
SPOOL
DEL.LST
05/04/82
15:38:00

SERIES-III PL/M-86 V2.0 COMPILATION OF MODULE DEL
 OBJECT MODULE PLACED IN :F1:DEL.OBJ
 COMPILER INVOKED BY: PLM86.86 :F1:DEL.P86 OPTIMIZE(3) XREF SET(F1) DEBUG

```

    $TITLE('ILNA TCL Host Process Delete Routine      11/04/81 12:10')
    $COMPACT DEBUG NOCOND
*** WARNING 10 IN 1 (LINE 2): RESPECIFIED PRIMARY CONTROL, IGNORED
    $SET(mipform)
                                /* George D Marshall 7-5117 */

    $IF f7
    $ELSE
    $INCLUDE (:F1:cpyrt.dcp)

=          /* Intel Corporation Proprietary Information.
=          This listing is supplied under the terms of a
=          license agreement with Intel Corporaton and
=          may not be copied nor disclosed except in
=          accordance with the terms of that agreement. */

    $ENDIF

1          del: DO;

    $IF f7
    $ELSE
    $INCLUDE (:F1:TCLGBL.INC)

=          /******
=          /** Global Literals **/
=          /******
=          /* TCL Global Literals
=          04/15/82 */
2  1  =  DECLARE
=          max$send$seg          LITERALLY  '07H', /* max no of back-to-back segs that one connection */
=                                     /* can send at a time */
=          tcl$header$len        LITERALLY  '20', /* bytes in tcl header */

=          /* ETHERNET-SPECIFIC VALUES */
=          dll$header$len        LITERALLY  '14', /* bytes in dll header */
=          min$pkt$len           LITERALLY  '46', /* minimum total pkt len - bytes */
=          max$seg$data$len$lit  LITERALLY  '1480', /* (1480) max no. of client bytes in seg */
=          tcl$protocol$code     LITERALLY  '5001H', /* DLLCONNECT user type field */
=          tcl$protocol$code$rev LITERALLY  '0150H', /* packet header user type field */

=          /* Misc values */
=          tcl$mip$port           LITERALLY  '4', /* mip port for IPSIN$MBX */
=          log$rb$mip$port       LITERALLY  '5', /* debugging: mip port for logging */
=          mip$echo$port         LITERALLY  '7', /* mip port of on-bd tcl echo server */

=          tcl$version$lit       LITERALLY  '101H', /* Version of this TCL for seg header */
=          def$net$id$lit        LITERALLY  '1', /* default Network ID: "this network" */
=          on$bd$tcl$echo$port   LITERALLY  '7', /* TCL port of on-board tcl echo server */
=          true                  LITERALLY  'OFFH',
=          false                 LITERALLY  '0',
=          forever               LITERALLY  'WHILE true',

```

```

=      Timeout$increase$state  LITERALLY '1', /* In this state the retransmission timeout
=                                     is rapidly increased */
=      Timeout$steady$state    LITERALLY '0'; /* In this state the timeout is
=                                     slowly decreased. This should not be
=                                     changed, it is the initial state since
=                                     a cdb is initialised to zero */

