# DIAGNOSTIC GUIDE

## HCC/MVS 3.0

**H**OST

**C**OMMUNICATION

**C**ONTROL

**4th edition**

Document number: 600225-A

# IMPRINT

**4th Edition**

This Manual has been written with utmost care. Textual or formal errors still cannot be excluded!

Protected trademarks are not marked as such in this Manual. The fact that these trademarks are not shown does not imply that the trade names are free for use.

All rights withheld, including those arising from applications for proprietary rights. The publisher retains all rights of disposition, such as copying or distribution.

This Manual is for internal use only.

Subject to changes without notice.

Extensions and changes to this Manual are based on PTF with status ZY00108, due to the fact that software changes were made at the same time.

Publisher: EMASS/GRAU Storage Systems GmbH,
Eschenstraße 3, D-89558 Böhmenkirch

(c) 1997 by EMASS/GRAU Storage Systems GmbH, Eschenstr. 3, D-89558 Böhmenkirch.

4th Edition in June 1997

# CONTENTS

**1 DIAGNOSTIC GUIDE FOR HCC/MVS 3.0 ..... 1-1**

**1.1 LITERATURE ..... 1-1**

**2 COMMUNICATION PROTOCOLS ..... 2-1**

**2.1 MESSAGE FORMAT ..... 2-1**
**2.2 HCC-AML COMMUNICATION ..... 2-2**
**2.3 HCC-HCC INTERCOMMUNICATION ..... 2-3**
2.3.1 COMMAND SCHEDULING ..... 2-3
2.3.2 MESSAGE SCHEDULING ..... 2-3
2.3.3 MVS MESSAGE SCHEDULING ..... 2-3
2.3.4 SIGNON SCHEDULING ..... 2-3
**2.4 HCC TOKEN RING COMMUNICATION ..... 2-4**
**2.5 HCC-HCC INTERCOMMUNICATION (LU 6.2) ..... 2-6**
2.5.1 COMMAND SCHEDULING ..... 2-6
2.5.2 MESSAGE SCHEDULING ..... 2-6
2.5.3 MVS MESSAGE SCHEDULING ..... 2-6
2.5.4 SIGN OF LIFE ..... 2-7
2.5.5 SIGN OF LIFE RETURN ..... 2-7
2.5.6 SETMAJOR ..... 2-7

**3 AML COMMUNICATION COMMANDS ..... 3-1**

**3.1 COMMAND FORMAT ..... 3-2**
**3.2 COMMAND OVERVIEW ..... 3-3**
**3.3 VOLSER STATUS ..... 3-4**
**3.4 COMMAND DESCRIPTION ..... 3-5**
3.4.1 ACOM (TEST CONNECTION) ..... 3-5
3.4.2 AOFF (SHUTDOWN CONNECTION) ..... 3-6
3.4.3 AUTO (AUTOMATIC MODE) ..... 3-7
3.4.4 BOFF (BARCODE READING BOFF) ..... 3-8
3.4.5 BON (BARCODE READING ON) ..... 3-9
3.4.6 CLU (CLOSE OPEN ENTRY DOOR) ..... 3-10
3.4.7 DL (DOWNLOAD ARCHIVE) ..... 3-11
3.4.8 DL (DOWNLOAD TO OPTICAL DISK) ..... 3-12
3.4.9 EJ (EJECT A TAPE) ..... 3-13
3.4.10 EJ (EJECT AN OPTICAL DISK) ..... 3-14
3.4.11 EJT (EJECT A TAPE COMPLETELY) ..... 3-15
3.4.12 EJT (EJECT OPTICAL DISK COMPLETELY) ..... 3-16
3.4.13 FLIP (TURN OPTICAL DISK OVER) ..... 3-17
3.4.14 INC (INVENTORY BY COORDINATE) ..... 3-18
3.4.15 INC (INVENTORY BY COORDINATE ON OPTICAL DISK) ..... 3-19
3.4.16 INV (INVENTORY BY VOLSER) ..... 3-20
3.4.17 INV (INVENTORY WITH VOLSER FOR OPTICAL DISK) ..... 3-21

3.4.18 KE (KEEP TAPE) ..... 3-22
3.4.19 KE (KEEP OPTICAL DISK FROM OAD) ..... 3-23
3.4.20 KEC (KEEP TO COORDINATE) ..... 3-24
3.4.21 KEC (KEEP OPTICAL DISK FROM OAD WITH COORDINATE) ..... 3-25
3.4.22 LJB (LOAD JUKEBOX) ..... 3-26
3.4.23 LMV (LOGICAL MOVE) ..... 3-27
3.4.24 MAN (MANUAL MODE) ..... 3-28
3.4.25 MO (MOUNT TAPE) ..... 3-29
3.4.26 MO - MOUNT OPTICAL DISK IN OAD ..... 3-30
3.4.27 MV (MOVE A TAPE) ..... 3-31
3.4.28 MV (MOVE OPTICAL DISK) ..... 3-32
3.4.29 ROPO (ROBOT INTO POSITION) ..... 3-33
3.4.30 ROSA (ROBOT START) ..... 3-34
3.4.31 ROSO (ROBOT STOP) ..... 3-35
3.4.32 SCH (SEARCH VOLSER) ..... 3-36
3.4.33 SCH (SEARCH VOLUME - OPTICAL DISK INSERT) ..... 3-37
3.4.34 UJB (UNLOAD JUKEBOX) ..... 3-38
3.4.35 ULC (UPLOAD ARCHIVE BY COORDINATE) ..... 3-39
3.4.36 ULC (UPLOAD WITH COORDINATE ON OPTICAL DISK) ..... 3-40
3.4.37 ULV (UPLOAD ARCHIVE BY VOLSER) ..... 3-41
3.4.38 ULC (UPLOAD WITH VOLSER ON OPTICAL DISK) ..... 3-42
3.4.39 UPC (UPDATE ARCHIVE STATUS BY COORDINATE) ..... 3-43
3.4.40 UPV (UPDATE ARCHIVE STATUS BY VOLSER) ..... 3-44
3.4.41 VI (VOLUME INSERT) ..... 3-45
3.4.42 VIC (VOLUME INSERT BY COORDINATE) ..... 3-46
3.4.43 VICC (VOLUME INSERT FROM COORDINATE TO COORDINATE) ..... 3-47
3.4.44 VICC (OPTICAL DISK INSERT) ..... 3-48

**4 3278 BUFFER ..... 4-1**

**4.1 BUFFER CONFIGURATION ..... 4-1**

**5 AML MESSAGES ..... 5-1**

**5.1 MESSAGE FORMAT ..... 5-1**
**5.2 MESSAGES ..... 5-2**

**6 MVS MESSAGES ..... 6-1**

**6.1 MOUNT MESSAGES ..... 6-1**
**6.2 KEEP MESSAGES ..... 6-1**
**6.3 EXCEPTION MESSAGES ..... 6-1**
**6.4 WTOR MESSAGES ..... 6-2**
**6.5 HSM WTOR MESSAGES ..... 6-2**
**6.6 NON-IBM MESSAGES ..... 6-2**

**7 SUMMARIES ..... 7-1**

**7.1 MODULE SUMMARY ..... 7-2**

7.2 MACRO SUMMARY ..... 7-5

8 RECORD FORMATS ..... 8-1

8.1 ARCHIVE VOLSER ..... 8-2
8.2 MESSAGE LOG ..... 8-5
8.3 EXCP READ/WRITE ..... 8-6

9 HCC MESSAGES ..... 9-1

9.1 MESSAGE FORMAT ..... 9-1
9.2 MESSAGES ..... 9-1

10 COMPILE AND LINK ..... 10-1

10.1 FOREGROUND ..... 10-1
10.2 BATCH COMPILE / LINK ..... 10-3

11 DISPLAY AND INITIALIZE HEADER ..... 11-1

11.1 JCL START PROCEDURE ..... 11-1
11.2 START COMMANDS ..... 11-2
11.3 START PARAMETER TCPU ..... 11-3

12 HCC/APPC DISPLAYS ..... 12-1

12.1 HCC-HCC EXAMPLES: ..... 12-1
12.2 VTAM TRACE CREATION EXAMPLE ..... 12-8
12.3 VTAM TRACE EXAMPLE ..... 12-10

13 TRACE ..... 13-1

13.1 FUNCTION: ..... 13-1
13.2 COMMAND SET: ..... 13-2
13.3 CALLING TRACE IN A PROGRAM: ..... 13-4

14 MESSAGE PROCESSING ..... 14-1

14.1 MESSAGES ..... 14-1
14.2 CHANGED MACROS ..... 14-1

15 MSG COMMAND ..... 15-1

16 WAIT COMMAND ..... 16-1

**17 REFRESH MODULE COMMAND ..... 17-1**

**INDEX ..... I-1**

# 1 DIAGNOSTIC GUIDE for HCC/MVS 3.0

## 1.1 LITERATURE

| Manual | Order number | Reference |
|---|---|---|
| General Information Manual | 600226-A | G |
| Installation/Customization Guide | 600227-A | G |
| System Reference Guide | 600233-A | G |
| Operators Guide | 600231-A | G |
| Command Reference | 600223-A | G |
| Conversion Notebook | 600224-A | G |
| Installation/Customization Reference | 600228-A | G |
| Messages and Codes | 600230-A | G |
| ISPF User Guide | 600229-A | G |
| Release Guide | 600232-A | G |
| Diagnostic Guide | 600225-A | I |
| Application Programming Macros | 600222-A | I |

G This Manual is part of the standard HCC documentation. Further copies of the respective Manual can be obtained from the publisher.

I This Manual is only intended for internal use within EMASS/GRAU Software GmbH and EMASS/GRAU Storage Systems. This Manual is not part of the standard HCC documentation and cannot be purchased as part of the additional documentation.

# 2 COMMUNICATION PROTOCOLS

## 2.1 MESSAGE FORMAT

The communication message in the overall HCC system has the following format:

```
¦------------------------- Overall length 80 characters -----------------------¦
¦                                                                               ¦
¦....+....1....+....2....+....3....+....4....+....5....+....6....+....7....+....8¦
¦<header ,                            Data                                     >¦
```

| | |
|---|---|
| <RRSSTTT | Message text depending on the record type TTT > |

| | |
|---|---|
| ¦ | Separation character (,) |
| TTT | Message type |
| RR | Receiver |
| SS | Sender |
| < | Message start character |
| > | Message termination character |

Participants (RR,SS) in the overall system can be:

RR = SS = M1 MAJOR HACC0
Hx HOST complex id
T1 TOKEN RING id
A1 AML System

TTT message types are:

TTT = A00 AML or
Hxx HOST intercommunication
Txx TOKEN RING

## 2.2 HCC-AML COMMUNICATION

Message type A00 is used for communication between HACC0 and the AML System, even when a TOKEN RING is installed.

Message type A00:

| <RRSSTTT | SQNR | COMD | A | RETC | S | R | DEV | VOLSER | ST | PARAM1 | PARAM2 | DD/HHMMSS | HI | SQNR> |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| <A1M1A00 | 2003 | MO | | | 1 | 1 | D02 | 004711 | | | | 25/061630 | H3 | 0123> |

| | | |
|---|---|---|
| Trailer: | HI | Owner's sequence number |
| | SQNR> | HACC0 id |

The record format is described in detail under Macro HACRMF (see Page 8-6).

The internal layout of this message is described in detail in the AML Communication command Section (see Page 3-1).

## 2.3 HCC-HCC INTERCOMMUNICATION

When a multi-HOST complex is installed, the single complexes can communicate with each other through the TOKEN RING.

### 2.3.1 COMMAND SCHEDULING

Initiating HCC command for a different complex:

Message type H00: CX CMD,Hx,HACC command

| | | |
|---|---|---|
| <RRSSH00 | HCC command > | |
| <H2H3H00 | CX SETMINOR (RR=Hx,SS=sender) > | Example |

### 2.3.2 MESSAGE SCHEDULING

Transferring any information message:

Message type H01: CX MVS,Hx,any message

| | | |
|---|---|---|
| <RRSSH01 | Any message e.g. for operating > | |
| <H3H4H01 | PLEASE INSERT YOUR EXTERNAL TAPES (RR=Hx,SS=sender) > | Example |

The protocol is created when the **CX** commands is used. Refer to HCC/MVS COMMAND REFERENCE.

