Z390/CICS Application Programmers Guide

The EXEC CICS commands and parameters listed here are the only ones currently supported in the Z390/CICS environment.

Differences between mainframe operation and Z390/CICS are explained.

For the operation of each command and parameter please refer to the Manuals listed in the Reference section at the end of this document.

If you create your own Z390/CICS applications, it would be wise to create a .BAT file that re-assembles them all in one go. The internal interfaces are volatile at present and this will be a frequent instruction. The current Z390/CICS environment and all test programs can be re-assembled using DFHALL.BAT

Assembly notes

CICS must be added as an option to CALL MZ390.
PROLOG and EPILOG are defaults.

NOEPILOG is supported but not fully tested, testing and correct usage is scheduled for a future release of Z390/CICS.

PROLOG inserts the following:

```
  DFHEIGBL -- Defines the basic set of control block DSECTs.
  DFHEISTG -- Define the prefix areas of the Dynamic Storage Area (DSA).
  DFHEIENT -- Replaces the first CSECT statement.
              Establish linkage and base registers.
              GETMAIN the DSA.
              Establish addressability to the EIB and TCTTE.
              Some COMMAREA management.
  DFHEIEND -- Replaces the END statement and defines the end of the DSA.
```

Other macros...

```
  DFHABBLK -- HANDLE ABEND block DSECT
  DFHADBLK -- HANDLE AID block DSECT
  DFHEIRET -- Part of EXEC CICS RETURN
              Some clean-up operations.
              FREEMAIN the DSA.
              Manage link-level and return to last linker.
  DFHEIBLK -- EIB DSECT
  DFHHCBLK -- HANDLE CONDITION block DSECT
  DFHPCT   -- Transaction definition
  DFHTCTTE -- TCTTE DSECT
  EXEC     -- Converts EXEC CICS statements into a unique macro call
              with a parameter list.
```

Copy books...

```
   DFHAID     Standard CICS equates for AID keys
   DFHPCTUS   User transaction codes
```

Register Usage

R0	Reserved for internal use
R1	Reserved for internal use
R2	Reserved for internal use
R10	TCTTE address, must not be modified
R11	EIB address, must not be modified
R12	Default base register
R13	DSA address, must not be modified

```
R14  Reserved for internal use
R15  Reserved for internal use
```

Multiple base registers

The standard entry for a CICS program is as follows:

```
eg.
DFHEISTG DSECT
MYFIELD  DS    CL100              demo user field
...
MYPROG   CSECT
```

This standard method with the PROLOG option (default) will generate a single code base of R12 and a single DSA base of R13.

If you want to extend the code base and/or the DSA base registers, convert your code in line with the sample given, and include the NOPROLOG option in CALL MZ390.

```
eg.
         DFHEIGBL
         DFHEISTG
MYFIELD  DS    CL100              demo user field
...
MYPROG   DFHEIENT CODEREG=(R8,R5),DATAREG=(R13,R6,R7)
```

```
Notes: You cannot override the first DATAREG value,
       it will always be R13. ie. if you code DATAREG=(R6,R7)
       you will get DATAREG=(R13,R7).

       There is no cross-checking for register conflicts.
```

Assembler Notes

Inclusion of the macro EQUREGS is mandatory.

Supported EXEC CICS commands (summary)

```
 1) ABEND ABCODE() CANCEL NODUMP
 2) FREEMAIN DATA() DATAPOINTER()
 3) GETMAIN SET() LENGTH() FLENGTH() INITIMG()
 4) HANDLE ABEND CANCEL/RESET/LABEL()/PROGRAM()
 5) HANDLE AID key() key
 6) HANDLE CONDITION condition() condition
 7) IGNORE CONDITION condition
 8) LINK PROGRAM() COMMAREA() LENGTH()
 9) LOAD PROGRAM() ENTRY() SET() LENGTH() FLENGTH()
10) POP HANDLE
11) PUSH HANDLE
12) RECEIVE INTO() LENGTH() NOHANDLE
13) RELEASE PROGRAM()
14) RETURN TRANSID() COMMAREA() LENGTH()
15) SEND FROM() LENGTH()
16) XCTL PROGRAM() COMMAREA() LENGTH()
```

