Z390/CICS Application Programming 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. The test VSAM catalog and files can be rebuilt using DFHALLV.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).
  DFHEIEND -- Replaces the END statement and defines the end of the 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

Other macros...
  DFHEIBLK -- EIB DSECT
  DFHEIRET -- Part of EXEC CICS RETURN
              Some clean-up operations

```
               FREEMAIN the DSA
               Manage link-level and return to last linker
  DFHPCT   -- Transaction definition
  DFHFCT   -- File definitions and options
  EXEC     -- Converts EXEC CICS statements into a unique macro call
               with a parameter list

Copy books...
   DFHAID     Standard CICS equates for AID keys
   DFHBMSCA   Mapping support equates
   DFHPCTUS   User transaction codes
   DFHFCTUS   User file definitions

Inclusion of the macro EQUREGS is mandatory.

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.

VSAM Notes

See the Z390/CICS VSAM Guide for guidance in the setup of a VSAM environment. This document also contains extensions to the VSAM facilities currently available.

Basic Mapping Support

The commands RECEIVE MAP, SEND MAP and SEND CONTROL are documented here.
For general BMS documentation and the mapping macros DFHMSD, DFHMDI and
DFHMDF see the Z390/CICS BMS Guide.

Supported EXEC CICS commands (summary by type)

1) General commands (02)
   a) HANDLE AID key() key
   b) HANDLE CONDITION condition() condition
   c) IGNORE CONDITION condition
   d) POP HANDLE
   e) PUSH HANDLE

2) Terminal Control (04)
   a) RECEIVE INTO() LENGTH() NOHANDLE
   b) SEND FROM() LENGTH()

3) File Control (06)
   a) READ FILE()/DATASET() INTO()/SET() LENGTH()/FLENGTH()
      RIDFLD() RBA/XRBA/RRN/GENERIC GTEQ/EQUAL KEYLENGTH()
   b) STARTBR FILE()/DATASET() RIDFLD() REQID()
      RBA/XRBA/RRN/GENERIC GTEQ/EQUAL KEYLENGTH()
   c) READNEXT FILE()/DATASET() INTO()/SET() LENGTH()/FLENGTH()

```
      RIDFLD() REQID() RBA/XRBA/RRN KEYLENGTH()
   d) READPREV FILE()/DATASET() INTO()/SET() LENGTH()/FLENGTH()
      RIDFLD() REQID() RBA/XRBA/RRN KEYLENGTH()
   e) ENDBR FILE()/DATASET() REQID()
   f) RESETBR FILE()/DATASET() RIDFLD() REQID()
      RBA/XRBA/RRN/GENERIC GTEQ/EQUAL KEYLENGTH()

4) Storage Control (08)
   a) FREEMAIN DATA()/DATAPOINTER()
   b) GETMAIN SET() LENGTH()/FLENGTH() INITIMG()

5) Temporary Storage Control (0A)
   a) DELETEQ TS QUEUE()/QNAME()
   b) READQ TS QUEUE()/QNAME() INTO()/SET() LENGTH() NUMITEMS()
      ITEM()/NEXT
   c) WRITEQ TS QUEUE()/QNAME() FROM() LENGTH() NUMITEMS() ITEM()
      REWRITE

6) Program Control (0E)
   a) ABEND ABCODE() CANCEL NODUMP
   b) HANDLE ABEND CANCEL/RESET/LABEL()/PROGRAM()
   c) LINK PROGRAM() COMMAREA() LENGTH()
   d) LOAD PROGRAM() ENTRY()/SET() LENGTH() FLENGTH()
   e) RELEASE PROGRAM()
   f) RETURN TRANSID() COMMAREA() LENGTH()
   g) XCTL PROGRAM() COMMAREA() LENGTH()

7) Interval Control (10 and 4A)
   a) ASKTIME ABSTIME()
   b) DELAY INTERVAL()/TIME() FOR/UNTIL HOURS() MINUTES() SECONDS()
   c) FORMATTIME   all parameters

8) BMS (18)
   a) RECEIVE MAP(name) MAPSET(name)
   b) SEND MAP(name) MAPSET(name)
      CURSOR/CURSOR() DATAONLY/MAPONLY ERASE/ERASEAUP
      ALARM FREEKB FRSET
   c) SEND CONTROL CURSOR/CURSOR() ERASE/ERASEAUP
      ALARM FREEKB FRSET

9) Dump Control (1C)
   DUMP TRANSACTION DUMPCODE() COMPLETE
   FROM() LENGTH()/FLENGTH()
   SEGMENTLIST() LENGTHLIST() NUMSEGMENTS()
```

```
Supported EXEC CICS commands (detail by type)
---------------------------------------------
 1) General commands (02)

    a) 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

    b) HANDLE CONDITION condition(label) condition etc.