### 2.3.3 MVS MESSAGE SCHEDULING

Informing a different complex about a MOUNT/KEEP message

Message type H02: internal use

| | | |
|---|---|---|
| <RRSSH02<br><RRSSH02 | cpux,jobname ,M,cuu,volser ><br>cpux,jobname ,K,cuu,volser > | |
| <H3H4H02 | PROD,V02R4401,M,4A1,004711 > | Example |

### 2.3.4 SIGNON SCHEDULING

SIGNON/SIGNOFF HOST complex participant

Message type H03: internal use (CX SOL,HX,HZ[,END])

| | | |
|---|---|---|
| <RRSSH03<br><RRSSH03 | Hz,VXRXMXX (signon) ><br>Hz,END (signoff) > | |
| <H3H4H03 | H4,V2R2M01 (signon for H4 to H3) > | Example |

## 2.4 HCC TOKEN RING COMMUNICATION

When a multi-HOST complex is installed, the single complexes can communicate with each other through the TOKEN RING.

Message type T00: SETMAJOR

| <RRSSTTT | SQNR | COMD | A | RETC | | | | | HI | | | DD/HHMMSS | HI | SQNR | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| <T1H3T00 | 1234 | SMJ | | | | | | | H3 | | | 19/071126 | H3 | 1234> | send |
| <H3T1T00 | 1234 | SMJ | P | | | | | | H3 | | | 19/071126 | H3 | 1234> | receive |

Message type T01: SWLINE

| <RRSSTTT | SQNR | COMD | A | RETC | | | | | | | | DD/HHMMSS | HI | SQNR | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| <T1H2T01 | 1008 | SWL | | | | | | | | | | 19/071126 | H2 | 1008> | send |
| <H2T1T01 | 1008 | SWL | P | | | | | | | | | 19/071126 | H3 | 1008> | receive |

Message type T02: ECHO

| <RRSSTTT | SQNR | COMD | A | RETC | | | | | | ARM1 | | DD/HHMMSS | HI | SQNR | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| <T1H1T02 | 0333 | ECO | | | | | | | | ECHO | | 19/071126 | H1 | 0333> | send |
| <H1T1T02 | 0333 | ECO | P | | | | | | | | | 19/071126 | H1 | 0333> | receive |

Message type T03: TOKEN message

| <RRSSTTT | Unexpected TOKEN RING message > | |
|---|---|---|
| <H3T1T03 | TOKnnni message > | receive |

Messages:

| | |
|---|---|
| TOK000I | Heartbeat with Archive computer (Standby) erroneous |
| TOK001I | Heartbeat with Archive computer (Standby) started |
| TOK002A | Could not reach monitoring computer |
| TOK003A | Standby for UNITxx is down |
| TOK004I | New M1-server not marked as active in Router table |
| TOK005I | Active Kx/Sx/Ux cannot be reached, will be regarded as defective |
| TOK006A | Computer defective, no other computer can be reached |
| TOK007I | DETECTED ERROR ON PRIMARY PATH;<br>STANDBY SWITCHED TO PRIMARY<br>(HCC COM SWITCH is activated on receipt of this message) |

## Message type T05: GETMAJOR

| <RRSSTTT | SQNR | COMD | A | RETC | | | | | HI | | | DD/HHMMSS | HI | SQNR | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| <T1H2T05 | 1010 | GMJ | | | | | | | | | | 03/120958 | H2 | 1010> | send |
| <H2T1T05 | 1010 | GMJ | P | | | | | | H3 | | | 03/120958 | H2 | 1010> | receive |

## Message type T06: GETLINE

| <RRSSTTT | SQNR | COMD | A | RETC | | | DEV | | | | | DD/HHMMSS | HI | SQNR | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| <T1H5T06 | 4331 | GLI | | | | | 566 | | | | | 17/120958 | H5 | 4331> | send |
| <H5T1T06 | 4331 | GLI | P | | | | 566 | | | STANDBY or PRIMARY | | 17/120958 | H5 | 4331> | receive |

## Message type T99: SHUTDOWN

| <RRSSTTT | SQNR | COMD | A | RETC | | | | | | PARAM1 | | DD/HHMMSS | HI | SQNR | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| <T1H2T99 | 4331 | SHT | | | | | | | | SHUTDOWN | | 17/120958 | H2 | 4331> | send |
| <H2T1T99 | 4331 | SHT | P | | | | | | | SHUTDOWN | | 17/120958 | H2 | 4331> | receive |

These protocols are generated when the **CX** commands are used, apart from message type T03 (refer to HCC/MVS COMMAND REFERENCE).

## 2.5 HCC-HCC INTERCOMMUNICATION (LU 6.2)

When a multi-HOST complex is installed, the single complexes can communicate with each other through LU 6.2.

### 2.5.1 COMMAND SCHEDULING

Initiating HCC command from complex H3 for complex H2:

Message type H00: CX CMD,H2,HACC command

| | | | |
|---|---|---|---|
| <RRSSH00 | HCC command | > | |
| <H2H3H00 | CX SETMINOR (RR=Hx,SS=sender) | > | Example |

### 2.5.2 MESSAGE SCHEDULING

Transferring any information message:

Message type H01: CX MVS,Hx, any message

| | | | |
|---|---|---|---|
| <RRSSH01 | Any message e.g. for operating | > | |
| <H3H4H01 | PLEASE INSERT YOUR EXTERNAL TAPES (RR=Hx,SS=sender) | > | Example |

These protocols are generated when the **CX** commands are used (refer to HCC/MVS COMMAND REFERENCE).

### 2.5.3 MVS MESSAGE SCHEDULING

Informing a different complex about a MOUNT/KEEP message

Message type H02: internal use

| | | | |
|---|---|---|---|
| <RRSSH02 | cpux,jobname ,M,cuu,volser | > | |
| <RRSSH02 | cpux,jobname ,K,cuu,volser | > | |
| <H3H4H02 | PROD,V02R4401,M,4A1,004711 | > | Example |

## 2.5.4 SIGN OF LIFE

Logical connection test

Message type H03: internal use

| | | | |
|---|---|---|---|
| <RRSSH03 | Hy,VxRxMxx,TT.MM.YY,Hz | > | |
| <H3H4H03 | H4,V2R3M01,14.05.92,H1 | > | Example |

Hy = as for SS (Sender)
Hz = HACC0 ID of the MAJOR complex

## 2.5.5 SIGN OF LIFE RETURN

Logical connection test

Message type H04: internal use

| | | | |
|---|---|---|---|
| <RRSSH04 | Hy,VxRxMxx,TT.MM.YY,Hz | > | |
| <H4H3H04 | H3,V2R3M01,14.05.92,H1 | > | Example |

Hy = as for SS (Sender)
Hz = HACC0 ID of the MAJOR complex

Reply from H3 to H03.

## 2.5.6 SETMAJOR

Naming a new MAJOR complex

Message type H05: internal use

| | | | |
|---|---|---|---|
| <RRSSH05 | Hy,VxRxMxx,TT.MM.YY,Hz | > | |
| <H3H4H05 | H4,V2R3M01,14.05.92,H4 | > | Example |

Hy = as for SS (Sender)
Hz = HACC0 ID of the new MAJOR complex

A **CX SETMAJOR** command triggers sending H05.

# 3 AML COMMUNICATION COMMANDS

The commands described in the following are used for communication between HCC and AML.

Two basic command groups exist:

1. Communication with the AML Archive PC (AR[1]/AMU[2])
2. Communication with the robots

[1] AR = Archive computer (only for ABBA/1 systems)
[2] AMU = Archive Management Unit (for AML/2, AML/E and AML/J systems)

## 3.1 COMMAND FORMAT

The individual commands are shown in the following syntax (a **MOUNT** command serves as example):

| SQNR | COMD | A | RETC | S | R | DEV | VOLSER | ST | PARAM1 | PARAM2 | DD/HHMMSS |
|---|---|---|---|---|---|---|---|---|---|---|---|
| ssss | MO | r | | s | s | sss | ssssss | | rrrrrrrr | | ss/ssssss |

| | |
|---|---|
| DD/HHMMSS | Time stamp |
| PARAM2 | Parameter 2 |
| PARAM1 | Parameter 1 |
| ST | Volser status |
| VOLSER | Volser |
| DEV | AML device |
| R | Robot Number |
| RETC | Return code |
| A | Acknowledgment |
| COMD | Command |
| SQNR | Sequence number |

s = SEND field required  r = RECEIVE field feedback

Fields are separated by commas. The information is 64 bytes long.

This format is embedded in the general communication protocol (refer to the COMMUNICATION PROTOCOLS Section):

Message type A00:

| header | Details of this part see Section AML COMMANDS | | | | | | | | | | | | trailer | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| <RRSSTTT | SQNR | COMD | A | RETC | S | R | DEV | VOLSER | ST | PARAM1 | PARAM2 | DD/HHMMSS | HI | SQNR> |

| | | |
|---|---|---|
| Trailer: | HI | Seq. Number of owner |
| | SQNR> | HACC0 id |

The record layout is described in detail in macro HACRMF (see Page 8-6).

## 3.2 COMMAND OVERVIEW

(shown with positive acknowledgment)

`ssss` = send field `rrrr` = receive field `srsr` = send and receive field

| SQNR | COMD | A | RETC | S | R | DEV | VOLSER | ST | Coordinates Home-Target PARAM1 | PARAM2 | DD/HHMMSS | Command group |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| ssss | ACOM | r | | s | | | rrrrrr | | rsrsrsrs | rsrsrsrs | ss/ssssss | COMMUNICATION |
| ssss | AOFF | r | | s | | | | | ssssssss | | ss/ssssss | CONTROL |
| ssss | ROSA | r | | s | s | | | rr | | | ss/ssssss | |
| ssss | ROSO | r | | s | s | | | | | | ss/ssssss | |
| ssss | ROPO | r | | s | s | | | | | ssssssss | ss/ssssss | |
| ssss | BON | r | | s | s | | | | | | ss/ssssss | ROBOT |
| ssss | BOFF | r | | s | s | | | | | | ss/ssssss | CONTROL |
| ssss | MAN | r | | s | s | | | | | | ss/ssssss | |
| ssss | AUTO | r | | s | s | | | | | | ss/ssssss | |
| ssss | MO | r | | s | s | sss | ssssss | | rrrrrrrr | rrrrrrrr | ss/ssssss | |
| ssss | KE | r | | s | s | sss | ssssss | | rrrrrrrr | rrrrrrrr | ss/ssssss | |
| ssss | KEC | r | | s | s | sss | rrrrrr | | rrrrrrrr | ssssssss | ss/ssssss | |
| ssss | MV | r | | s | s | sss | ssssss | | ssssssss | ssssssss | ss/ssssss | |
| ssss | EJ | r | | s | s | sss | ssssss | | rrrrrrrr | rrrrrrrr | ss/ssssss | |
| ssss | EJT | r | | s | s | sss | ssssss | | rrrrrrrr | rrrrrrrr | ss/ssssss | |
| ssss | VI | r | | s | s | sss | rrrrrr | | rrrrrrrr | rrrrrrrr | ss/ssssss | ROBOT |
| ssss | VIC | r | | s | s | sss | rrrrrr | | rsrsrsrs | rsrsrsrs | ss/ssssss | FUNCTIONS |
| ssss | VICC | r | | s | s | sss | ssssss | ss | ssssssss | ssssssss | ss/ssssss | |
| ssss | INC | r | | s | s | | rrrrrr | rr | ssssssss | | ss/ssssss | |
| ssss | INV | r | | s | s | | ssssss | rr | rrrrrrrr | | ss/ssssss | |
| ssss | SCH | r | | s | s | | rrrrrr | | rsrsrsrs | ssssssss | ss/ssssss | |
| ssss | CLU | r | | s | s | sss | | | | | ss/ssssss | |
| ssss | DL | r | | s | | | ssssss | ss | ssssssss | | ss/ssssss | |
| ssss | ULC | r | | s | r | | rrrrrr | rr | ssssssss | | ss/ssssss | ARCHIVE PC |
| ssss | ULV | r | | s | r | | ssssss | rr | rrrrrrrr | | ss/ssssss | COMMUNICATION |
| ssss | LJB | r | | s | s | sss | ssssss | | | | ss/ssssss | |
| ssss | UJB | r | | s | s | sss | ssssss | | rrrrrrrr | rrrrrrrr | ss/ssssss | |
| ssss | FLIP | r | | s | s | sss | ssssss | | | | ss/ssssss | |
| ssss | UPC | r | | s | | | | ss | ssssssss | | ss/ssssss | |
| ssss | UPV | r | | s | | | ssssss | ss | | | ss/ssssss | |
| ssss | LMV | r | | s | s | | rrrrrr | rr | ssssssss | ssssssss | ss/ssssss | |

## 3.3 VOLSER STATUS

A 2 byte field "lp" specifies the VOLSER status.