```

```
$ENDIF
```

```
/* Record of Compile-time options in the object code: */
```

```
/* prog version string commented out - too many bytes */
```

```
/*DECLARE prog_version_string (*) BYTE DATA('p_v_n',1,*/
```

```
/* 'DEL: ',*/
```

```
/*$IF log*/
```

```
/* 'log ',*/
```

```
/*$ENDIF*/
```

```
/*$IF dbg*/
```

```
/* 'dbg ',*/
```

```
/*$ENDIF*/
```

```
/*$IF mipform*/
```

```
/* 'mf ',*/
```

```
/*$ENDIF*/
```

```
/*$IF dbp*/
```

```
/* 'DBP ',*/
```

```
/*$ENDIF*/
```

```
/* 0);*/
```

```
/* end of compile-time record */
```

```
3 1
```

```
DECLARE
```

```
  lcid$vector(*) WORD EXTERNAL, /* list of allocated CIDs */
```

```
  cur$max$cdb$ BYTE EXTERNAL;
```

```
$IF f7
```

```
$ELSE
```

```
$SAVE NOLIST INCLUDE (:F1:KAOS.DCP)
```

```
$IF f7
```

```
$ELSE
```

```
$SAVE NOLIST INCLUDE (:F1:MIP.DCP)
```

```
$IF f7
```

```
$ELSE
```

```
$SAVE NOLIST INCLUDE (:F1:DLL.DCP)
```

```
$IF f7
```

```
$ELSE
```

```
$SAVE NOLIST INCLUDE (:F1:TCLMBX.INC)
```

```
$IF f7
```

```
$ELSE
```

```
$SAVE NOLIST INCLUDE (:F1:THACF.INC)
```

```
106 1
```

```
  setup$cdb: PROCEDURE(index, ptr$o) EXTERNAL;
```

```
/* in TSTART */
```

```
107 2
```

```
  DECLARE index BYTE,
```

```
          ptr$o WORD;
```

```

108 2      END setup$cdb;

109 1      abort$conn: PROCEDURE(cdb$index, cdb$ptr) EXTERNAL;      /* in IP */
110 2      DECLARE cdb$index  BYTE,
                cdb$ptr    POINTER;
111 2      END abort$conn;

112 1      delete$cdb: PROCEDURE(index, ptr, code) EXTERNAL;      /* in IP */
113 2      DECLARE (index, code)  BYTE,
                ptr POINTER;
114 2      END delete$cdb;

$IF f7
$ELSE
$SAVE NOLIST INCLUDE (:F1:TCLRBC.INC)

                /******
                /** mip$del$proc */
                /******

116 1      mip$del$proc: PROCEDURE(dev$id, process$id) PUBLIC;
                /* routine to delete all connections used */
                /* by a host process which has been deleted. */
                /* this is called by mip in response to a */
                /* mip msg from the host OS. Process IDs */
                /* here refer to the local OS' internal */
                /* name for its process, and may or may */
                /* not have any relationship to SCL process */
                /* names or IDs */

                /* Note: should minimize the stack depth
                incurred by this routine, as we are on MIP's
                stack, set up by ROM code, and therefore
                not very amenable to expansion. Took call
                to log$event out to save stack depth. */

117 2      DECLARE
                index      BYTE,
                dev$id     BYTE,      /* device ID of deleted process */
                process$id WORD,     /* process ID deleted process */
                mdp$cdb$p POINTER,
                mdp$c      BASED mdp$cdb$p
$IF f7
$ELSE
$SAVE NOLIST INCLUDE (:F1:TCLCDB.INC)
;

118 2      DO index = 0 TO cur$max$cdb$-1;      /* go thru every cdb, looking for proc id */
119 3          IF lcid$vector(index) <> 0 THEN
120 3              DO;

                /* first, grab the cdb lock to avoid */
                /* deleting a cdb while it's in use. */

121 4          CALL cq$wait$sem(.sched$lock);
122 4          CALL setup$cdb(index, .mdp$cdb$p); /* set up address of cdb */
123 4          IF (mdp$c.owner$device = dev$id) AND
                (mdp$c.owner$process$id = process$id) THEN
124 4              DO;      /* delete the cdb */
125 5                  CALL abort$conn(index, mdp$cdb$p); /* abort the conn, sending RST if needed */
126 5              END;

```

```
127 4          CALL cq$signal(.sched$lock); /* now give back the cdb lock */
128 4          CALL cq$schedule; /* give RP a chance to run in loopback case */
129 4          END;
130 3          END;

131 2          END mip$del$proc;

132 1          END del;
```