       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

    c) IGNORE CONDITION condition ...

       Notes:
          Ignoring an error may lead to unpredictable abends.
```

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

INVREQ on EXEC CICS RETURN will abend the task ASRA as I cannot ignore a RETURN.

LENGERR or ERROR by default...
NOHANDLE and any outstanding HANDLE AID will not invoke this condition.

Errors
BAD PARM
IGNORE TYPE NOT RECOGNISED

d) 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

e) PUSH HANDLE

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

Errors
PUSH TYPE NOT RECOGNISED

2) Terminal Control (04)

a) 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

b) 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

3) File Control (06)

Notes:
FLENGTH and XRBA are extensions, do not use these parameters if the source code is likely to be ported back to a mainframe environment.

RBA access to a KSDS is not supported.

DATASET is supported for legacy applications. It is noted that this parameter no longer appears in the Manuals.

When conditions are raised as a result of a VSAM error, the RPL feedback codes (2nd and 4th bytes) are placed in EIBRCODE +1 and +2.

a) READ FILE()/DATASET() INTO()/SET() LENGTH()/FLENGTH()

RIDFLD() RBA/XRBA/RRN/GENERIC GTEQ/EQUAL KEYLENGTH()

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.

LENGTH/FLENGTH notes:
If not a label then ...
If INTO is specified, then the length received is the implied length of INTO. This may raise the LENGERR condition if the data length is larger.

If SET is specified, the complete record is returned and LENGERR cannot occur.

If it is a label then ...
If INTO or SET is specified, then it specifies the maximum data length that can be received. LENGERR can be raised if the data length is larger. The true data length is returned in label.

KEYLENGTH
Can be specified as a constant or label.
A constant must not exceed 32767.
A label must be 2 bytes and must not exceed 32767.
Keylengths greater than 128 are ignored.

The parameter is ignored for ESDS and RRDS.
KEYLENGTH and GENERIC must be paired.

If KEYLENGTH is zero by constant or label then parameters are changed internally:
GENERIC/EQUAL or GENERIC/GTEQ
Becomes KEYLENGTH(1) Key=X'00' GENERIC GTEQ

RBA   RIDFLD has a 4-byte RBA
XRBA RIDFLD has an 8-byte RBA
RRN   RIDFLD has a 4-byte relative record number

GTEQ/EQUAL
   The parameter is ignored for ESDS and RRDS.

GENERIC
   KEYLENGTH must be specified.
   The parameter is ignored for ESDS and RRDS.

Errors
   BAD PARM
   BOTH FILE AND DATASET ARE SPECIFIED
   BOTH GTEQ AND EQUAL ARE SPECIFIED
   BOTH INTO AND SET ARE SPECIFIED
   BOTH LENGTH AND FLENGTH ARE SPECIFIED
   BOTH RBA AND XRBA ARE SPECIFIED
   BOTH RRN AND (X)RBA ARE SPECIFIED
   FILE OR DATASET MUST BE SPECIFIED
   GENERIC CANNOT BE SPECIFIED WITH RRN OR (X)RBA
   GENERIC REQUIRES KEYLENGTH
   INTO OR SET MUST BE SPECIFIED
   INVALID FILE OR DATASET
   KEYLENGTH REQUIRES GENERIC
   RIDFLD IS MANDATORY

Conditions (RESP/RESP2)
   FILENOTFOUND/1
   DISABLED/50
   ILLOGIC/110
   INVREQ/20
   INVREQ/25
   INVREQ/42
   LENGERR/E1
   NOTFND/80
   NOTOPEN/60

b) STARTBR FILE()/DATASET() RIDFLD() REQID() RBA/XRBA/RRN/GENERIC
   GTEQ/EQUAL KEYLENGTH()

REQID