"l" is the logical status and "p" the physical status.

| | | |
|---|---|---|
| l=logical status: | C = | Cleaning tape |
| | M = | Data tape |
| | 0 = | Init status (no VOLSER assigned) |
| p=physical status: | B = | Tape in Archive |
| | E = | Tape ejected |
| | M = | Tape on tape unit |
| | L = | Init status (VOLSER="FREI00") |
| | U = | Undefined |

## 3.4 COMMAND DESCRIPTION

### 3.4.1 ACOM (test connection)

AML command: ACOM
HCC command: ACOM [s]
COMMON NAME: TEST CONNECTION
FUNCTION: Tests connection to a system and its robots.

| SQNR | COMD | A | RETC | S | R | DEV | VOLSER | ST | PARAM1 | PARAM2 | DD/HHMMSS | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| ssss | ACOM | r | | s | | | rrrrrr | | rrrrrrrr | rrrrrrrr | ss/ssssss | SYNTAX |
| 0123 | ACOM | | | 1 | | | | | HACC | V2R04M00 | 08/041251 | SEND |
| 0123 | ACOM | P | | 1 | | | V3.A01 | | .V01.A02 | .V01.A02 | 08/041251 | ACK POS |

**SEND PARAMETERS:**

PARAM1 contains Host software name: HCC
PARAM2 contains Host software release: VxRyyMzz
zz = modification level
Ryy = release
Vx = version

**RETURN POSITIVE:**

The connection to the system is ready.
VOLSER contains ARCHIVE-release-number
PARAM1 contains ROBOT-IC 1 release-number (when ROBOT 1 is active)
PARAM2 contains ROBOT-IC 2 release-number (when ROBOT 2 is active)

**RETURN NEGATIVE:**

No errors.

**COMMENTS:**

The ROBOT-IC software release can only be shown when the robot is active. This information is sent to the ARCHIVE PC after successful robot initialization and is deleted at robot deactivation (ARCHIVE PC and ROBOT-IC communication using **ROBS** commands).

When the ROBOT-IC release number is returned with ACOM the robot is active otherwise it is not ready.

## 3.4.2 AOFF (shutdown connection)

AML command: AOFF
HCC command: AOFF [s]
COMMON NAME: SHUTDOWN CONNECTION
FUNCTION: Archive PC deactivates all communication functions to HOST, towers and robots.

| SQNR | COMD | A | RETC | S | R | DEV | VOLSER | ST | PARAM1 | PARAM2 | DD/HHMMSS | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| ssss | AOFF | r | | s | | | | | SHUTDOWN | | ss/ssssss | SYNTAX |
| 0123 | AOFF | | | 1 | | | | | SHUTDOWN | | 08/041251 | SEND |
| 0123 | AOFF | P | | 1 | | | | | SHUTDOWN | | 08/041251 | ACK POS |

**RETURN POSITIVE:**

The Archive PC is completely deactivated. The Archive dataset is correctly closed.

A restart can only be effected in manual operation or after POWER ON.

**RETURN NEGATIVE:**

No error codes.

**COMMENTS:**

This function serves for REMOTE SHUTDOWN of the Archive PC without destroying the index of archive dataset as with REMOTE POWER OFF. This command must precede any REMOTE POWER OFF process.

### 3.4.3 AUTO (automatic mode)

AML command: AUTO
HCC command: AUTO [s,r]
COMMON NAME: AUTOMATIC MODE
FUNCTION: AUTO is the normal processing mode of a system and an associated robot. AUTO mode can be switched to manual mode with the MAN command.

| SQNR | COMD | A | RETC | S | R | DEV | VOLSER | ST | PARAM1 | PARAM2 | DD/HHMMSS | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| ssss | AUTO | r | | s | s | | | | | | ss/ssssss | SYNTAX |
| 0123 | AUTO | | | 1 | 2 | | | | | | 08/041251 | SEND |
| 0123 | AUTO | P | | 1 | 2 | | | | | | 08/041251 | ACK POS |

**RETURN POSITIVE:**

The specified robot in the specified system is in automatic processing mode.

**RETURN NEGATIVE:**

No errors.

**COMMENTS:**

AUTO is the default processing mode after HCC start.

### 3.4.4 BOFF (barcode reading boff)

AML command: BOFF
HCC command: BOFF [s,r]
COMMON NAME: BARCODE READING OFF
FUNCTION: Switches off barcode reading for a specific robot in a specific system.

| SQNR | COMD | A | RETC | S | R | DEV | VOLSER | ST | PARAM1 | PARAM2 | DD/HHMMSS | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| ssss | BOFF | r | | s | s | | | | | | ss/ssssss | SYNTAX |
| 0123 | BOFF | | | 1 | 2 | | | | | | 08/041251 | SEND |
| 0123 | BOFF | P | | 1 | 2 | | | | | | 08/041251 | ACK POS |

**RETURN POSITIVE:**

Barcode reading is bypassed. Tapes will be taken out of slots without verifying the barcode label. BOFF-status can be reset with BON.

**RETURN NEGATIVE:**

No errors.

### 3.4.5 BON (barcode reading on)

AML command: BON
HCC command: BON [s,r]
COMMON NAME: BARCODE READING ON
FUNCTION: Switches on barcode reading for a specific robot in a specific system.

| SQNR | COMD | A | RETC | S | R | DEV | VOLSER | ST | PARAM1 | PARAM2 | DD/HHMMSS | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| ssss | BON | r | | s | s | | | | | | ss/ssssss | SYNTAX |
| 0123 | BON | | | 1 | 2 | | | | | | 08/041251 | SEND |
| 0123 | BON | P | | 1 | 2 | | | | | | 08/041251 | ACK POS |

**RETURN POSITIVE:**

Barcode reading is active. The robot verifies the barcode label before taking a tape out of a slot. BON-status can be reset with BOFF.

**RETURN NEGATIVE:**

No errors.

## 3.4.6 CLU (close open entry door)

AML command: CLU
HCC command: CLU cuu
COMMON NAME: CLOSE OPEN ENTRY DOOR (3480)
FUNCTION: Tasks a specific robot of a specific system to close the open entry door of a 3480 tape device.

| SQNR | COMD | A | RETC | S | R | DEV | VOLSER | ST | PARAM1 | PARAM2 | DD/HHMMSS | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| ssss | CLU | r | | s | s | sss | | | | | ss/ssssss | SYNTAX |
| 0123 | CLU | | | 1 | 2 | D13 | | | | | 08/041251 | SEND |
| 0123 | CLU | P | | 1 | 2 | D13 | | | | | 08/041309 | ACK POS |
| 0123 | CLU | N | | 1 | 2 | D13 | | | | | 08/041306 | ACK NEG |

**RETURN POSITIVE:**

The open entry door of the specified tape device has been closed.

**RETURN NEGATIVE:**

Refer to ERROR CODES on Page 5-2.

## 3.4.7 DL (download archive)

AML command: DL
HCC command: DLC,DLV
COMMON NAME: DOWNLOAD ARCHIVE INFORMATIONS
FUNCTION: Downloads volser, status, coordinate from HCC to the AML Archive mirror of a specific system.

| SQNR | COMD | A | RETC | S | R | DEV | VOLSER | ST | PARAM1 | PARAM2 | DD/HHMMSS | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| ssss | DL | r | | s | | sss | ssssss | ss | ssssssss | | ss/ssssss | SYNTAX |
| 0123 | DL | | | 1 | | D13 | 004711 | ME | 02151510 | | 08/041251 | SEND |
| 0123 | DL | P | | 1 | | D13 | 004711 | ME | 02151510 | | 08/041309 | ACK POS |
| 0123 | DL | N | N401 | 1 | | D13 | 004711 | ME | 02151510 | | 08/041306 | ACK NEG |

**RETURN POSITIVE:**

Information on volser, status, coordinate is passed to the AML Archive mirror.

(Status: refer to VOLSER STATUS)

**RETURN NEGATIVE:**

Refer to ERROR CODES on Page 5-2.

## 3.4.8 DL (download to optical disk)

AML command: DL
HCC command:
COMMON NAME:
FUNCTION: Downloads volser and coordinate

| SQNR | COMD | A | RETC | S | R | DEV | VOLSER | ST | PARAM1 | PARAM2 | DD/HHMMSS | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| ssss | DL | r | | s | s | | ssssss | ss | sssssssss | | ss/ssssss | SYNTAX |
| 0123 | DL | | | 1 | 2 | | 04711A | Ox | 01010101 | | 08/041251 | SEND |
| 0123 | DL | P | | 1 | 2 | | 04711A | Ox | 01010101 | | 08/041309 | ACK POS |
| 0123 | DL | N | Nnnn | 1 | 2 | | 04711A | Ox | 01010101 | | 08/041300 | ACK NEG |

**SEND PARAMETERS**

CMD: DL
VOLSER: Volser / A or B side
STATUS: OB, OE, OJ, OO, OM
PARAM1: Coordinate of this volser

**RETURN POSITIVE:**

A: P
STATUS: Unchanged
VOLSER: Unchanged
PARAM1: Unchanged

**RETURN NEGATIVE:**

A: N

Additional ERROR CODES
none

## 3.4.9 EJ (eject a tape)

AML command: EJ
HCC command: EJ,EJDSN
COMMON NAME: EJECT A TAPE
FUNCTION: Tasks a specific robot in a specific system to move a volser from a source coordinate to a free slot in an eject area.

| SQNR | COMD | A | RETC | S | R | DEV | VOLSER | ST | PARAM1 | PARAM2 | DD/HHMMSS | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| ssss | EJ | r | | s | s | sss | ssssss | | rrrrrrrr | rrrrrrrr | ss/ssssss | SYNTAX |
| 0123 | EJ | | | 1 | 2 | E01 | 004711 | | | | 08/041251 | SEND |
| 0123 | EJ | P | | 1 | 2 | E01 | 004711 | | 03051109 | 00020423 | 08/041311 | ACK POS |
| 0123 | EJ | N | N503 | 1 | 2 | E01 | 004711 | | | | 08/041255 | ACK NEG |

**RETURN POSITIVE:**

The specified volser has been ejected into the specified eject device.
PARAM1: source coordinate of specified volser
PARAM2: target coordinate of the eject unit

**RETURN NEGATIVE:**

ERROR CODES (refer to Page 5-2):

N005: ROBOT not ready
N011: Coordinate assignment for wrong robot
N104: Cartridge lost
N105: Cartridge stuck in gripper
N301: Unknown volser
N302: Volser not in library
N304: Barcode label not readable
N305: No medium found in slot
N306: Wrong volser in source slot
N503: Eject device overrun

## 3.4.10 EJ (eject an optical disk)

AML command: EJ
HCC command:
COMMON NAME:
FUNCTION: Ejects a medium (2 volsers) from an eject coordinate to a free slot in an eject device.

| SQNR | COMD | A | RETC | S | R | DEV | VOLSER | ST | PARAM1 | PARAM2 | DD/HHMMSS | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| ssss | EJ | r | | s | s | sss | ssssss | | rrrrrrrr | rrrrrrrr | ss/ssssss | SYNTAX |
| 0123 | EJ | | | 1 | 2 | Exx | 04711A | | | | 08/041251 | SEND |
| 0123 | EJ | P | | 1 | 2 | Exx | 04711A | | 01010101 | 00020101 | 08/041309 | ACK POS |
| 0123 | EJ | N | Nnnn | 1 | 2 | Exx | 04711A | | | | 08/041300 | ACK NEG |

**SEND PARAMETERS**

CMD: EJ
DEV: Eject device; fixed for optical disks
VOLSER: Volser / A or B side
PARAM1: Source coordinate of the volser (optional)

**RETURN POSITIVE:**

A: P
VOLSER: Unchanged
PARAM1: Source coordinate
PARAM2: Target coordinate / I/E device

**RETURN NEGATIVE:**

A: N

Additional ERROR CODES
N404: Coordinate media mismatch
N501: Door open
N503: I/E full (without crash)

**COMMENT**

AMU Controls the eject, is responsible for messages that the I/E device is full (no crash).