Supported EXEC CICS commands (detail)

```
 1) ABEND ABCODE() CANCEL NODUMP

       ABCODE can be specified as ABCODE('xxxx') or ABCODE(label)
       label must point to a 4-byte field.
```

```
   Errors
      ABCODE MUST NOT BEGIN WITH 'A'
      ABCODE IS INVALID
      BAD PARM

2) FREEMAIN DATA() DATAPOINTER()

      DATA(label)
         label may only be an indirect reference to the address.

      DATAPOINTER
         Must be specified as a permitted general register value.

   Errors
      BAD PARM
      DATA AND DATAPOINTER SPECIFIED
      DATA OR DATAPOINTER IS MANDATORY

   Conditions (RESP/RESP2)
      INVREQ/1   Can be IGNOREd

3) GETMAIN SET() LENGTH() FLENGTH() INITIMG()

      SET is mandatory
         Must be specified as a permitted general register value.

      LENGTH
         Can be specified as a constant, literal or label.
         A constant must not exceed 32767.
         A literal or label must be 2 bytes and must not exceed 32767.

      FLENGTH
         Can be specified as a constant, literal or label.
         A constant must not exceed 2G-1.
         A literal or label must be 4 bytes and must not exceed 2G-1.

      INITIMG is optional
         If omitted, the storage contents are not predictable.
         Can be specified as a constant, literal or label.
         Only the first byte generated by the parameter is used.

   Errors
      BAD PARM
      LENGTH AND FLENGTH SPECIFIED
      LENGTH OR FLENGTH IS MANDATORY
      SET IS MANDATORY

4) HANDLE ABEND CANCEL
   HANDLE ABEND RESET

   HANDLE ABEND LABEL(label)
      label may take three forms:
         Direct reference
         Indirect reference
         Adcon literal

   HANDLE ABEND PROGRAM()
      Can be specified as PROGRAM('xxxxxxxx') or PROGRAM(label)
      label must point to an 8-byte field.

      Any received COMMAREA when the EXEC CICS HANDLE ABEND is issued
      is passed to the handling program when an abend occurs.
```

Notes: When an XCTL is executed, any HANDLE ABEND LABEL at the current logical level is cleared as the current program is no longer in use. HANDLE ABEND PROGRAMs are not cleared.

Errors
- BAD PARM
- INVALID PROGRAM
- HANDLE TYPE NOT RECOGNISED
- PARMS MISSING OR TOO MANY PARMS

5) HANDLE AID key(label) key etc.

The following parameters are not supported...
CLRPARTN, LIGHTPEN, OPERID, TRIGGER

The manual is not clear about ANYKEY (no label). I have assumed that it clears all settings for CLEAR, PA and PF keys.

label may take three forms:
- Direct reference
- Indirect reference
- Adcon literal

```
eg. EXEC CICS HANDLE AID PA1(GOPA1) PA2(INDGOPA1) PA3(=A(GOPA1))
     ...
     GOPA1    DS 0H
     ...
     INDGOPA1 DC A(GOPA1)
```

Errors
- BAD PARM
- HANDLE TYPE NOT RECOGNISED

6) HANDLE CONDITION condition(label) condition etc.

Only the following conditions are supported:
ERROR, INVREQ, LENGERR, PGMIDERR

label may take three forms:
- Direct reference
- Indirect reference
- Adcon literal

See HANDLE AID for examples of these forms.

Errors
- BAD PARM
- HANDLE TYPE NOT RECOGNISED

7) IGNORE CONDITION condition ...

Only the following conditions are supported:
ERROR, INVREQ, LENGERR, PGMIDERR

Notes:
Ignoring an error may lead to unpredictable abends.
Advice for the asking.

PGMIDERR or ERROR by default...
The EXEC CICS command treated as never existed.