   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.
If omitted, zero is assumed.

RBA RIDFLD has a 4-byte RBA
XRBA RIDFLD has an 8-byte RBA
RRN RIDFLD has a 4-byte relative record number

KEYLENGTH
Can be specified as a constant or label.
A constant must not exceed 32767.
A label must be 2 bytes and must not exceed 32767.
Keylengths greater than 128 are ignored.

The parameter is ignored for ESDS and RRDS.
KEYLENGTH and GENERIC must be paired.

If KEYLENGTH is zero by constant or label then parameters are
changed internally:
GENERIC/EQUAL or GENERIC/GTEQ
Becomes KEYLENGTH(1) Key=X'00' GENERIC GTEQ

GTEQ/EQUAL
The parameter is ignored for ESDS and RRDS.

GENERIC
KEYLENGTH must be specified.
The parameter is ignored for ESDS and RRDS.

Errors
BAD PARM
BOTH FILE AND DATASET ARE SPECIFIED
BOTH GTEQ AND EQUAL ARE SPECIFIED
BOTH RBA AND XRBA ARE SPECIFIED
BOTH RRN AND (X)RBA ARE SPECIFIED
FILE OR DATASET MUST BE SPECIFIED
GENERIC CANNOT BE SPECIFIED WITH RRN OR (X)RBA
GENERIC REQUIRES KEYLENGTH
INVALID FILE OR DATASET
KEYLENGTH REQUIRES GENERIC
RIDFLD IS MANDATORY

Conditions (RESP/RESP2)
DISABLED/50
FILENOTFOUND/1

ILLOGIC/110
INVREQ/20
INVREQ/25
INVREQ/33
INVREQ/42
NOTFND/80
NOTOPEN/60

Note: NOTFND cannot occur for an ESDS or RRDS

c) READNEXT FILE()/DATASET() INTO()/SET() LENGTH()/FLENGTH() RIDFLD() REQID() RBA/XRBA/RRN KEYLENGTH()

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.

LENGTH/FLENGTH notes:
If either is not a label then ...
If INTO is specified, then the length received is the implied length of INTO. This may raise the LENGERR condition if the data length is larger.

If SET is specified, the complete record is returned and LENGERR cannot occur.

If either is a label then ...
If INTO or SET is specified, then it specifies the maximum data length that can be received. LENGERR can be raised if the data length is larger. The true data length is returned in label.

REQID
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.
          If omitted, zero is assumed.

       RBA   RIDFLD has a 4-byte RBA
       XRBA  RIDFLD has an 8-byte RBA
       RRN   RIDFLD has a 4-byte relative record number

       KEYLENGTH
          Can be specified as a constant or label.
          A constant must not exceed 32767.
          A label must be 2 bytes and must not exceed 32767.
          Keylengths greater than 128 are ignored.

          The parameter is ignored for ESDS and RRDS.

          If KEYLENGTH is zero by constant or label then parameters
are
          changed internally:
             GENERIC/EQUAL or GENERIC/GTEQ
                Becomes KEYLENGTH(1) Key=X'00' GENERIC GTEQ

       Errors
          BAD PARM
          BOTH FILE AND DATASET ARE SPECIFIED
          BOTH INTO AND SET ARE SPECIFIED
          BOTH LENGTH AND FLENGTH ARE SPECIFIED
          BOTH RBA AND XRBA ARE SPECIFIED
          BOTH RRN AND (X)RBA ARE SPECIFIED
          FILE OR DATASET MUST BE SPECIFIED
          INTO OR SET MUST BE SPECIFIED
          INVALID FILE OR DATASET
          RIDFLD IS MANDATORY

       Conditions (RESP/RESP2)
          DISABLED/50
          ENDFILE/90
          FILENOTFOUND/1
          ILLOGIC/110
          INVREQ/20
          INVREQ/25
          INVREQ/26
          INVREQ/34
          INVREQ/42
          LENGERR/E1
          NOTFND/80
```