DEFN	ADDR	SIZE	NAME, ATTRIBUTES, AND REFERENCES	
109	0000H		ABORTCONN	PROCEDURE EXTERNAL(45) STACK=0000H 125
115			ABORTREQ	LITERALLY '8'
63	0000H	4	ALARM	POINTER IN PROC (CQSETALARM) PARAMETER 63
69	0000H	4	ALARM	POINTER IN PROC (CQCLEARALARM) PARAMETER 69
66	0000H	4	ALARM	POINTER IN PROC (CQCHECKALARM) PARAMETER 66
60	0000H	4	ALARM	POINTER IN PROC (CQCREATEALARM) PARAMETER 60
104	0000H	2	BUFMIPIBX	WORD EXTERNAL(42)
91	0000H	2	SUFO	WORD IN PROC (CQDLLRXRETBUF) PARAMETER 91
115			BUFTOOSHORT	LITERALLY '8'
110	0000H	1	CDBINDEX	BYTE IN PROC (ABORTCONN) PARAMETER 110
110	0000H	4	CDBPTR	POINTER IN PROC (ABORTCONN) PARAMETER 110
115			CLOSEREQ	LITERALLY '2'
113	0000H	1	CODE	BYTE IN PROC (DELETEDCDB) PARAMETER 113
65	0000H		CQCHECKALARM	PROCEDURE BYTE EXTERNAL(23) STACK=0000H
68	0000H		CQCLEARALARM	PROCEDURE EXTERNAL(24) STACK=0000H
59	0000H		CQCREATEALARM	PROCEDURE EXTERNAL(21) STACK=0000H
6	0000H		CQCREATelist	PROCEDURE EXTERNAL(3) STACK=0000H
29	0000H		CQCREATEMAILBOX	PROCEDURE EXTERNAL(11) STACK=0000H
14	0000H		CQCREATEPROCESS	PROCEDURE EXTERNAL(6) STACK=0000H
17	0000H		CQCREATESEMAPHORE	PROCEDURE EXTERNAL(7) STACK=0000H
38	0000H		CQCRECEIVE	PROCEDURE POINTER EXTERNAL(14) STACK=0000H
26	0000H		CQCWAIT	PROCEDURE BYTE EXTERNAL(10) STACK=0000H
101	0000H		CQDLLCONNECT	PROCEDURE BYTE EXTERNAL(36) STACK=0000H
95	0000H		CQDLLREAD	PROCEDURE WORD EXTERNAL(34) STACK=0000H
98	0000H		CQDLLREADC	PROCEDURE WORD EXTERNAL(35) STACK=0000H
90	0000H		CQDLLRXRETBUF	PROCEDURE EXTERNAL(32) STACK=0000H
93	0000H		CQDLLSTART	PROCEDURE EXTERNAL(33) STACK=0000H
86	0000H	2	CQDLLTXFREEMBX	WORD EXTERNAL(30)
87	0000H		CQDLLTXSEND	PROCEDURE EXTERNAL(31) STACK=0000H
11	0000H		CQHALTANDCATCHFIRE	PROCEDURE EXTERNAL(5) STACK=0000H
53	0000H		CQICRECEIVE	PROCEDURE POINTER EXTERNAL(19) STACK=0000H
47	0000H		CQICWAIT	PROCEDURE BYTE EXTERNAL(17) STACK=0000H
50	0000H		CQISEND	PROCEDURE EXTERNAL(18) STACK=0000H
44	0000H		CQISIGNAL	PROCEDURE EXTERNAL(16) STACK=0000H
74	0000H		CQMIPCONNECT	PROCEDURE BYTE EXTERNAL(26) STACK=0000H
80	0000H		CQMIPGETADDRESS	PROCEDURE POINTER EXTERNAL(28) STACK=0000H
83	0000H		CQMIPGETMIPFORM	PROCEDURE POINTER EXTERNAL(29) STACK=0000H
77	0000H		CQMIPREGISTER	PROCEDURE BYTE EXTERNAL(27) STACK=0000H
71	0000H		CQMIPSEND	PROCEDURE BYTE EXTERNAL(25) STACK=0000H
41	0000H		CQMRECEIVE	PROCEDURE POINTER EXTERNAL(15) STACK=0000H
56	0000H		CQREADCLOCK	PROCEDURE EXTERNAL(20) STACK=0000H
35	0000H		CQRECEIVE	PROCEDURE POINTER EXTERNAL(13) STACK=0000H
9	0000H		CQSCHEDULE	PROCEDURE EXTERNAL(4) STACK=0000H 128
32	0000H		CQSEND	PROCEDURE EXTERNAL(12) STACK=0000H
62	0000H		CQSETALARM	PROCEDURE EXTERNAL(22) STACK=0000H
20	0000H		CQSIGNAL	PROCEDURE EXTERNAL(8) STACK=0000H 127
4	0000H		CQSTART	PROCEDURE EXTERNAL(2) STACK=0000H
23	0000H		CQWAITSEM	PROCEDURE EXTERNAL(9) STACK=0000H 121
3	0000H	1	CURMAXCOBS	BYTE EXTERNAL(1) 118
57	0000H	2	DATARTNO	WORD IN PROC (CQREADCLOCK) PARAMETER 57
2			DEFNETIDLIT	LITERALLY '1'
115			DEFSTATUSREQ	LITERALLY '4'