HCC Passes the optical disk eject control to the AMU, only a fixed I/E device must be specified. Specification of EJVOL parameter in the HACCPARM for the optical disk volser ranges.

## 3.4.11 EJT (eject a tape completely)

AML command: EJT
HCC command: Internal
COMMON NAME: EJECT A TAPE AND ERASE VOLSER ENTRY
FUNCTION: Tasks a specific robot in a specific system to move a volser from a source coordinate to a position in an eject area. The volser entry is completely deleted from the AML archive mirror.

| SQNR | COMD | A | RETC | S | R | DEV | VOLSER | ST | PARAM1 | PARAM2 | DD/HHMMSS | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| ssss | EJT | r | | s | s | sss | ssssss | | rrrrrrrr | rrrrrrrr | ss/ssssss | SYNTAX |
| 0123 | EJT | | | 1 | 2 | E01 | 004711 | | | | 08/041251 | SEND |
| 0123 | EJT | P | | 1 | 2 | E01 | 004711 | | 03051109 | 00020423 | 08/041311 | ACK POS |
| 0123 | EJT | N | N011 | 1 | 2 | E01 | 004711 | | | | 08/041255 | ACK NEG |

**RETURN POSITIVE:**

The specified volser has been ejected into the specified eject device and the volser entry has been removed from the Archive mirror, for example, the volser is now unknown to the AML Archive mirror. (Reinsert refer to VICC).
PARAM1: source coordinate of specified volser
PARAM2: target coordinate of the eject device

**RETURN NEGATIVE:**

ERROR CODES (refer to Page 5-2):

N005: ROBOT not ready
N011: Coordinate assignment for wrong robot
N104: Cartridge lost
N105: Cartridge stuck in gripper
N301: Unknown volser
N302: Volser not in library
N304: Barcode label not readable
N305: No medium found in slot
N306: Wrong volser on source position
N503: Eject device overrun

## 3.4.12 EJT (eject optical disk completely)

AML command: EJT
HCC command:
COMMON NAME:
FUNCTION: Ejects a medium (2 volsers) from an eject coordinate to a free slot in an eject device, followed by slot release.

| SQNR | COMD | A | RETC | S | R | DEV | VOLSER | ST | PARAM1 | PARAM2 | DD/HHMMSS | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| ssss | EJT | r | | s | s | sss | ssssss | | rrrrrrrr | rrrrrrrr | ss/ssssss | SYNTAX |
| 0123 | EJT | | | 1 | 2 | Exx | 04711A | | | | 08/041251 | SEND |
| 0123 | EJT | P | | 1 | 2 | Exx | 04711A | | 01010101 | 00020101 | 08/041309 | ACK POS |
| 0123 | EJT | N | Nnnn | 1 | 2 | Exx | 04711A | | | | 08/041300 | ACK NEG |

**SEND PARAMETERS**

CMD: EJT
DEV: Eject Device; fixed for optical disks
VOLSER: Volser / A or B side
PARAM1: Source coordinate of the volser (optional)

**RETURN POSITIVE:**

A: P
VOLSER: Unchanged
PARAM1: Source coordinate
PARAM2: Target coordinate / I/E device

**RETURN NEGATIVE:**

A: N

Additional ERROR CODES
N404: Coordinate media mismatch
N501: Door open
N503: I/E full (without Crash)

**COMMENT**

AMU Controls the eject, is responsible for messages that the I/E device is full (no crash).

HCC Passes the optical disk eject control to the AMU, only a fixed I/E device must be specified. Specification of EJVOL parameter in the HACCPARM for the optical disk volser ranges.

## 3.4.13 FLIP (turn optical disk over)

AML command: FLIP
HCC command:
COMMON NAME:
FUNCTION: Turns over an optical disk (OD) in an Operator Accessible Drive (OAD).

| SQNR | COMD | A | RETC | S | R | DEV | VOLSER | ST | PARAM1 | PARAM2 | DD/HHMMSS | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| ssss | FLIP | r | | s | s | sss | ssssss | | | | | SYNTAX |
| 0123 | FLIP | | | 1 | 2 | D02 | 04711B | | | | | SEND |
| 0123 | FLIP | P | | 1 | 2 | D02 | 04711B | | | | | ACK POS |
| 0123 | FLIP | N | Nnnn | 1 | 2 | D02 | 04711B | | | | | ACK NEG |

**SEND PARAMETERS:**

CMD: FLIP
DEV: Drive
VOLSER: New volser
PARAM1: Drive coordinate (optional)

**RETURN POSITIVE:**

A: P

**RETURN NEGATIVE:**

A: N

Additional error codes
N209: Invalid device type

**COMMENT:**

AMU No volser check

## 3.4.14 INC (inventory by coordinate)

AML command: INC
HCC command: INC [s,r,],coor ..
COMMON NAME: INVENTORY BY COORDINATE
FUNCTION: Tasks a specific robot in a specific system to verify the contents of a coordinate.

| SQNR | COMD | A | RETC | S | R | DEV | VOLSER | ST | PARAM1 | PARAM2 | DD/HHMMSS | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| ssss | INC | r | | s | s | | rrrrrr | rr | ssssssss | | ss/ssssss | SYNTAX |
| 0123 | INC | | | 1 | 2 | | | | 03010101 | | 08/041251 | SEND |
| 0123 | INC | P | | 1 | 2 | | 004712 | MB | 03010101 | | 08/041319 | ACK POS |
| 0123 | INC | N | nnnn | 1 | 2 | | | | 03010101 | | 08/041254 | ACK NEG |

**RETURN POSITIVE:**

The specified coordinate has been located.
VOLSER: verified volser by barcode reading
ST: status of the volser according to Archive PC

**RETURN NEGATIVE:**

ERROR CODES (refer to Page 5-2):

N005: ROBOT not active
N007: Not executable, warnings may be preceded
N011: Invalid ROBOT-COORD assignment
N401: Unknown coordinate
N505: Inventory error

## 3.4.15 INC (inventory by coordinate on optical disk)

AML command: INC
HCC command:
COMMON NAME:
FUNCTION: Retrieves the status and volser inventory for a specific coordinate

| SQNR | COMD | A | RETC | S | R | DEV | VOLSER | ST | PARAM1 | PARAM2 | DD/HHMMSS | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| ssss | INC | r | | s | s | | rrrrrr | rr | rrrrrrrr | rrrrrrrr | ss/ssssss | SYNTAX |
| 0123 | INC | | | 1 | 2 | | | | 01010101 | | 08/041251 | SEND |
| 0123 | INC | P | | 1 | 2 | | 04711A | Ox | 01010101 | | 08/041309 | ACK POS |
| 0123 | INC | N | Nnnn | 1 | 2 | | 04711A | Ox | 01010101 | | 08/041300 | ACK NEG |

**SEND PARAMETERS**

CMD: INC
PARAM1: Home coordinate of the volser
VOLSER: Volser at this coordinate (optional)
STATUS: Status of this volser (optional)

**RETURN POSITIVE:**

A: P
STATUS: OB - on home coordinate
OE - ejected
OM - mounted on OAD
OO - other side
OJ - in jukebox
VOLSER: Volser detected by robot for this coordinate
PARAM1: Unchanged

**RETURN NEGATIVE:**

A: N

Additional ERROR CODES
N401: Unknown coordinate

## 3.4.16 INV (inventory by volser)

AML command: INV
HCC command: INV volser[,UPD]
COMMON NAME: INVENTORY BY VOLSER
FUNCTION: Tasks a specific robot in a specific system to verify status and coordinate of a volser.

| SQNR | COMD | A | RETC | S | R | DEV | VOLSER | ST | PARAM1 | PARAM2 | DD/HHMMSS | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| ssss | INV | r | | s | s | | ssssss | rr | rrrrrrrr | | ss/ssssss | SYNTAX |
| 0123 | INV | | | 1 | 2 | | 004712 | | | | 08/041251 | SEND |
| 0123 | INV | P | | 1 | 2 | | 004712 | MB | 03010101 | | 08/041313 | ACK POS |
| 0123 | INV | N | N005 | 1 | 2 | | 004712 | | | | 08/041253 | ACK NEG |

**RETURN POSITIVE:**

The coordinate of the specified volser has been located.
PARAM1: Coordinate of the requested volser
ST: Status of the volser (see COMMAND SYNTAX OVERVIEW)

**RETURN NEGATIVE**

ERROR CODES (see also Page 5-2):

N005: ROBOT not active
N011: Invalid ROBOT-VOLSER assignment
N301: Unknown volser

## 3.4.17 INV (inventory with volser for optical disk)

AML command: INV
HCC command:
COMMON NAME:
FUNCTION: Retrieves the status and volser inventory for a specific coordinate

| SQNR | COMD | A | RETC | S | R | DEV | VOLSER | ST | PARAM1 | PARAM2 | DD/HHMMSS | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| ssss | INV | r | | s | s | | ssssss | rr | rrrrrrrr | | ss/ssssss | SYNTAX |
| 0123 | INV | | | 1 | 2 | | 04711A | | | | 08/041251 | SEND |
| 0123 | INV | P | | 1 | 2 | | 04711A | Ox | 01010101 | | 08/041309 | ACK POS |
| 0123 | INV | N | Nnnn | 1 | 2 | | 04711A | | | | 08/041300 | ACK NEG |

**SEND PARAMETERS**

CMD: INV
VOLSER: Volser to be checked / A or B side

**RETURN POSITIVE:**

A: P
STATUS: OB - on home coordinate
OE - ejected
OM - mounted on OAD
OO - other side
OJ - in jukebox
VOLSER: Unchanged
PARAM1: Volser coordinate

**RETURN NEGATIVE:**

A: N

Additional ERROR CODES
N301: Unknown volser

**COMMENT:**

HCC No error message (inventory difference) for AMU status OO, OM, OJ

## 3.4.18 KE (keep tape)

AML command: KE
HCC command: Internal function
COMMON NAME: KEEP A TAPE
FUNCTION: Tasks a specific robot in a specific system to keep a volser out from a tape-device.

| SQNR | COMD | A | RETC | S | R | DEV | VOLSER | ST | PARAM1 | PARAM2 | DD/HHMMSS | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| ssss | KE | r | | s | s | sss | ssssss | | rrrrrrrr | rrrrrrrr | ss/ssssss | SYNTAX |
| 0123 | KE | | | 1 | 2 | D02 | | | | | 08/041251 | SEND |
| 0123 | KE | P | | 1 | 2 | D02 | 004711 | | 03051109 | 02051109 | 08/041311 | ACK POS |
| 0123 | KE | N | N203 | 1 | 2 | D02 | 004711 | | | | 08/041255 | ACK NEG |

**RETURN POSITIVE:**

The specified volser has been mounted on specified device.
PARAM1: Source coordinate of volser (device)
PARAM1: Target coordinate of volser

**RETURN NEGATIVE:**

ERROR CODES (see also Page 5-2):

N005: ROBOT not active
N011: Invalid ROBOT-VOLSER assignment
N104: Cartridge lost
N105: Cartridge stuck in gripper
N201: Unknown device
N203: Device not occupied (detected by ARCHIVE PC)
N205: Device not occupied (detected by ROBOT)
N206: Device entry door is closed

## 3.4.19 KE (keep optical disk from OAD)

AML command: KE
HCC command:
COMMON NAME:
FUNCTION: Unloads an optical disk from the OAD

| SQNR | COMD | A | RETC | S | R | DEV | VOLSER | ST | PARAM1 | PARAM2 | DD/HHMMSS | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| ssss | KE | r | | s | s | sss | ssssss | | rrrrrrrr | rrrrrrrr | ss/ssssss | SYNTAX |
| 0123 | KE | | | 1 | 2 | D02 | 04711A | | | | 08/041251 | SEND |
| 0123 | KE | P | | 1 | 2 | D02 | 04711A | | 00000201 | 01010101 | 08/041309 | ACK POS |
| 0123 | KE | N | Nnnn | 1 | 2 | D02 | 04711A | | | | 08/041300 | ACK NEG |

**SEND PARAMETERS**

CMD: KE
DEV: Drive (OAD)
VOLSER: Ejected volser, "top side" specified
PARAM1: Drive coordinate - OAD (optional)
PARAM2: Volser coordinate (optional)

**RETURN POSITIVE:**

A: P
VOLSER: Volser kept
PARAM1: Drive coordinate
PARAM2: Volser home coordinate

**RETURN NEGATIVE:**

A: N

Additional ERROR CODES
N209: Invalid unit type
N307: UJB/KE successful, but with unexpected volser

**COMMENT:**

AMU "Wrong side up" ODs will be detected and corrected during the next mount

## 3.4.20 KEC (keep to coordinate)

AML command: KEC
HCC command: Internal function
COMMON NAME: KEEP TO COORDINATE
FUNCTION: Tasks a specific robot of a specific system to move a volser from a tape device to a specified coordinate. The original home position of the cartridge is changed to the value of the target coordinate (PARAM2).

| SQNR | COMD | A | RETC | S | R | DEV | VOLSER | ST | PARAM1 | PARAM2 | DD/HHMMSS | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| ssss | KEC | r | | s | s | sss | rrrrrr | | rrrrrrrr | ssssssss | ss/ssssss | SYNTAX |
| 0123 | KEC | | | 1 | 1 | D03 | | | | 01120208 | 08/041251 | SEND |
| 0123 | KEC | P | | 1 | 1 | D03 | 012345 | | 00000301 | 01120208 | 08/041311 | ACK POS |
| 0123 | KEC | N | nnnn | 1 | 1 | D03 | | | | 01120208 | 08/041255 | ACK NEG |

**SEND PARAMETERS:**

PARAM2: Target coordinate for KEC

**RETURN POSITIVE:**

VOLSER: Volser previously mounted
PARAM1: Source coordinate
PARAM2: Unchanged

**RETURN NEGATIVE:**

Refer to ERROR CODES on Page 5-2.