NOTOPEN/60

c) READPREV FILE()/DATASET() INTO()/SET() LENGTH()/FLENGTH()
RIDFLD() REQID() RBA/XRBA/RRN KEYLENGTH()

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.

LENGTH/FLENGTH notes:
If either is not a label then ...
If INTO is specified, then the length received is the implied length of INTO. This may raise the LENGERR condition if the data length is larger.

If SET is specified, the complete record is returned and LENGERR cannot occur.

If either is a label then ...
If INTO or SET is specified, then it specifies the maximum data length that can be received. LENGERR can be raised if the data length is larger. The true data length is returned in label.

REQID
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.
If omitted, zero is assumed.

RBA RIDFLD has a 4-byte RBA
XRBA RIDFLD has an 8-byte RBA
RRN RIDFLD has a 4-byte relative record number

KEYLENGTH

Can be specified as a constant or label.
A constant must not exceed 32767.
A label must be 2 bytes and must not exceed 32767.
Keylengths greater than 128 are ignored.

The parameter is ignored for ESDS and RRDS.
If KEYLENGTH is specified, the value must be equal to the keylength defined for the file.

Errors
BAD PARM
BOTH FILE AND DATASET ARE SPECIFIED
BOTH INTO AND SET ARE SPECIFIED
BOTH LENGTH AND FLENGTH ARE SPECIFIED
BOTH RBA AND XRBA ARE SPECIFIED
BOTH RRN AND (X)RBA ARE SPECIFIED
FILE OR DATASET MUST BE SPECIFIED
INTO OR SET MUST BE SPECIFIED
INVALID FILE OR DATASET
RIDFLD IS MANDATORY

Conditions (RESP/RESP2)
DISABLED/50
ENDFILE/90
FILENOTFOUND/1
ILLOGIC/110
INVREQ/20
INVREQ/24
INVREQ/26
INVREQ/41
LENGERR/E1
NOTFND/80
NOTOPEN/60

Note: ENDFILE can occur when a READPREV attempts to read past
the beginning of the file.

d) ENDBR FILE()/DATASET() REQID()

DATASET is supported for legacy applications. It is noted that this parameter no longer appears in the Manuals.

REQID
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.
If omitted, zero is assumed.

Note: In real CICS, ENDBR cannot cause a file to open, but it will in Z390/CICS. The ENDBR command will be invalid, and may result in a transaction abend.

Errors
BAD PARM
BOTH FILE AND DATASET ARE SPECIFIED
FILE OR DATASET MUST BE SPECIFIED
INVALID FILE OR DATASET

Conditions (RESP/RESP2)
DISABLED/50
FILENOTFOUND/1
ILLOGIC/110
INVREQ/20
INVREQ/35
NOTOPEN/60

e) RESETBR FILE()/DATASET() RIDFLD() REQID() RBA/XRBA/RRN/GENERIC GTEQ/EQUAL KEYLENGTH()

REQID
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.
If omitted, zero is assumed.

RBA RIDFLD has a 4-byte RBA
XRBA RIDFLD has an 8-byte RBA
RRN RIDFLD has a 4-byte relative record number

KEYLENGTH
Can be specified as a constant or label.
A constant must not exceed 32767.
A label must be 2 bytes and must not exceed 32767.
Keylengths greater than 128 are ignored.

The parameter is ignored for ESDS and RRDS.
KEYLENGTH and GENERIC must be paired.

```
            If KEYLENGTH is zero by constant or label then parameters
are
            changed internally:
               GENERIC/EQUAL or GENERIC/GTEQ
                  Becomes KEYLENGTH(1) Key=X'00' GENERIC GTEQ

         GTEQ/EQUAL
            The parameter is ignored for ESDS and RRDS.

         GENERIC
            KEYLENGTH must be specified.
            The parameter is ignored for ESDS and RRDS.

         Note: In real CICS, RESETBR cannot cause a file to open, but it
               will in Z390/CICS. The RESETBR command will be invalid,
and
               may result in a transaction abend.

         Errors
            BAD PARM
            BOTH FILE AND DATASET ARE SPECIFIED
            BOTH GTEQ AND EQUAL ARE SPECIFIED
            BOTH RBA AND XRBA ARE SPECIFIED
            BOTH RRN AND (X)RBA ARE SPECIFIED
            FILE OR DATASET MUST BE SPECIFIED
            GENERIC CANNOT BE SPECIFIED WITH RRN OR (X)RBA
            GENERIC REQUIRES KEYLENGTH
            INVALID FILE OR DATASET
            KEYLENGTH REQUIRES GENERIC
            RIDFLD IS MANDATORY

         Conditions (RESP/RESP2)
            DISABLED/50
            FILENOTFOUND/1
            ILLOGIC/110
            INVREQ/20
            INVREQ/25
            INVREQ/36
            INVREQ/42
            NOTFND/80
            NOTOPEN/60

            Note: NOTFND cannot occur for an ESDS or RRDS
```

4) Storage Control (08)

a) 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
BOTH DATA AND DATAPOINTER ARE SPECIFIED
DATA OR DATAPOINTER MUST BE SPECIFIED

Conditions (RESP/RESP2)
INVREQ/1

b) 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
BOTH LENGTH AND FLENGTH ARE SPECIFIED
LENGTH OR FLENGTH MUST BE SPECIFIED
SET IS MANDATORY

5) Temporary Storage Control (0A)

a) DELETEQ TS QUEUE()/QNAME()

The parameters MAIN and AUXILIARY are accepted and discarded.