	0000H		DEL.	PROCEDURE STACK=0000H					
63	0000H	2	DELAYH	WORD IN PROC (CQSETALARM) PARAMETER	63				
63	0000H	2	DELAYL	WORD IN PROC (CQSETALARM) PARAMETER	63				
112	0000H		DELETECDB.	PROCEDURE EXTERNAL(46) STACK=0000H					
117	0006H	1	DEVID.	BYTE IN PROC (MIPDELPROC) PARAMETER AUTOMATIC		117	123		
2			DLLHEADERLEN	LITERALLY '14'					
15	0000H	2	ENTRYO	WORD IN PROC (CQCREATEPROCESS) PARAMETER	15				
12	0000H	2	ERRORCODE.	WORD IN PROC (CQHALTANDCATCHFIRE) PARAMETER		12			
2			FALSE.	LITERALLY '0'					
2			FOREVER.	LITERALLY 'WHILE true'					
104	0000H	2	FREEIRBMBX	WORD EXTERNAL(40)					
104	0000H	2	FREELIRBMBX.	WORD EXTERNAL(41)					
105			HACFCHKACB	LITERALLY '438'					
105			HACFDLLCONN.	LITERALLY '401'					
105			HACFDLLREADHOST.	LITERALLY '402'					
105			HACFIPDEFIRB	LITERALLY '432'					
105			HACFMIPCONNECT	LITERALLY '404'					
105			HACFMIPREGISTER.	LITERALLY '434'					
105			HACFMIPSEND.	LITERALLY '416'					
105			HACFPUTCNT	LITERALLY '405'					
105			HACFRPCOMPLETE	LITERALLY '413'					
105			HACFRPDEFIRB	LITERALLY '433'					
105			HACFRPPUTDATAP	LITERALLY '412'					
105			HACFRPPUTNEWBLK.	LITERALLY '409'					
105			HACFRPPUTPTR	LITERALLY '411'					
105			HACFRPPUTSENDERBSSBACK	LITERALLY '410'					
105			HACFSENDERBLKLEN	LITERALLY '403'					
105			HACFTPGETBLKLEN.	LITERALLY '408'					
105			HACFTPGETCHK	LITERALLY '407'					
105			HACFTPIRBACB	LITERALLY '415'					
105			HACFTPIRBP	LITERALLY '414'					
105			HACFTPIRBTYPE.	LITERALLY '406'					
105			HACFTPTIMEOUT.	LITERALLY '435'					
105			HACFTQSEQ.	LITERALLY '417'					
105			HACFXQ	LITERALLY '436'					
105			HACFZQ	LITERALLY '437'					
115			ILLEGALREQ	LITERALLY '10'					
117	0004H	1	INDEX.	BYTE IN PROC (MIPDELPROC)	118*	118	119	122	125 130
113	0000H	1	INDEX.	BYTE IN PROC (DELETECDB) PARAMETER	113				
107	0000H	1	INDEX.	BYTE IN PROC (SETUPCDB) PARAMETER	107				
18	0000H	2	INIT	WORD IN PROC (CQCREATESEMAPHORE) PARAMETER	18				
115			INVALIDPOINTER	LITERALLY '20'					
115			INVALIDREQ	LITERALLY '2'					
104	0000H	2	IPINMBX.	WORD EXTERNAL(39)					
3	0000H		LCIDVECTOR	WORD ARRAY(0) EXTERNAL(0)	119				
7	0000H	4	LISTP.	POINTER IN PROC (CQCREATELIST) PARAMETER		7			
115			LOCABORT	LITERALLY '12'					
115			LOCTIMEOUT	LITERALLY '16'					
2			LOGRBMIPPORT	LITERALLY '5'					
51	0000H	2	MAILBOXO	WORD IN PROC (CQISEND) PARAMETER	51				
39	0000H	2	MAILBOXO	WORD IN PROC (CQCRECEIVE) PARAMETER	39				
36	0000H	2	MAILBOXO	WORD IN PROC (CQRECEIVE) PARAMETER	36				
33	0000H	2	MAILBOXO	WORD IN PROC (CQSEND) PARAMETER	33				
30	0000H	2	MAILBOXO	WORD IN PROC (CQCREATEMAILBOX) PARAMETER		30			
63	0000H	2	MAILBOXO	WORD IN PROC (CQSETALARM) PARAMETER	63				