For all error conditions, the Archive mirror of the ARCHIVE PC retains the original home coordinate of the cartridge to be kept.

**COMMENTS:**

KEC is a combination of LMV (logical move) and KE (normal keep).
KEC is used by HCC to maintain special devices such as HWS or PMA.

## 3.4.21 KEC (keep optical disk from OAD with coordinate)

AML command: KEC
HCC command:
COMMON NAME:
FUNCTION: Unloads an optical disk (OD) from the OAD

| SQNR | COMD | A | RETC | S | R | DEV | VOLSER | ST | PARAM1 | PARAM2 | DD/HHMMSS | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| ssss | KEC | r | | s | s | sss | rrrrrr | | rrrrrrrr | ssssssss | ss/ssssss | SYNTAX |
| 0123 | KEC | | | 1 | 2 | D02 | | | | 01010101 | 08/041251 | SEND |
| 0123 | KEC | P | | 1 | 2 | D02 | 04711A | | 00000201 | 01010101 | 08/041309 | ACK POS |
| 0123 | KEC | N | Nnnn | 1 | 2 | D02 | 04711A | | | | 08/041300 | ACK NEG |

**SEND PARAMETERS**

CMD: KEC
DEV: Drive (OAD)
VOLSER: Ejected volser, "top side" specified (optional)
PARAM1: Drive coordinate (optional)
PARAM2: Volser home coordinate

**RETURN POSITIVE:**

A: P
VOLSER: Volser kept
PARAM1: Drive coordinate
PARAM2: Unchanged

**RETURN NEGATIVE:**

A: N

Additional ERROR CODES
N209: Invalid unit type
N307: UJB/KE successful, but with unexpected volser

**COMMENT:**

AMU "Wrong side up" ODs will be detected and corrected during the next mount

## 3.4.22 LJB (LOAD jukebox)

AML command: LJB
HCC command:
COMMON NAME:
FUNCTION: Loads an optical disk in the Input/Output-Station (I/O-S).

| SQNR | COMD | A | RETC | S | R | DEV | VOLSER | ST | PARAM1 | PARAM2 | DD/HHMMSS | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| ssss | LJB | r | | s | s | sss | ssssss | | | | | SYNTAX |
| 0123 | LJB | | | 1 | 2 | D01 | 04711A | | | | | SEND |
| 0123 | LJB | P | | 1 | 2 | D01 | 04711A | | | | | ACK POS |
| 0123 | LJB | N | Nnnn | 1 | 2 | D01 | 04711A | | | | | ACK NEG |

**SEND PARAMETERS:**

CMD: LJB
DEV: Drive (I/O-S)
VOLSER: Volser requested
PARAM1: Volser coordinate (optional)
PARAM2: Drive coordinate (optional)

**RETURN POSITIVE:**

A: P

**RETURN NEGATIVE:**

A: N

Additional error codes
N209: Invalid unit type

**COMMENT:**

AMU No check for free slots in the jukebox, no Archive on jukebox contents

## 3.4.23 LMV (logical move)

AML command: LMV
HCC command: Internal function
COMMON NAME: LOGICAL MOVE
FUNCTION: Pseudo move only affecting the ARCHIVE PC: a volser and associated status is "moved" from a source to a target coordinate in the specified system. LMV is a move without robot access.

| SQNR | COMD | A | RETC | S | R | DEV | VOLSER | ST | PARAM1 | PARAM2 | DD/HHMMSS | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| ssss | LMV | r | | s | | | rrrrrr | rr | sssssssss | ssssssss | ss/ssssss | SYNTAX |
| 0123 | LMV | | | 1 | | | | | 02121314 | 00110203 | 08/041251 | SEND |
| 0123 | LMV | P | | 1 | | | 004711 | MB | 02121315 | 00110203 | 08/041252 | ACK POS |
| 0123 | LMV | N | nnnn | 1 | | | | | 02121314 | 00110203 | 08/041252 | ACK NEG |

**SEND PARAMETERS:**

PARAM1: Coordinate of source position
PARAM2: Coordinate of target position

**RETURN POSITIVE:**

VOLSER: Volser that has been logically moved
STATUS: Status of VOLSER

**RETURN NEGATIVE:**

Refer to ERROR CODES on Page 5-2.

**COMMENTS:**

LMV (logical move) operates the same as MV (physical move), however no errors occur when no cartridge is found at the source coordinate.

## 3.4.24 MAN (manual mode)

AML command: MAN
HCC command: MAN [s,r]
COMMON NAME: MANUAL MODE
FUNCTION: Disables the specific robot of a specific system logically, no messages are sent to the robot-IC. The ARCHIVE PC simulates all robot messages with a positive acknowledgment. MAN is reset by AUTO.

| SQNR | COMD | A | RETC | S | R | DEV | VOLSER | ST | PARAM1 | PARAM2 | DD/HHMMSS | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| ssss | MAN | r | | s | s | | | | | | ss/ssssss | SYNTAX |
| 0123 | MAN | | | 1 | 2 | | | | | | 08/041251 | SEND |
| 0123 | MAN | P | | 1 | 2 | | | | | | 08/041251 | ACK POS |

**RETURN POSITIVE:**

Manual mode is now active. The specified robot is inactive, the related messages to this robot are immediately answered as if the robot had processed the command. Data in the Archive mirror is updated on both levels: HCC and ARCHIVE PC.

**RETURN NEGATIVE:**

No errors.

## 3.4.25 MO (mount tape)

AML command: MO
HCC command: Internal function
COMMON NAME: MOUNT A TAPE
FUNCTION: Tasks a specific robot in a specific system to mount a volser into a tape device

| SQNR | COMD | A | RETC | S | R | DEV | VOLSER | ST | PARAM1 | PARAM2 | DD/HHMMSS | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| ssss | MO | r | | s | s | sss | ssssss | | rrrrrrrr | rrrrrrrr | ss/ssssss | SYNTAX |
| 0123 | MO | | | 1 | 2 | D02 | 004711 | | | | 08/041251 | SEND |
| 0123 | MO | P | | 1 | 2 | D02 | 004711 | | 02051109 | 00000201 | 08/041309 | ACK POS |
| 0123 | MO | N | N304 | 1 | 2 | D02 | 004711 | | | | 08/041300 | ACK NEG |

**RETURN POSITIVE:**

The specified volser has been mounted into specified device.
PARAM1: Source coordinate of volser (tape library)
PARAM2: Target coordinate of volser (device)

**RETURN NEGATIVE:**

ERROR CODES (refer to Page 5-2):

N005: ROBOT not ready
N011: Coordinate assignment for wrong robot
N104: Cartridge lost
N105: Cartridge stuck in gripper
N201: Unknown device
N202: Device occupied (detected by ARCHIVE PC)
N204: Device occupied (detected by ROBOT)
N206: Device entry door is locked
N207: Device entry door not lockable
N301: Unknown volser
N302: Volser ejected
N304: Barcode label not readable
N305: No volume label detected
N306: Wrong medium in slot
N401: Unknown coordinate
N402: No cartridge on coordinate

## 3.4.26 MO - mount optical disk in OAD

AML command: MO
HCC command:
COMMON NAME:
FUNCTION: Loads an optical disk into the OAD

| SQNR | COMD | A | RETC | S | R | DEV | VOLSER | ST | PARAM1 | PARAM2 | DD/HHMMSS | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| ssss | MO | r | | s | s | sss | ssssss | | rrrrrrrr | rrrrrrrr | ss/ssssss | SYNTAX |
| 0123 | MO | | | 1 | 2 | D02 | 004711 | | | | 08/041251 | SEND |
| 0123 | MO | P | | 1 | 2 | D02 | 004711 | | 01010101 | 00000201 | 08/041309 | ACK POS |
| 0123 | MO | N | Nnnn | 1 | 2 | D02 | 004711 | | | | 08/041300 | ACK NEG |

**SEND PARAMETERS:**

CMD: MO
DEV: Drive (OAD)
VOLSER: Volser requested

PARAM1: Volser home coordinate (optional)
PARAM2: Drive coordinate - OAD (optional)

**RETURN POSITIVE:**

A: P
PARAM1: Volser home coordinate
PARAM2: Drive coordinate - OAD

**RETURN NEGATIVE:**

A: N

Additional ERRORCODES
N209: Invalid unit type

## 3.4.27 MV (move a tape)

AML command: MV
HCC command: MOVEH, MOVET
COMMON NAME: MOVE A TAPE
FUNCTION: Tasks a specific robot in a specific system to move a volser from a source coordinate to a target coordinate.

| SQNR | COMD | A | RETC | S | R | DEV | VOLSER | ST | PARAM1 | PARAM2 | DD/HHMMSS | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| ssss | MV | r | | s | s | | ssssss | | ssssssss | rrrrrrrr | ss/ssssss | SYNTAX |
| 0123 | MV | | | 1 | 2 | | | | 02051109 | 00000201 | 08/041251 | SEND |
| 0123 | MV | P | | 1 | 2 | | 004711 | | 02051109 | 00000201 | 08/041308 | ACK POS |
| 0123 | MV | N | N306 | 1 | 2 | | 012345 | | 02051109 | 00000201 | 08/041253 | ACK NEG |

**SEND PARAMETERS:**

PARAM1: Source coordinate of volser
PARAM2: Target coordinate of volser

**RETURN POSITIVE:**

The specified volser has been moved from the source to the target position independent of source device type or target device type. The VOLSER of the moved cartridge is returned.

**RETURN NEGATIVE:**

VOLSER of wrongly located cartridge is returned.

ERROR CODES (refer to Page 5-2):

N005: ROBOT not active
N011: Invalid ROBOT-VOLSER assignment
N104: Cartridge lost
N105: Cartridge stuck in gripper
N302: Volser ejected
N304: Barcode label not readable
N305: No volume label detected
N306: Wrong volser on source position
N401: Unknown coordinate
N402: No cartridge in source slot
N403: Target slot occupied

## 3.4.28 MV (move optical disk)

AML command: MV
HCC command:
COMMON NAME:
FUNCTION: Moves a medium (2 volsers) from an eject coordinate to a target coordinate

| SQNR | COMD | A | RETC | S | R | DEV | VOLSER | ST | PARAM1 | PARAM2 | DD/HHMMSS | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| ssss | MV | r | | s | s | | ssssss | | sssssssss | sssssssss | ss/ssssss | SYNTAX |
| 0123 | MV | | | 1 | 2 | | | | 01010101 | 01010103 | 08/041251 | SEND OR |
| 0123 | MV | | | 1 | 2 | | 04711A | | | 01010103 | 08/041251 | SEND OR |
| 0123 | MV | | | 1 | 2 | | 04711A | | 01010101 | 01010103 | 08/041251 | SEND |
| 0123 | MV | P | | 1 | 2 | | 04711A | | 01010101 | 00020101 | 08/041309 | ACK POS |
| 0123 | MV | N | Nnnn | 1 | 2 | | 04711A | | 01010101 | 01010101 | 08/041300 | ACK NEG |

**SEND PARAMETERS**

CMD: MV
VOLSER: Volser / A or B side (required/optional)
PARAM1: Volser source coordinate (required/optional)
PARAM2: Volser target coordinate (required), valid for the side involved

**RETURN POSITIVE:**

A: P
VOLSER: Volser / A or B side that has been moved
PARAM1: Unchanged
PARAM2: Unchanged

**RETURN NEGATIVE:**

A: N

Additional ERROR CODES
N404: Coordinate media mismatch

**COMMENT**

AMU AMU performs a barcode check when the parameters VOLSER, PARAM1, PARAM2 are specified.