INVREQ or ERROR by default...
EXEC CICS RELEASE never existed.

```
         EXEC CICS RETURN...
            This will abend the task ASRA as I cannot ignore a RETURN.

       LENGERR or ERROR by default...
         Only on EXEC CICS RECEIVE.
         NOHANDLE and any outstanding HANDLE AID will not invoke this
         condition.

         Testing is not complete for this condition.

   Errors
      BAD PARM
      IGNORE TYPE NOT RECOGNISED

8) LINK PROGRAM() COMMAREA(label) LENGTH()

      Executes another CICS program.
      If COMMAREA is present, the address/length are passed.
      Return is to the linker.

      PROGRAM is mandatory
         Can be specified as PROGRAM('xxxxxxxx') or PROGRAM(label)
         label must point to an 8-byte field.

      COMMAREA(label) is optional
         label may take three forms:
            Direct reference
            Indirect reference
            Adcon literal

      LENGTH
         Can be specified as LENGTH(value) or LENGTH(label)
         LENGTH(value) supports the use of the length attribute.
         label must point to a 2-byte hex value.

         LENGTH can be omitted. When it is, the implied length of the
         COMMAREA is used. LENGTH is mandatory when COMMAREA is an
         indirect reference.

   Errors
      BAD PARM
      INVALID PROGRAM
      PROGRAM IS MISSING
      LENGTH IS MANDATORY FOR INDIRECT COMMAREA
      LENGTH WITHOUT COMMAREA

   Conditions (RESP/RESP2)
      PGMIDERR/3   Can be IGNOREd

9) LOAD PROGRAM() ENTRY() SET() LENGTH(label) FLENGTH(label)

      Loads a module.
      The intention in the Z390/CICS environment is to load a table
      or some other data, not an executable program.

      PROGRAM is mandatory
         Can be specified as PROGRAM('xxxxxxxx') or PROGRAM(label)
         label must point to an 8-byte field.

         At present, only modules with a suffix of .390 may be LOADed.

      ENTRY and SET are optional
         Must be specified as a permitted general register value.
         Both are equivalent in Z390/CICS.

      LENGTH is optional
         LENGTH(label) is the only format.
         label must point to a 2-byte field.

      FLENGTH is optional
         FLENGTH(label) is the only format.
         label must point to a 4-byte field.

   Note: At task end the LOADed module is not RELEASEd.

   Errors
      BAD PARM
      INVALID PROGRAM
      LENGTH AND FLENGTH SPECIFIED
      PROGRAM IS MISSING

   Conditions (RESP/RESP2)
      PGMIDERR/3   Can be IGNOREd

10) POP HANDLE

   For the HANDLE ABEND, a POP is the equivalent of a HANDLE ABEND
   RESET.

   Errors
      POP TYPE NOT RECOGNISED

   Conditions (RESP/RESP2)
      INVREQ/0     Can be IGNOREd

11) PUSH HANDLE

   For the HANDLE ABEND, a PUSH is the equivalent of a HANDLE ABEND
   CANCEL.

   Errors
      PUSH TYPE NOT RECOGNISED

12) RECEIVE INTO(label) LENGTH(label) NOHANDLE

   INTO(label) and LENGTH(label) are mandatory.
   LENGTH must point to a 2-byte field.

   Although MAXLENGTH is not implemented yet, there is an internal
   maximum length set to the implied length of the INTO label.

   NOHANDLE is optional.

   Errors
      BAD PARM
      BOTH INTO AND LENGTH ARE REQUIRED
      LENGTH ERROR

   Conditions (RESP/RESP2)
      LENGERR/0    Can be IGNOREd

13) RELEASE PROGRAM()
      Releases a previously LOADed module.

      PROGRAM is mandatory
         Can be specified as PROGRAM('xxxxxxxx') or PROGRAM(label)
         label must point to an 8-byte field.

   Errors
      BAD PARM
      INVALID PROGRAM
      PROGRAM IS MISSING

   Conditions (RESP/RESP2)
      INVREQ/5     Can be IGNOREd
      INVREQ/6     Can be IGNOREd

14) RETURN TRANSID() COMMAREA(label) LENGTH()
      Returns to the last caller.

      TRANSID
         Optional, but when COMMAREA is specified, TRANSID is mandatory.
         Can be specified as TRANSID('xxxx') or TRANSID(label)
         label must point to a 4-byte field.

      COMMAREA(label) is optional
         label may take three forms:
            Direct reference
            Indirect reference
            Adcon literal

      LENGTH
         Can be specified as LENGTH(value) or LENGTH(label)
         LENGTH(value) supports the use of the length attribute.
         label must point to a 2-byte hex value.

         LENGTH can be omitted. When it is, the implied length of the
         COMMAREA is used. LENGTH is mandatory when COMMAREA is an
         indirect reference.

   Errors
      BAD PARM
      INVALID TRANSID
      TRANSID IS MISSING
      LENGTH IS MANDATORY FOR INDIRECT COMMAREA
      LENGTH WITHOUT COMMAREA

   Conditions (RESP/RESP2)
      INVREQ/2     Cannot be IGNOREd

15) SEND FROM(label) LENGTH()

      FROM(label) is mandatory.
         label may take three forms:
            Direct reference
            Indirect reference
            Adcon literal

      LENGTH
         Can be specified as LENGTH(value) or LENGTH(label)
         LENGTH(value) supports the use of the length attribute.
         label must point to a 2-byte hex value.

   Errors
      BAD PARM
      FROM IS MANDATORY
      LENGTH IS MANDATORY

16) XCTL PROGRAM() COMMAREA(label) LENGTH()
```

Executes another CICS program.

If COMMAREA is present and both the address and length are the same as passed to the current program, then address/length are passed to the new program.

If the address or length differs, then a copy of the COMMAREA is taken and the new address/length are passed to the new program.

Return is to the last linker.

PROGRAM is mandatory
- Can be specified as PROGRAM('xxxxxxxx') or PROGRAM(label)
- label must point to an 8-byte field.

COMMAREA(label) is optional
- label may take three forms:
  - Direct reference
  - Indirect reference
  - Adcon literal

LENGTH
- Can be specified as LENGTH(value) or LENGTH(label)
- LENGTH(value) supports the use of the length attribute.
- label must point to a 2-byte hex value.

  LENGTH can be omitted. When it is, the implied length of the COMMAREA is used. LENGTH is mandatory when COMMAREA is an indirect reference.

Errors
- BAD PARM
- INVALID PROGRAM
- PROGRAM IS MISSING
- LENGTH IS MANDATORY FOR INDIRECT COMMAREA
- LENGTH WITHOUT COMMAREA

Conditions (RESP/RESP2)
- PGMIDERR/3   Can be IGNOREd

Appendicies

Keypress information

Aid	Press
ENTER	Enter or Return
CLEAR	CTRL+C
PA1-PA3	CTRL+F1 to CTRL+F3
PF1-PF12	F1 to F12
PF13-PF24	CTRL+ALT+F1 to CTRL+ALT+F12

References

SC34-6433 CICS Application Programmers Guide
SC34-6434 CICS Application Programmers Reference

Trademarks

CICS is a registered trademark of International Business Corporation.

Author: Melvyn Maltz
Shipping Date: June 25, 2007

Z390 version: V1.3.04c
Z390/CICS version: V2
→