54	0000H	2	MAILBOXO	WORD IN PROC (CQICRECEIVE) PARAMETER	54
2			MAXSEGDATALENLIT . .	LITERALLY '1480'	
2			MAXSENDSEG	LITERALLY '07H'	
102	0000H	2	MBXO	WORD IN PROC (CQDLLCONNECT) PARAMETER	102
75	0000H	2	MBXO	WORD IN PROC (CQMIPCONNECT) PARAMETER	75
117	0000H	124	MDPC	STRUCTURE BASED(MDPCDBP) IN PROC (MIPDELPROC)	
	0000H	1	STATE	BYTE	
	0001H	1	OWNERDEVICE	BYTE	123
	0002H	2	OWNERPROCESSID . . .	WORD	123
	0004H	2	LOCCID	WORD	
	0006H	2	LOCPORT	WORD	
	0008H	2	REMNET	WORD	
	000AH	6	REMHOST	WORD ARRAY(3)	
	0010H	2	REMPORT	WORD	
	0012H	2	PERSIST	WORD	
	0014H	2	ABORTTOHI	WORD	
	0016H	2	REMCID	WORD	
	0018H	4	RETRANTODW	DWORD	
	001CH	2	RESERVED	WORD	
	001EH	2	TIMEDSEQNO	WORD	
	0020H	4	SEGTRANSTIMEDW . . .	DWORD	
	0024H	4	CUMRETRANDW	DWORD	
	0028H	2	PERSISTCNT	WORD	
	002AH	4	CBTQHDR	POINTER	
	002EH	4	PCBQHDR	POINTER	
	0032H	4	DEFSTATUSP	POINTER	
	0036H	2	MYACKNO	WORD	
	0038H	2	SEEN	WORD	
	003AH	1	MYCREDIT	BYTE	
	003BH	1	CURBLKINDEX	BYTE	
	003CH	2	CBDATAINDEX	WORD	
	003EH	2	RCVBYTESCONSUMED . .	WORD	
	0040H	2	CURBLKLENLEFT	WORD	
	0042H	2	HISACKNO	WORD	
	0044H	2	NEXTTRANSMIT	WORD	
	0046H	1	CLOSEDREASON	BYTE	
	0047H	1	HISCREDIT	BYTE	
	0048H	2	HIGHESTSENT	WORD	
	004AH	1	CBTQBUFCNT	BYTE	
	004BH	1	PCBQBUFCNT	BYTE	
	004CH	2	PKTSREJ	WORD	
	004EH	2	PKTSRETRAN	WORD	
	0050H	2	NOCONFID	WORD	
	0052H	2	LASTNOCONFID	WORD	
	0054H	1	RETRANSMITSTATE . . .	BYTE	
	0055H	1	SENDFLAG	BYTE	
	0056H	2	PENDINGRCVDATA	WORD	
	0058H	2	RCVBUFREJCNT	WORD	
	005AH	2	AYTCOUNT	WORD	
	005CH	4	DATAALARMCB	WORD ARRAY(2)	
	0060H	1	DATAACBIRBTYPE	BYTE	
	0061H	1	DATAACBFLAG	BYTE	
	0062H	10	DATAACBREM	BYTE ARRAY(10)	
	006CH	4	CTLALARMCB	WORD ARRAY(2)	
	0070H	1	CTLACBIRBTYPE	BYTE	
	0071H	1	CTLACBFLAG	BYTE	

