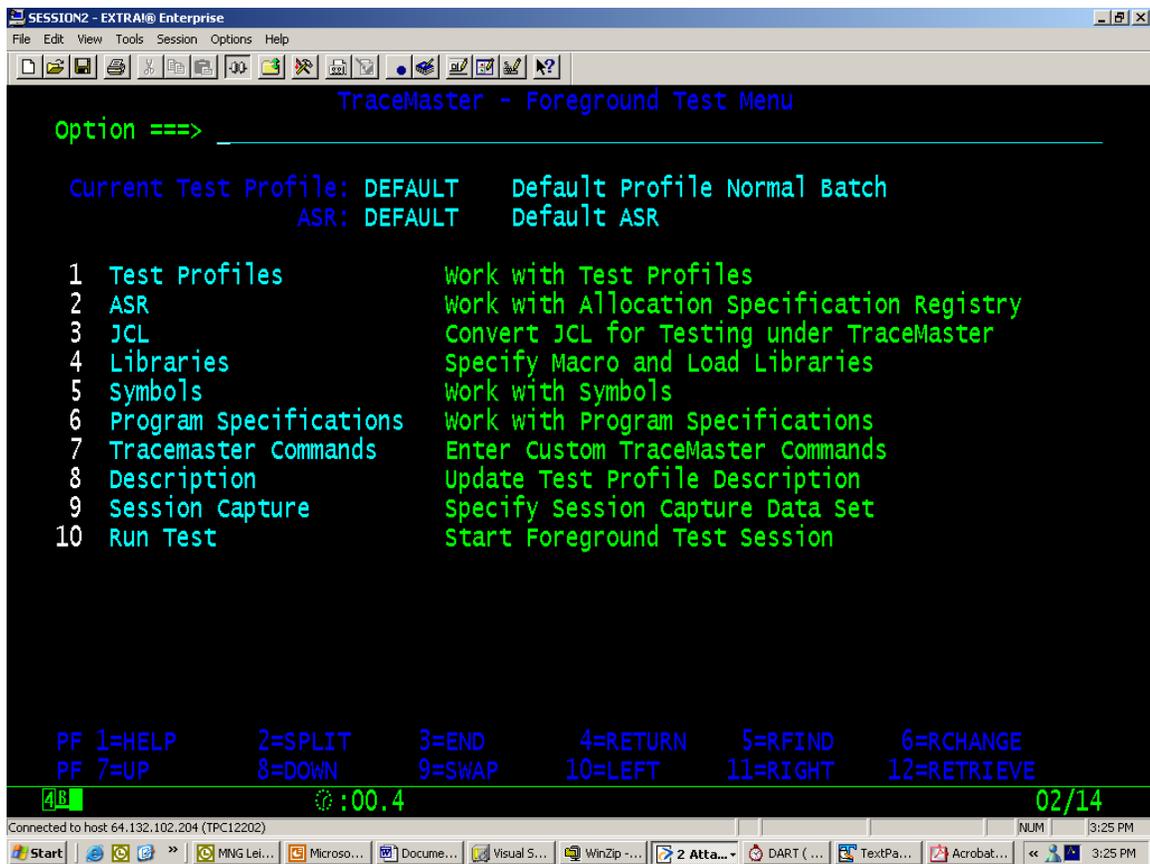
PF 1=HELP      2=SPLIT      3=END      4=RETURN      5=RFIND      6=RCHANGE
PF 7=UP        8=DOWN        9=SWAP     10=LEFT       11=RIGHT     12=RETRIEVE