## 3.4.29 ROPO (robot into position)

AML command: ROPO
HCC command: ROPO p[s,r]
COMMON NAME: ROBOT INTO POSITION
FUNCTION: Moves a specific robot of a specific system into a position defined by PARAM2.

| SQNR | COMD | A | RETC | S | R | DEV | VOLSER | ST | PARAM1 | PARAM2 | DD/HHMMSS | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| ssss | ROPO | r | | s | s | | | | | ssssssss | ss/ssssss | SYNTAX |
| 0123 | ROPO | | | 1 | 2 | | | | | 00090501 | 08/041251 | SEND |
| 0123 | ROPO | P | | 1 | 2 | | | | | 00090501 | 08/041259 | ACK POS |
| 0123 | ROPO | N | nnnn | 1 | 2 | | | | | 00090501 | 08/041252 | ACK NEG |

**SEND PARAMETERS:**

PARAM2: `00090501` = back to end of track
`00090601` = middle position
`00090701` = forward to start of track
`ttssxxyy` = any valid position in library
`(0009xx01` = reserved internal maintenance positions (xx=01-99))

**RETURN POSITIVE:**

The specified robot in the specified system is now positioned as requested.

**RETURN NEGATIVE:**

ERROR CODES refer to Page 5-2.

## 3.4.30 ROSA (robot start)

AML command: ROSA
HCC command: ROSA [s,r]
COMMON NAME: ROBOT START
FUNCTION: Starts a specific robot in a specific system.

| SQNR | COMD | A | RETC | S | R | DEV | VOLSER | ST | PARAM1 | PARAM2 | DD/HHMMSS | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| ssss | ROSA | r | | s | s | | | r | | | ss/ssssss | SYNTAX |
| 0123 | ROSA | | | 1 | 2 | | | | | | 08/041251 | SEND |
| 0123 | ROSA | P | | 1 | 2 | | | ON | | | 08/041303 | ACK POS |
| MS00 | ROBS | M | | 1 | 2 | | | ON | | | 08/041304 | |
| 0123 | ROSA | N | N007 | 1 | 2 | | | | | | 08/041254 | ACK NEG |
| MS00 | ROBS | M | | 1 | 2 | | | OF | | | 08/041255 | |

**RETURN POSITIVE:**

The specified robot in the specified system is ready for operation. The acknowledgment returns after just a few seconds when the robot is already active, and after several minutes when the robot is not active but can be started. A ROBS message is issued before this positive acknowledgment because the robot status has changed.

On return, the status field contains the barcode operation mode of the robot:
ON barcode reading on
OF barcode reading off

**RETURN NEGATIVE**

The specified robot in the specified system is not ready for operation.

ERROR-CODES: (Refer to Page 5-2)

N007: Not executable due to unknown problems, refer to preceding warning messages
N011: Assignment error, invalid robot number specified

## 3.4.31 ROSO (robot stop)

AML command: ROSO
HCC command: ROSO [s,r]
COMMON NAME: ROBOT STOP
FUNCTION: Stops a specific robot in a specific system. After the currently active command has completed, the robot returns to its home-position and shuts down with power off.

| SQNR | COMD | A | RETC | S | R | DEV | VOLSER | ST | PARAM1 | PARAM2 | DD/HHMMSS | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| ssss | ROSO | r | | s | s | | | | | | ss/ssssss | SYNTAX |
| 0123 | ROSO | | | 1 | 2 | | | | | | 08/041251 | SEND |
| 0123 | ROSO | P | | 1 | 2 | | | | | | 08/041255 | ACK POS |
| 0123 | ROSO | N | | 1 | 2 | | | | | | 08/041253 | ACK NEG |

**RETURN POSITIVE:**

The specified robot in the specified system is not accessible. A ROBS message will follow immediately after positive acknowledgment to indicate the robot status change. The robot can only be restarted with a manual power on.

**RETURN NEGATIVE:**

ERROR CODES (refer to Page 5-2):

N005: Robot not active, command not applicable
N007: Not executable due to unknown problems, refer to preceding warning messages
N011: Assignment error, invalid robot number specified

## 3.4.32 SCH (search volser)

AML command: SCH
HCC command: VI
COMMON NAME: SEARCH VOLSER
FUNCTION: Tasks a specific robot in a specific system to search for the next volser in a specified coordinate range. When a volser has been detected, the VOLSER is returned and SCH terminates.

| SQNR | COMD | A | RETC | S | R | DEV | VOLSER | ST | PARAM1 | PARAM2 | DD/HHMMSS | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| ssss | SCH | r | | s | s | | rrrrrr | | rsrsrsrs | ssssssss | ss/ssssss | SYNTAX |
| 0123 | SCH | | | 1 | 2 | | | | 00010101 | 00010325 | 08/041251 | SEND |
| 0123 | SCH | P | | 1 | 2 | | 004712 | | 00010202 | 00010325 | 08/041320 | ACK POS |
| 0123 | SCH | E | | 1 | 2 | | | | 00010203 | 00010325 | 08/041316 | ACK EOF |
| 0123 | SCH | N | nnnn | 1 | 2 | | | | 00010107 | 00010325 | 08/041317 | ACK NEG |

**SEND PARAMETERS:**

PARAM1: SEARCH start coordinate
PARAM2: SEARCH end coordinate

**RETURN POSITIVE:**

VOLSER: Volser verified by barcode reading
PARAM1: Coordinate of the volser found
PARAM2: Remains unchanged

**RETURN AT "EOD" CONDITION:**

"END OF DEVICE" is returned when no volumes are found in the specified range: acknowledgment =E.

**RETURN NEGATIVE:**

Refer to ERROR CODES on Page 5-2.

## 3.4.33 SCH (search volume - optical disk insert)

AML command: SCH
HCC command:
COMMON NAME:
FUNCTION: Searches for the next volser within a specific coordinate range. A found volser is reported and the command terminates.

| SQNR | COMD | A | RETC | S | R | DEV | VOLSER | ST | PARAM1 | PARAM2 | DD/HHMMSS | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| ssss | SCH | r | | s | s | sss | ssssss | | ssssssss | ssssssss | ss/ssssss | SYNTAX |
| 0123 | SCH | | | 1 | 2 | I01 | | | 00010101 | 00010120 | 08/041251 | SEND |
| 0123 | SCH | P | | 1 | 2 | I01 | 04711A | | 00010105 | 00010120 | 08/041309 | ACK POS |
| 0123 | SCH | E | | 1 | 2 | I01 | | | 00010105 | 00010120 | 08/041309 | ACK END |
| 0123 | SCH | N | Nnnn | 1 | 2 | I01 | | | 00010105 | 00020101 | 08/041300 | ACK NEG |

**SEND PARAMETERS**

CMD: SCH
PARAM1: Search start coordinate
PARAM2: Search end coordinate
DEV: Insert device for the optical disk (optional)

**RETURN POSITIVE:**

A: P
VOLSER: Volser verified by barcode reading on the A or B side
PARAM1: Coordinate of the volser found
PARAM2: Unchanged

**RETURN AT „END“ CONDITION**

A: E
PARAM1: Coordinate of the volser last found or start coordinate
PARAM2: Unchanged

**RETURN NEGATIVE:**

A: N

Additional ERROR CODES
None

## 3.4.34 UJB (unload jukebox)

AML command: UJB
HCC command:
COMMON NAME:
FUNCTION: Unloads an optical disk in the Input/Output-Station (I/O-S).

| SQNR | COMD | A | RETC | S | R | DEV | VOLSER | ST | PARAM1 | PARAM2 | DD/HHMMSS | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| ssss | UJB | r | | s | s | sss | ssssss | | rrrrrrrr | rrrrrrrr | | SYNTAX |
| 0123 | UJB | | | 1 | 2 | D01 | 04711A | | | | | SEND |
| 0123 | UJB | P | | 1 | 2 | D01 | 04711A | | 00000201 | 01010101 | | ACK POS |
| 0123 | UJB | N | Nnnn | 1 | 2 | D01 | 04711A | | | | | ACK NEG |

**SEND PARAMETERS:**

CMD: UJB
DEV: Drive (I/O-S)
VOLSER: Ejected volser, "top side" specified
PARAM1: Drive coordinate (optional)
PARAM2: Volser coordinate (optional)

**RETURN POSITIVE:**

A: P
PARAM1 Drive coordinate
PARAM2 Volser coordinate

**RETURN NEGATIVE:**

A: N
VOLSER Volser detected by robot

Additional error codes
N307: UJB/KE successful, but unexpected volser

**COMMENT:**

AMU "Wrong side up" ODs will be detected and corrected during the next mount

HCC A dismount message with volser ?????? can occur when initialization is aborted; volser=blank is sent instead, followed by a move to the Problem box.

## 3.4.35 ULC (upload archive by coordinate)

AML command: ULC
HCC command: ULC, ACC
COMMON NAME: UPLOAD ARCHIVE INFORMATIONS BY COORDINATE
FUNCTION: Uploads volser and status of a coordinate from a specific AML system to the HCC Archive mirror.

| SQNR | COMD | A | RETC | S | R | DEV | VOLSER | ST | PARAM1 | PARAM2 | DD/HHMMSS | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| ssss | ULC | r | | s | r | | Rrrrrr | rr | ssssssss | | ss/ssssss | SYNTAX |
| 0123 | ULC | | | 1 | | | | | 02151510 | | 08/041251 | SEND |
| 0123 | ULC | P | | 1 | 2 | | 001000 | CB | 02151510 | | 08/041252 | ACK POS |
| 0123 | ULC | N | | 1 | | | | | 02151510 | | 08/041252 | ACK NEG |

**RETURN POSITIVE:**

Volser, status and the associated robot number are returned.
Status: refer to VOLSER STATUS on Page 3-4.

**RETURN NEGATIVE:**

ERROR CODES refer to Page 5-2.

**COMMENT:**

This function serves for HCC archive recovery or to compare both archives.

## 3.4.36 ULC (upload with coordinate on optical disk)

AML command: ULC
HCC command:
COMMON NAME:
FUNCTION: Uploads the volser and status for a specific coordinate

| SQNR | COMD | A | RETC | S | R | DEV | VOLSER | ST | PARAM1 | PARAM2 | DD/HHMMSS | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| ssss | ULC | r | | s | s | | rrrrrr | rr | ssssssss | | ss/ssssss | SYNTAX |
| 0123 | ULC | | | 1 | 2 | | | | 01010101 | | 08/041251 | SEND |
| 0123 | ULC | P | | 1 | 2 | | 04711A | Ox | 01010101 | | 08/041309 | ACK POS |
| 0123 | ULC | N | Nnnn | 1 | 2 | | | | 01010101 | | 08/041300 | ACK NEG |

**SEND PARAMETERS**

CMD: ULC
PARAM1: Volser home coordinate
VOLSER: Volser at this coordinate
STATUS: Status of this volser

**RETURN POSITIVE:**

A: P
STATUS: OB - on home coordinate
OE - ejected
OM - mounted on OAD
OO - other side
OJ - in jukebox
VOLSER: Volser / A or B side from the AMU Archive
PARAM1: Unchanged

**RETURN NEGATIVE:**

A: N

Additional ERROR CODES
None

## 3.4.37 ULV (upload archive by volser)

AML command: ULV
HCC command: ULV, ACV
COMMON NAME: UPLOAD ARCHIVE INFORMATION BY VOLSER
FUNCTION: Uploads coordinate and status of a volser from a specific AML system to the HCC Archive mirror.

| SQNR | COMD | A | RETC | S | R | DEV | VOLSER | ST | PARAM1 | PARAM2 | DD/HHMMSS | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| ssss | ULV | r | | s | r | | ssssss | | | | ss/ssssss | SYNTAX |
| 0123 | ULV | | | 1 | | | 004711 | | | | 08/041251 | SEND |
| 0123 | ULV | P | | 1 | 2 | | 004711 | MB | 02151510 | | 08/041251 | ACK POS |
| 0123 | ULV | N | N301 | 1 | | | 004711 | | | | 08/041251 | ACK NEG |

**RETURN POSITIVE:**

The coordinate, status and associated robot number are returned.
Status: refer to VOLSER STATUS on Page 3-4.