QUEUE may be specified as:
A quoted string which must not exceed 8 bytes.
A label which points to an 8-byte field.
A literal not exceeding 8 bytes.
Only label or literal may be used to specify a QUEUE with hex characters.

QNAME may be specified as:
A quoted string which must not exceed 16 bytes.
A label which points to a 16-byte field.
A literal not exceeding 16 bytes.
Only label or literal may be used to specify a QNAME with hex characters.

Errors
BAD PARM
BOTH QUEUE AND QNAME ARE SPECIFIED
DELETEQ TYPE NOT RECOGNIZED
INVALID QUEUE OR QNAME
QUEUE OR QNAME MUST BE SPECIFIED

Conditions (RESP/RESP2)
INVREQ/0
QIDERR/0

b) READQ TS QUEUE()/QNAME() INTO()/SET() LENGTH() NUMITEMS() ITEM()/NEXT

The parameters MAIN and AUXILIARY are accepted and discarded.

QUEUE may be specified as:
A quoted string which must not exceed 8 bytes.
A label which points to an 8-byte field.
A literal not exceeding 8 bytes.
Only label or literal may be used to specify a QUEUE with hex characters.

QNAME may be specified as:
A quoted string which must not exceed 16 bytes.

A label which points to a 16-byte field.
A literal not exceeding 16 bytes.
Only label or literal may be used to specify a QNAME with hex characters.

LENGTH
May 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 INTO is used. LENGTH is mandatory when SET is used.

ITEM
May be specified as ITEM(value) or ITEM(label)
label must point to a 2-byte hex value.

Errors
BAD PARM
BOTH INTO AND SET ARE SPECIFIED
BOTH ITEM AND NEXT ARE SPECIFIED
BOTH QUEUE AND QNAME ARE SPECIFIED
INTO OR SET MUST BE SPECIFIED
INVALID QUEUE OR QNAME
ITEM OR NEXT MUST BE SPECIFIED
QUEUE OR QNAME MUST BE SPECIFIED
READQ TYPE NOT RECOGNIZED
SET REQUIRES LENGTH

Conditions (RESP/RESP2)
INVREQ/0
LENGERR/0
ITEMERR/0
QIDERR/0

c) WRITEQ TS QUEUE()/QNAME() FROM() LENGTH() NUMITEMS() ITEM() REWRITE

The parameters MAIN and AUXILIARY are accepted and discarded.

QUEUE may be specified as:
A quoted string which must not exceed 8 bytes.
A label which points to an 8-byte field.
A literal not exceeding 8 bytes.
Only label or literal may be used to specify a QUEUE with

hex characters.

QNAME may be specified as:
A quoted string which must not exceed 16 bytes.
A label which points to a 16-byte field.
A literal not exceeding 16 bytes.
Only label or literal may be used to specify a QNAME with hex characters.

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

LENGTH
May 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 FROM is used. LENGTH is mandatory when FROM is an indirect reference.

ITEM
May be specified as ITEM(value) or ITEM(label)
label must point to a 2-byte hex value.

Errors
BAD PARM
BOTH QUEUE AND QNAME ARE SPECIFIED
FROM IS MANDATORY
IF NUMITEMS IS SPECIFIED, ITEM AND REWRITE ARE INVALID
INVALID QUEUE OR QNAME
ITEM AND REWRITE MUST BOTH BE SPECIFIED
LENGTH IS MANDATORY FOR INDIRECT FROM
LENGTH WITHOUT FROM
QUEUE OR QNAME MUST BE SPECIFIED
WRITEQ TYPE NOT RECOGNIZED

Conditions (RESP/RESP2)
INVREQ/0
LENGERR/0
ITEMERR/0
QIDERR/0

6) Program Control (0E)

a) 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

b) 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

c) 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

d) 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

e) 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
INVREQ/6

f) 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 See the section on IGNORE for this condition

g) 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

7) Interval Control (10 and 4A)

a) ASKTIME ABSTIME()

Errors
BAD PARM

b) DELAY INTERVAL()/TIME() FOR/UNTIL HOURS() MINUTES() SECONDS()

INTERVAL
Can be specified as INTERVAL(s) through to INTERVAL(hhmmss).
ie. INTERVAL(234) means wait for 2 minutes 34 seconds.