	0072H	10	CTLACBREM.	BYTE ARRAY(10)			
117	0000H	4	MDPCDBP.	POINTER IN PROC (MIPDELPROC)	122	123	125
51	0000H	4	MESSAGEP	POINTER IN PROC (CQISEND) PARAMETER			51
33	0000H	4	MESSAGEP	POINTER IN PROC (CQSEND) PARAMETER			33
2			MINPKTLEN.	LITERALLY '46'			
116	0000H	103	MIPDELPROC	PROCEDURE PUBLIC STACK=000EH			
2			MIPECHOPT.	LITERALLY '7'			
81	0000H	4	MIP_FORM	POINTER IN PROC (CQMIPGETADDRESS) PARAMETER			81
99	0000H	2	MODIFIER	WORD IN PROC (CQDLLREADC) PARAMETER			99
96	0000H	2	MODIFIER	WORD IN PROC (CQDLLREAD) PARAMETER			96
72	0000H	4	MSGP	POINTER IN PROC (CQMIPSEND) PARAMETER			72
115			NORESOURCESRESP.	LITERALLY '4'			
99	0000H	2	OBJECT	WORD IN PROC (CQDLLREADC) PARAMETER			99
96	0000H	2	OBJECT	WORD IN PROC (CQDLLREAD) PARAMETER			96
115			OKCLOSEDRESP	LITERALLY '9'			
115			OKEOMRESP.	LITERALLY '3'			
115			OKFINRESP.	LITERALLY '5'			
115			OKRESP	LITERALLY '1'			
2			ONBDTCLECHOPT.	LITERALLY '7'			
115			OPENAREQ	LITERALLY '0'			
115			OPENCONFLICT	LITERALLY '18'			
115			OPENPREQ	LITERALLY '1'			
15	0000H	2	PCBO	WORD IN PROC (CQCREATEPROCESS) PARAMETER			15
88	0000H	2	PKTO	WORD IN PROC (CQDLLTXSEND) PARAMETER			88
75	0000H	1	PORTID	BYTE IN PROC (CQMIPCONNECT) PARAMETER			75
115			POSTRBUFREQ.	LITERALLY '7'			
15	0000H	2	PRI.	WORD IN PROC (CQCREATEPROCESS) PARAMETER			15
78	0000H	2	PROCEDUREO	WORD IN PROC (CQMIPREGISTER) PARAMETER			78
117	0004H	2	PROCESSID.	WORD IN PROC (MIPDELPROC) PARAMETER AUTOMATIC			117 123
113	0000H	4	PTR.	POINTER IN PROC (DELETEDCB) PARAMETER			113
84	0000H	4	PTR.	POINTER IN PROC (CQMIPGETMIPFORM) PARAMETER			84
107	0000H	2	PTRO	WORD IN PROC (SETUPCDB) PARAMETER			107
115			REABORT	LITERALLY '14'			
115			REQMAX	LITERALLY '8'			
99	0000H	4	RETURNBUFF	POINTER IN PROC (CQDLLREADC) PARAMETER			99
96	0000H	4	RETURNBUFF	POINTER IN PROC (CQDLLREAD) PARAMETER			96
104	0000H	2	RPMBX.	WORD EXTERNAL(38)			
104	0000H	2	SCHEDLOCK.	WORD EXTERNAL(43)	121	127	
48	0000H	2	SEMAPHOREO	WORD IN PROC (CQICWAIT) PARAMETER			48
45	0000H	2	SEMAPHOREO	WORD IN PROC (CQISIGNAL) PARAMETER			45
27	0000H	2	SEMAPHOREO	WORD IN PROC (CQCWAIT) PARAMETER			27
24	0000H	2	SEMAPHOREO	WORD IN PROC (CQWAITSEM) PARAMETER			24
21	0000H	2	SEMAPHOREO	WORD IN PROC (CQSIGNAL) PARAMETER			21
18	0000H	2	SEMAPHOREO	WORD IN PROC (CQCREATESEMAPHORE) PARAMETER			18
115			SENDEOMREQ	LITERALLY '6'			
115			SENDREQ.	LITERALLY '5'			
106	0000H		SETUPCDB	PROCEDURE EXTERNAL(44) STACK=0000H			122
72	0000H	2	SOCKET	WORD IN PROC (CQMIPSEND) PARAMETER			72
15	0000H	2	STACKO	WORD IN PROC (CQCREATEPROCESS) PARAMETER			15
115			STATUSREQ.	LITERALLY '3'			
2			TCLHEADERLEN	LITERALLY '20'			
2			TCLMIPPORT	LITERALLY '4'			
2			TCLPROTOCOLCODE.	LITERALLY '5001H'			
2			TCLPROTOCOLCODEREV	LITERALLY '0150H'			
2			TCLVERSIONLIT.	LITERALLY '101H'			
2			TIMEOUTINCREASESTATE	LITERALLY '1'			

2		TIMEOUTSTEADYSTATE .	LITERALLY '0'	
104	0000H	2 TPMBX.	WORD EXTERNAL(37)	
2		TRUE	LITERALLY 'OFFH'	
102	0000H	2 TYPE	WORD IN PROC (CQDLLCONNECT) PARAMETER	102
115		UNKNOWNCIDRESP	LITERALLY '6'	
42	0000H	2 WCBO	WORD IN PROC (CQMRECEIVE) PARAMETER	42

MODULE INFORMATION:

CODE AREA SIZE = 0067H 103D
 CONSTANT AREA SIZE = 0000H 0D
 VARIABLE AREA SIZE = 0005H 5D
 MAXIMUM STACK SIZE = 000EH 14D
 490 LINES READ
 1 PROGRAM WARNING
 0 PROGRAM ERRORS

END OF PL/M-86 COMPILATION