**RETURN NEGATIVE:**

ERROR CODES refer to Page 5-2.

**COMMENT:**

This function serves for HCC archive recovery and to compare the two archives.

## 3.4.38 ULC (upload with volser on optical disk)

AML command: ULV
HCC command:
COMMON NAME:
FUNCTION: Uploads the volser and status for a specific volser

| SQNR | COMD | A | RETC | S | R | DEV | VOLSER | ST | PARAM1 | PARAM2 | DD/HHMMSS | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| ssss | ULC | r | | s | s | | rrrrrr | rr | rrrrrrrr | | ss/ssssss | SYNTAX |
| 0123 | ULC | | | 1 | 2 | | 04711A | | | | 08/041251 | SEND |
| 0123 | ULC | P | | 1 | 2 | | 04711A | Ox | 01010101 | | 08/041309 | ACK POS |
| 0123 | ULC | N | Nnnn | 1 | 2 | | 04711A | | | | 08/041300 | ACK NEG |

**SEND PARAMETERS**

CMD: ULV
VOLSER: Volser
STATUS: OB, OE, OJ, OO, OM
PARAM1: Volser coordinate

**RETURN POSITIVE:**

A: P
STATUS: OB - on home coordinate
OE - ejected
OM - mounted on OAD
OO - other side
OJ - in jukebox
VOLSER: Unchanged
PARAM1: Coordinate of this volser

**RETURN NEGATIVE:**

A: N

Additional ERROR CODES
None

## 3.4.39 UPC (update archive status by coordinate)

AML command: UPC
HCC command: UPC
COMMON NAME: UPDATE ARCHIVE STATUS BY COORDINATE
FUNCTION: Updates the status of a coordinate in a specific AML Archive mirror and HCC Archive mirror.

| SQNR | COMD | A | RETC | S | R | DEV | VOLSER | ST | PARAM1 | PARAM2 | DD/HHMMSS | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| ssss | UPC | r | | s | | | | ss | ssssssss | | ss/ssssss | SYNTAX |
| 0123 | UPC | | | 1 | | | | MB | 02151510 | | 08/041251 | SEND |
| 0123 | UPC | P | | 1 | | | | MB | 02151510 | | 08/041251 | ACK POS |
| 0123 | UPC | N | N401 | 1 | | | | MB | 02151510 | | 08/041251 | ACK NEG |

**RETURN POSITIVE:**

The status of the specified coordinate is updated in the AML Archive mirror and HCC Archive mirror.
Status: refer to VOLSER STATUS on Page 3-4.

**RETURN NEGATIVE:**

ERROR CODES refer to Page 5-2.

## 3.4.40 UPV (update archive status by volser)

AML command: UPV
HCC command: UPV
COMMON NAME: UPDATE ARCHIVE STATUS BY VOLSER
FUNCTION: Updates the status of a volser in a specific AML Archive mirror and HCC Archive mirror.

| SQNR | COMD | A | RETC | S | R | DEV | VOLSER | ST | PARAM1 | PARAM2 | DD/HHMMSS | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| ssss | UPV | r | | s | | | ssssss | ss | | | ss/ssssss | SYNTAX |
| 0123 | UPV | | | 1 | | | CL0001 | CE | | | 08/041251 | SEND |
| 0123 | UPV | P | | 1 | | | CL0001 | CE | | | 08/041251 | ACK POS |
| 0123 | UPV | N | N301 | 1 | | | CL0001 | CE | | | 08/041251 | ACK NEG |

**RETURN POSITIVE:**

The status of the specified volser is updated in the AML Archive mirror and HCC Archive mirror.
Status: refer to VOLSER STATUS on Page 3-4.

**RETURN NEGATIVE:**

ERROR CODES refer to Page 5-2.

## 3.4.41 VI (volume insert)

AML command: VI
HCC command: VI
COMMON NAME: VOLUME INSERT
FUNCTION: Tasks a specific robot in a specific system to insert one volser from an insert device to its home position.

| SQNR | COMD | A | RETC | S | R | DEV | VOLSER | ST | PARAM1 | PARAM2 | DD/HHMMSS | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| ssss | VI | r | | s | s | sss | rrrrrr | | rrrrrrrr | rrrrrrrr | ss/ssssss | SYNTAX |
| 0123 | VI | | | 1 | 2 | I01 | | | | | 08/041251 | SEND |
| 0123 | VI | P | | 1 | 2 | I01 | 004712 | | 00010312 | 03111213 | 08/041312 | ACK POS |
| 0123 | VI | E | | 1 | 2 | I01 | | | | | 08/041300 | ACK EOF |
| 0123 | VI | N | N104 | 1 | 2 | I01 | | | | | 08/041252 | ACK NEG |

**RETURN POSITIVE:**

A cartridge has been inserted from the ROBOT related insert device to the home position.
PARAM1: Insert device coordinate
PARAM2: Target coordinate of the ejected volser
VOLSER: Volser verified by barcode reading

**RETURN AT "EOD" CONDITION:**

"E" is returned without any other information when the insert device is empty.

**RETURN NEGATIVE:**

ERROR CODES (refer to Page 5-2):

N005: Robot not active
N011: Invalid ROBOT-VOLSER assignment
N104: Cartridge lost
N105: Cartridge stuck in gripper
N301: Unknown volser
N304: Barcode label not readable
N305: No volume label detected

## 3.4.42 VIC (volume insert by coordinate)

AML command: VIC
HCC command: Internal function
COMMON NAME: VOLUME INSERT BY COORDINATE
FUNCTION: Tasks a specific robot of a specific system to insert cartridges from a determined coordinate range of an insert device.

| SQNR | COMD | A | RETC | S | R | DEV | VOLSER | ST | PARAM1 | PARAM2 | DD/HHMMSS | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| ssss | VIC | r | | s | s | | rrrrrr | ss | rsrsrsrs | rsrsrsrs | ss/ssssss | SYNTAX |
| 0123 | VIC | | | 1 | 2 | | | CE | 00010102 | 00010102 | 08/041251 | SEND |
| 0123 | VIC | P | | 1 | 2 | | 004711 | CE | 00010102 | 02311207 | 08/041319 | ACK POS |
| 0123 | VIC | N | nnnn | 1 | 2 | | | CE | 00010102 | 00010102 | 08/041319 | ACK NEG |

**SEND PARAMETERS:**

PARAM1: Start coordinate for VIC
PARAM2: End coordinate for VIC

**RETURN POSITIVE:**

VOLSER: Volser detected
PARAM1: Volser source coordinate
PARAM2: Volser target coordinate after VIC

**RETURN NEGATIVE:**

Refer to ERROR CODES on Page 5-2.

**COMMENTS:**

The **VIC** command can be used to store a cartridge to any slot, for example when a preceding **DL** command modified the target coordinate. HCC uses the **VIC** command for direct moves from the VI-area into FPMA.

## 3.4.43 VICC (volume insert from coordinate to coordinate)

AML command: VICC
HCC command: Internal function
COMMON NAME: VOLUME INSERT FROM COORDINATE TO COORDINATE
FUNCTION: Tasks a specific robot of a specific system to move a cartridge from a determined coordinate of an insert device to the specified target coordinate.

| SQNR | COMD | A | RETC | S | R | DEV | VOLSER | ST | PARAM1 | PARAM2 | DD/HHMMSS | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| ssss | VICC | r | | s | s | | ssssss | ss | ssssssss | ssssssss | ss/ssssss | SYNTAX |
| 0123 | VICC | | | 1 | 2 | | 004711 | MB | 00010102 | 02311207 | 08/041251 | SEND |
| 0123 | VICC | P | | 1 | 2 | | 004711 | | 00010102 | 02311207 | 08/041319 | ACK POS |
| 0123 | VICC | N | N306 | 1 | 2 | | 004712 | | 00010102 | 02311207 | 08/041319 | ACK NEG |

**SEND PARAMETERS:**

VOLSER: Volser detected by the **SCH** command
PARAM1: Coordinate determined by the **SCH** command
PARAM2: Any target coordinate

**RETURN POSITIVE:**

The volser has been inserted to the target coordinate as well as to the AML Archive mirror as a new cartridge in the library. (Refer to EJT for ejecting).

**RETURN NEGATIVE:**

Refer to ERROR CODES on Page 5-2.

**COMMENTS:**

HCC uses the **VICC** command to move a cartridge into the DYNAMIC area. The volser of the cartridge is defined as a "virtual volser" in HCC's Archive mirror but is unknown to the AML Archive mirror. After completion of the VICC instruction, the volser is also inserted into the AML archive mirror, for example, the cartridge has now become a normal library volser.

The processing sequence for "virtual volsers" is:

1. SCH: returns VOLSER and source COORDINATE
2. VICC: moves VOLSER from source to target COORDINATE (free slot of DYN)
3. normal MOUNT/KEEP processing
4. EJT: removes the cartridge completely from the AML library

## 3.4.44 VICC (optical disk insert)

AML command: VICC
HCC command:
COMMON NAME:
FUNCTION: Moves a medium (2 volsers) from a source coordinate (insert device) to a target coordinate.

| SQNR | COMD | A | RETC | S | R | DEV | VOLSER | ST | PARAM1 | PARAM2 | DD/HHMMSS | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| ssss | VICC | r | | s | s | | ssssss | | sssssssss | sssssssss | ss/ssssss | SYNTAX |
| 0123 | VICC | | | 1 | 2 | | 04711A | | 00010105 | 01010101 | 08/041251 | SEND |
| 0123 | VICC | P | | 1 | 2 | | 04711A | | 00010105 | 01010101 | 08/041319 | ACK POS |
| 0123 | VICC | N | Nnnn | 1 | 2 | | | | 00010101 | 01010101 | 08/041319 | ACK NEG |

**SEND PARAMETERS:**

CMD: VICC
VOLSER: Volser to be inserted / A or B side
PARAM1: Coordinate from Search
PARAM2: Home coordinate of the volser
DEV: Insert device for the optical disk (optional)

**RETURN POSITIVE:**

A: P
VOLSER: Unchanged
PARAM1: Unchanged
PARAM2: Unchanged

**RETURN NEGATIVE:**

A: N

**COMMENTS:**

AMU Insert area must not be defined as dynamic

HCC Only one **VICC** is sent per medium (A side or B side)

# 4 3278 BUFFER

## 4.1 BUFFER CONFIGURATION

The data buffer has the same format for all communication levels and is therefore compatible for all carrier systems supporting a 3x74 unit. HCC uses the following access methods:

- EXCP SINGLE HOST through local 3x74 control unit
- EXCP TOKEN-RING router through local 327x control unit
- VTAM NATIVE (local and remote)

The buffer for an IBM 3277/3278 unit with 24*80 character has the following format:

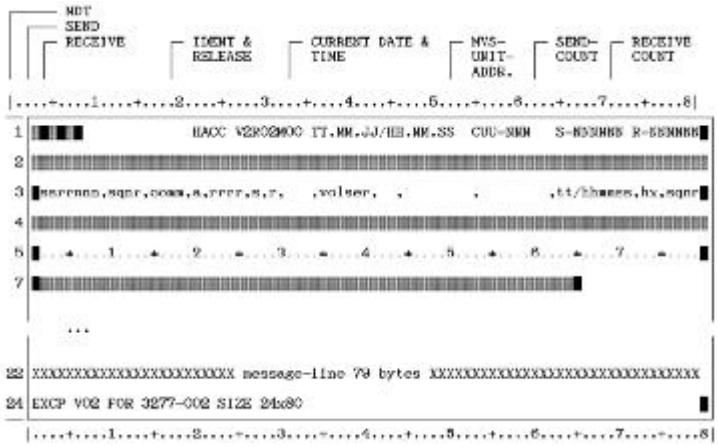


*Figure: 3278 buffer configuration (EXCP)*

| | |
|---|---|
| (1,1) MDT-CONTROL: | 0 reset MDT-bits of all unprotected fields by HCC<br>1 no reset MDT-bits by HCC |
| (1,3) SEND-CONTROL: | 0 data send allowed (apart from line 4)<br>1 data send suppressed for line 2 |
| (1,5) RECEIVE-CONTROL: | 0 no data present in line 4<br>1 data present in line 4 |
| (2,1) SEND LINE | 80 bytes (refer to Command Description) |
| (4,1) RECEIVE LINE | 80 bytes (refer to Command Description) |
| (7,2) COMMAND LINE | 64 bytes |