INTERVAL(label) is also permitted (extension).
label must point to a 6-byte character field with leading character zeros as needed.
eg. INTERVAL(MYTIME)
...

```
            MYTIME DC C'000234'

      TIME
         Can be specified as TIME(s) through to TIME(hhmmss).
         ie. TIME(234) means resume the task at 2 mins 34secs after
         midnight. Expiration time rules apply, see the IBM
Application
         Programming Guide.

         TIME(label) is also permitted (extension).
         label must point to a 6-byte character field with leading
         character zeros as needed.
         eg. TIME(MYTIME)
             ...
             MYTIME DC C'000234'

      FOR HOURS() MINUTES() SECONDS()
         FOR is an alternative to INTERVAL.
         HOURS/MINUTES/SECONDS must be numeric values.

      UNTIL HOURS() MINUTES() SECONDS()
         UNTIL is an alternative to TIME.
         HOURS/MINUTES/SECONDS must be numeric values.
         The result from the parameters is a time-of-day.
         eg. UNTIL SECONDS(10000) means resume the task at 02:46:40.
         Expiration time rules apply, see the IBM Application
         Programming Guide.

      If no parameters are specified, then DELAY INTERVAL(0) is
      assumed.

      Errors
         BAD PARM
         BOTH FOR AND UNTIL ARE SPECIFIED
         BOTH INTERVAL AND TIME ARE SPECIFIED
         FOR/UNTIL SPECIFIED, BUT NO TIME PARMS
         HOURS/MINUTES/SECONDS ARE INVALID WITH INTERVAL OR TIME
         HOURS/MINUTES/SECONDS ARE SPECIFIED WITHOUT FOR/UNTIL
         INTERVAL/TIME CANNOT BE SPECIFIED WITH FOR/UNTIL
         INTERVAL/TIME MUST BE 1 TO 6 BYTES
         REQID NOT SUPPORTED

      Conditions (RESP/RESP2)
         INVREQ/4
         INVREQ/5
```

INVREQ/6

c) FORMATTIME all parameters

Notes:
STRINGFORMAT is discarded as there is only one option.

DATESEP(label) and TIMESEP(label) are added as extensions.
Only the first byte is used.

DATESTRING returns the following 25-byte string.
The timezone (eg. GMT) is not returned.
eg. "Mon, 17 Dec 2007 10:20:30"

Errors
ABSTIME IS MANDATORY
BAD PARM

Conditions (RESP/RESP2)
INVREQ/1

8) BMS (18)
a) RECEIVE MAP(name) MAPSET(name)

TERMINAL and ASIS are accepted and discarded.
INTO, SET, FROM and LENGTH are not supported.

MAP is mandatory and must be a quoted string with a maximum of 7 characters.

MAPSET can be a quoted string, maximum 7 characters. or an address pointing to an 8-byte field containing no more than 7 characters.
If MAPSET is omitted, then the MAPname is used.

Errors
BAD PARM
INVALID MAP NAME
INVALID MAPSET NAME
LENGTH NOT SUPPORTED
MAP IS MANDATORY

Conditions (RESP/RESP2)
Note: EIBRESP2 is an extension for MAPFAIL, please see the Z390/CICS BMS Guide for more information.

Many of the conditions can arise through a mismatch of map and structure. Typically a map is re-assembled but the programs using it are not.

INVMPSZ/0

MAPFAIL/1 The map cannot be found in the mapset.
MAPFAIL/2 A short read key (CLEAR or PA) has been pressed or there are no modified fields.
MAPFAIL/3 Data has been received, but there are no named fields in the map.
MAPFAIL/4 An SBA has been located, but there is no field in the map that matches.
MAPFAIL/5 An SBA has been located, but it matches an unnamed field.
MAPFAIL/6 The data received for this field is longer than the DFHMDF LENGTH= parameter.
MAPFAIL/7 There has been a mismatch between the physical map and the DSECT.
MAPFAIL/8 Data to be processed by PICIN is over 31 digits or is not numeric after being PACKed.

b) SEND MAP(name) MAPSET(name)
CURSOR/CURSOR() DATAONLY/MAPONLY ERASE/ERASEAUP
ALARM FREEKB FRSET

TERMINAL and WAIT are accepted and discarded.
ACCUM, FROM and LENGTH are not supported.