4B  :00.5 02/14

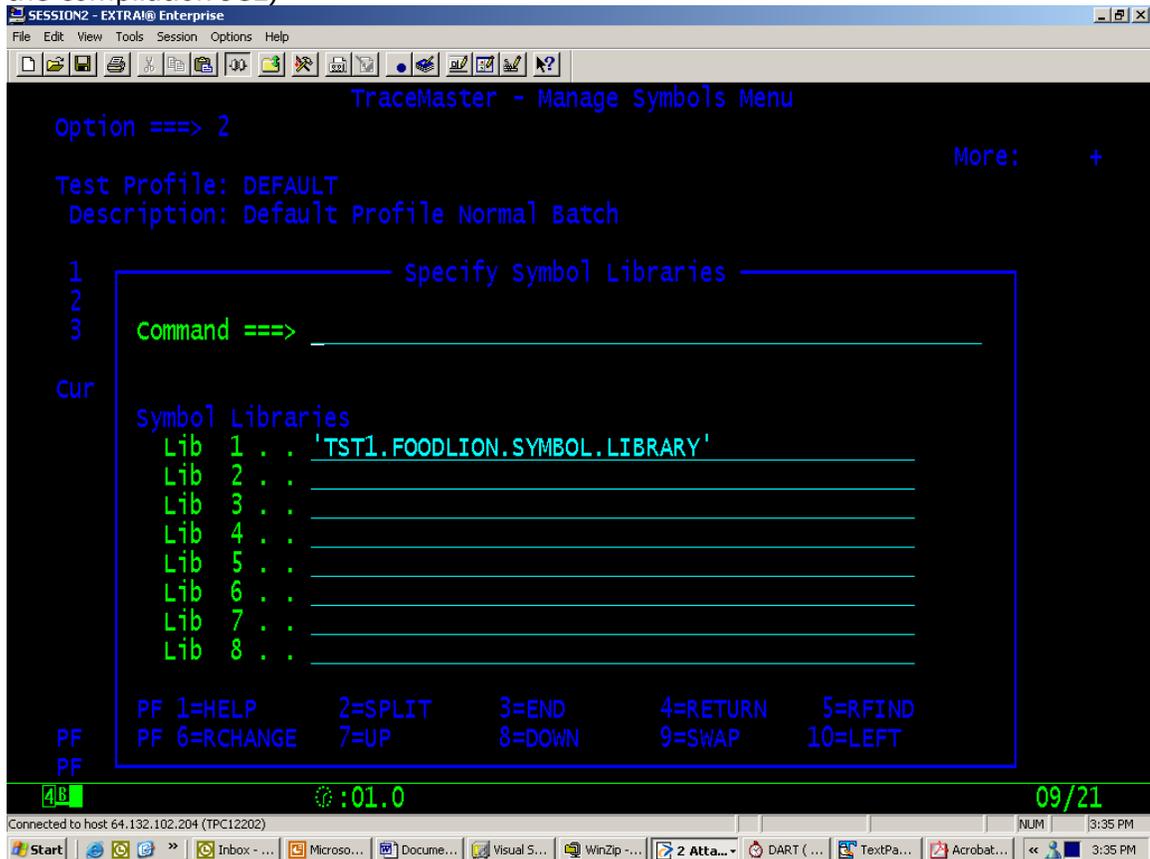
```

STEP 3

Set symbol library and Convert the JCL – Now go back to the main menu and enter the option for foreground testing.



Here select option 5(Symbols) and give the symbol library name(same as used in the compilation JCL)



After setting the symbol library use option 3 for converting the JCL which executes the program for Tracemaster. Specify the member and the PDS and press enter to get the following screen.

```

SESSION2 - EXTRA Enterprise
File Edit View Tools Session Options Help
TraceMaster - Step Selection List Row 1 to 13 of 16
Command ==> Scroll ==> PAGE

Line commands: B - Browse Step          S - Select for Tracing
                X - Deselect

Program Name  Step Name  Procedure Step Name  Procedure Name  Status
-----
IDCAMS        STEP010
MS0888        STEP010X
AFCP2016      STEP020
MS0888        STEP020X
SORT          STEP030
MS0888        STEP30X
SORT          STEP040
MS0888        STEP40X
SORT          STEP050
MS0888        STEP50X
IDCAMS        STEP060
MS0888        STEP060X
IDCAMS        STEP070
PF 1=HELP    2=SPLIT    3=END      4=RETURN    5=RFIN    6=RCHANGE
PF 7=UP      8=DOWN     9=SWAP     10=LEFT     11=RIGHT   12=RETRIEVE

4B :02.7 02/15

```

It would list all the steps of the JCL. Select the program to be debugged by putting S in front of the step executing the program and press PF3.

STEP 4

Run in debug mode - : Now press option 10 (Run Test) to run the program in debug mode. The screen with source code would be displayed.

```

SESSION2 - EXTRA Enterprise
File Edit View Tools Session Options Help
+==SOURCE(C)===== LIONDATC =====+
000068      10 FILLER PIC X(10) VALUE X'202E3C282B7C2621242A'.
000069      10 FILLER PIC X(10) VALUE X'293B2D2F2C255F3E3F79'.
000070      10 FILLER PIC X(10) VALUE X'3A2340273D225B5D7B7D'.
000071      10 FILLER PIC X(3)  VALUE X'5C7E5E'.
000072      01 ASCII-INFO REDEFINES ASCII-DATA PIC X(95).
000073      01 ASCII-TABLE REDEFINES ASCII-DATA.
000074      05 ASCII-BYTE PIC X OCCURS 95 TIMES.
000075      01 FILLER PIC X(17) VALUE
000076      'LIONDATC WS-EN
>>>>>  PROCEDURE DIVISION.
000078      *
000079      *****
000080      *
000081      *
000082      *
000083      *
000084      0100-INITIALIZATION.
000085      *
000086      MOVE 'NR' TO WS-VAR1
000087      MOVE WS-VAR1(1:2) TO WS-VAR2(1:2)
+=====+
M076: symbol information successfully loaded
==>
4B :01.6 24/06

```

Here, you can set the breakpoints and start executing the program by pressing PF12 or typing RUN command on command line. Other options are same as those in CICS.

After the testing is over, exit by typing 'END' command.