MAP is mandatory and must be a quoted string with a maximum of 7 characters.

MAPSET can be a quoted string, maximum 7 characters. or an address pointing to an 8-byte field containing no more than 7 characters.
If MAPSET is omitted, then the MAPname is used.

Errors
BAD PARM
CONTROL IS NOT COMPATABLE WITH SEND MAP
CURSOR POSITION AND SYMBOLIC CURSOR SPECIFIED
DATAONLY AND MAPONLY SPECIFIED

```
         ERASE AND ERASEAUP SPECIFIED
         FROM NOT SUPPORTED
         INVALID MAP NAME
         INVALID MAPSET NAME
         LENGTH NOT SUPPORTED
         MAP IS MANDATORY

      Conditions (RESP/RESP2)
      Note: EIBRESP2 is an extension for MAPFAIL, please see the
            Z390/CICS BMS Guide for more information.

         INVMPSZ/0

         MAPFAIL/1  The map cannot be found in the mapset.
         MAPFAIL/8  Data to be processed by PICOUT is over 31 digits
or
                    is not numeric after being PACKed.

   c) SEND CONTROL CURSOR/CURSOR() ERASE/ERASEAUP
      ALARM FREEKB FRSET

      TERMINAL and WAIT are accepted and discarded.
      ACCUM and SET are not supported.

      CURSOR() refers to the 24x80 screen.
      CURSOR is not documented.
         SEND CONTROL CURSOR ERASEAUP means erase all input fields
         and don't move the cursor.

      Errors
         BAD PARM
         CURSOR POSITION AND SYMBOLIC CURSOR SPECIFIED
         ERASE AND ERASEAUP SPECIFIED

 9) Dump Control (1C)
    DUMP TRANSACTION DUMPCODE() COMPLETE
    FROM() LENGTH()/FLENGTH()
    SEGMENTLIST() LENGTHLIST() NUMSEGMENTS()

    TRANSACTION is mandatory.

    DUMPCODE is mandatory and can be a constant or label.
    label must point to a 4-byte field.
    No syntax checking is done.
```

```
    COMPLETE
       If there are no storage area parameters then COMPLETE is the
       default. Produces a SNAP dump ID=997,TEXT='DUMP dddd COMPLETE'

       If there are storage area parameters and COMPLETE is not
       specified, only the storage areas are dumped.

    FROM() LENGTH()/FLENGTH()
       Produces a SNAP dump ID=997,TEXT='DUMP dddd AREA'

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

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

    SEGMENTLIST/LENGTHLIST/NUMSEGMENTS
       Produces multiple SNAP dumps ID=997,TEXT='DUMP dddd SEGMENT
nnn'

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

    Errors
       BAD PARM
       BOTH LENGTH AND FLENGTH ARE SPECIFIED
       DUMPCODE IS GREATER THAN 4 BYTES
       DUMPCODE MUST BE SPECIFIED
       LENGTH OR FLENGTH REQUIRES FROM
       LENGTH OR FLENGTH MUST BE SPECIFIED
       SEGMENTLIST, LENGTHLIST AND NUMSEGMENTS MUST ALL BE SPECIFIED
          OR ALL ABSENT
       TRANSACTION MUST BE SPECIFIED

Appendicies
-----------

Keypress information
   Aid/Function  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
ERASE EOF	CTRL+F6
ERASE INPUT	CTRL+F7

Change Summary

June 27, 2008

- Added ERASE EOF and ERASE INPUT to the keypress table
- Added commands RECEIVE MAP, SEND MAP, SEND CONTROL and references to the new BMS document.

January 18, 2008

- Added READ/STARTBR/RESETBR parameters:
  - RRN, GTEQ, EQUAL, GENERIC, KEYLENGTH
- Added READNEXT/READPREV parameters RRN, KEYLENGTH
- Added ASKTIME, DELAY, FORMATTIME
- Added DUMP TRANSACTION

References

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

Trademarks

Author: Melvyn Maltz
Shipping Date: June 27, 2008
Z390 version: V1.4.02
Z390/CICS version: V5
→