# 5 AML MESSAGES

## 5.1 MESSAGE FORMAT

Two message types are used for HCC-AML communication:

1. asynchronous messages
2. synchronous messages

All messages are issued in the HCC-AML standard format:

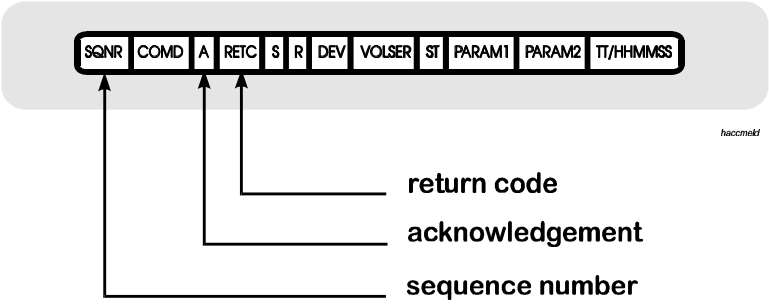


*Fig. 1: HCC-AML Standard format* *haccmeld*

The following messages are shown in the form:

```
A,RETC message text
```

Definitions:
Synchronous messages:
as reply to HCC initiated requests,
SQNR 4 bytes numeric in the range 0000-9999,
acknowledgment: P = positive
E = end condition
N = negative, return code available

Asynchronous messages:
initiated by ROBOT-IC or AMU or AR (Archive computer),
SQNR 4 bytes alphanumeric,
acknowledgment: W = warning indicator
M = robot status

W,W000 Warning with sense code in PARAM1, precedes synchronous messages
M,ROBS Master message on robot status changes
Asynchronous messages can arise at any time.

## 5.2 MESSAGES

**N001:** SYNTAX ERROR

**N002:** UNEXPECTED RESPONSE FROM ROBOT, SEVERE ERROR

**N003:** SEVERE ERROR IN SETUP FILE OF ARCHIVE PC

**N004:** SEVERE ERROR IN ARCHIVE

**N005:** ROBOT NOT READY

**N006:** ROBOT ERROR (SEE PRECEEDING WARNING)

**N007:** NOT EXECUTABLE ROBOT COMMAND

**N008:** ROBOT EMERGENCY SHUTDOWN

**N009:** ROBOT SWITCHED INTO ADJUSTMENT MODE

**N010:** UNKNOWN ROBOT COMMAND

**N011:** COORDINATE ASSIGNMENT FOR WRONG ROBOT

**N012:** COMMAND INTERRUPTED BY MANUAL INTERVENTION

**N014:** COMMAND INTERRUPTED BY PROGRAM REQUEST

**N015:** TOWER HAS NOT TURNED INTO POSITION

**N016:** ROBOT ERROR (EXCP-OUT 5001)

**N017:** COMMAND NOT EXECUTABLE

**N100:** ROBOT CRASH

**N101:** ROBOT CRASH AT MEDIUM LOAD/STORE

**N102:** TIMEOUT (ROBOT ERROR)

**N103:** TIMEOUT IN PC-IC COMMUNICATION

**N104:** MEDIUM LOST

**N105:** MEDIUM STUCK IN GRIPPER

**N110:** CRASH AT TAKING A MEDIUM FROM A BOX

**N111:** CRASH AT INSERTING A MEDIUM TO A BOX

**N112:** CRASH AT TAKING A MEDIUM FROM A DEVICE

**N113:** CRASH AT INSERTING A MEDIUM INTO A DEVICE

**N201:** UNKNOWN DEVICE

**N202:** TAPE UNIT STILL OCCUPIED

**N203:** DEVICE EMPTY

**N204:** DEVICE OCCUPIED

**N205:** DEVICE EMPTY

**N206:** MEDIUM CAN NOT BE ACCESSED FROM DEVICE

**N207:** OPEN ENTRY DOOR AT TAPE UNIT NOT LOCKABLE

**N208:** MEDIUM CANNOT BE TAKEN FROM STACK

**N301:** UNKNOWN VOLSER

**N302:** VOLSER NOT IN LIBRARY

**N304:** BARCODE LABEL NOT READABLE

**N305:** NO MEDIUM FOUND IN BOX

**N306:** WRONG MEDIUM ON SPECIFIED COORDINATE

**N401:** COORDINATE NOT DEFINED

**N402:** NO MEDIUM ON SPECIFIED COORDINATE

**N403:** POSITION NOT EMPTY

**N408:** GRIPPER ERROR - MEDIUM MAY BE DROPPED

**N501:** DOOR OF I/O-DEVICE IS NOT CLOSED

**N502:** I/O-AREA DEFINITIONS ARE NOT AGREED

**N503:** EJECT DEVICE OVERFLOW

**N504:** PUT INTO PROBLEMBOX

**N505:** PUT INTO PROBLEMBOX, PROBLEMBOX IS FULL NOW

**N506:** WRONG VOLSER - MEDIA MOVED INTO PROBLEMBOX

Use the HCC command **DRE** to display the significance of the AML messages.

Further codes can be found in the AML or AMU Operating Manuals.

# 6 MVS MESSAGES

The following MVS operating system messages, including HSM and non-IBM software products, must be available to HCC through SSI broadcasting. No suppression mechanisms may be active for messages for example, IEECVXIT or JES3-Exits with regard to the required messages.

The exact message format can be found in the relative literature.

## 6.1 MOUNT MESSAGES

```
IEC400A M cuu,volser,...
IEC501A M cuu,volser,...
IEF233A M cuu,volser,...
IEF233D M cuu,volser,...
IEC705I TAPE ON cuu,volser...
IAT5210 ...
```

## 6.2 KEEP MESSAGES

```
IEF234E (K¦R¦D) cuu,volser,...
IEC502E (K¦R¦D¦RK¦RD),cuu,volser,...
IAT5410 ...
```

## 6.3 EXCEPTION MESSAGES

```
IEC135A U cuu,volser      (not file protected RACF)
IEC507D E cuu.volser,...  (expiration date not occurred)
IEC509A F cuu,volser      (file protect ring required)
IEC701D M cuu,..volser    (IEHINIT tape labeling)
IGF500I  SWAP cuu to cuu ...
IGF502E  PROCEED WITH SWAP OF cuu TO cuu
IGF505I  SWAP FROM cuu TO cuu COMPLETE
IGF509I  SWAP cuu - I/O ERROR
IEF244I jobname,step - UNABLE TO ALLOCATE...
IEF290E jobname,step - UNABLE TO ALLOCATE...
IEF289I jobname,step,ddname NEEDS...
IEF247I jobname - cuu,...
```

## 6.4 WTOR MESSAGES

```
IEC507D REPLY 'U'-USE or 'M'-UNLOAD...
IEC510D F cuu,volser      (file protect ring required)
IEC534D A ddd,volser,... REPLY 'U'=USE OR 'M'=UNLOAD
IEF433D jobname - WAIT REQUESTED - REPLY HOLD OR NOHOLD
IEF238D jobname - n UNIT(S) NEEDED ...
IGF500D  REPLY 'YES','DEVICE','NO'...
IGF509D  REPLY DEVICE OR 'NO'
```

## 6.5 HSM WTOR MESSAGES

```
ARC0310A CAN TAPE ...
ARC0311A SYSTEM STIMER ...
ARC0312A TAPES ... NEEDED
ARC0314A CAN THE ...
ARC0366A REPLY ...
ARC0380A RECALL WAITING ...
ARC0381A ALLOCATION REQUEST ...
```

## 6.6 NON-IBM MESSAGES

### 6.6.1.1 CA1

```
TMS...   (max. 'volser' 6 bytes)
```

### 6.6.1.2 STAM

```
STM202I ...
```

# 7 SUMMARIES

The following SUMMARIES are listed in detail:

- MODUL SUMMARY
- MACRO SUMMARY

## 7.1 MODULE SUMMARY

HCC comprises a larger number of modules working together in subtasks and subroutines. HCC only becomes active when an event based on the WAIT-POST-principle arrives.

| MODUL | FUNCTION | FETCH | LINK-ATTR. |
|---|---|---|---|
| ZHC00100 | System-Entry, Name of Started Task | MSTR | AC(1) |
| ZHC 00200 | Message Issuer to Operator (WTO,TPUT) | LOAD | RENT |
| ZHC 00300 | Send Message to ROBCOM | LOAD | |
| ZHC 00400 | ROB Error messages and action codes | LOAD | |
| ZHC 00500 | ROBCOM Dispatcher | LOAD | |
| ZHC 00600 | ROBCOM priority control | LOAD | |
| ZHC 00700 | ROBCOM return handler | LOAD | |
| ZHC 00800 | HACICT/HACICE/HACICC control blocks | LOAD | |
| ZHC 01000 | RACF-interface | LOAD | |
| ZHC 01100 | ROB-Error-Analyzer | LOAD | |
| ZHC 01200 | SSI EXIT-Routine | LOAD | RENT |
| ZHC 01300 | Read HCC-Parms HACPARM1 | LOAD | |
| ZHC 01400 | HCC-Command Syntax-Checker | LOAD | |
| ZHC 01500 | HCC-Command-Processor HCC0 | LOAD | |
| ZHC 01600 | Convert HEX to EBCDIC | LOAD | |
| ZHC 01700 | HACCVT and SSI Initialization | LOAD | |
| ZHC 01800 | MVS-Message Filter | LOAD | |
| ZHC 01900 | MOUNT-Processing | LOAD | |
| ZHC 02000 | Binary Search | LOAD | RENT |
| ZHC 02100 | Incore/inplace Sort | LOAD | RENT |
| ZHC 02200 | FULLSCR TPUT/TGET Support | LOAD | |
| ZHC 02300 | HCC Initialization/Recovery | LOAD | |
| ZHC 02400 | Create FPMA SLUTAB, complete FPMATAB | LOAD | |
| ZHC U2500 | Tape Management User Exit UCC1 | LOAD | |
| ZHC T2500 | Tape Management User Exit TLMS | LOAD | |
| ZHC S2500 | Tape Management User Exit seq.file | LOAD | |
| ZHC 02600 | Display CSA-information | LOAD | |
| ZHC 02700 | Setup UCB-Addresses | LOAD | |
| ZHC 02800 | Table of authorized TSO-Users | LOAD | |
| ZHC 02900 | Convert DATE/TIME-BIN | LOAD | RENT |
| ZHC 03000 | Messages HAC000-HAC099 | LOAD | RENT |
| ZHC 03100 | Messages HAC100-HAC199 | LOAD | RENT |
| ZHC 03200 | Messages HAC200-HAC299 | LOAD | RENT |
| ZHC 03300 | Messages HAC300-HAC399 | LOAD | RENT |
| ZHC 03400 | Messages HAC400-HAC499 | LOAD | RENT |
| ZHC 03700 | VI-exit sample | LOAD | |
| ZHC 03800 | EJ-exit sample | LOAD | |
| ZHC 03900 | Display or modify storage | LOAD | |
| ZHC 04100 | Command processor HOST complex | LOAD | |
| ZHC 04200 | First level return handler HOST complex | LOAD | |
| ZHC 04300 | Command processor loosely coupled HOSTs | LOAD | |
| ZHC 04400 | Return handler loosely coupled HOSTs | LOAD | |
| ZHC 04500 | Display/VARY Command-Processor | LOAD | |
| ZHC 04600 | FPMA command processor | LOAD | |
| ZHC 05000 | Read/Update CUULOG | ATTACH | |
| ZHC 05200 | SSI-Message Buffering | ATTACH | |

| MODUL | FUNCTION | FETCH | LINK-ATTR. |
|---|---|---|---|
| ZHC05300 | STIMER Reawaken | ATTACH | |
| ZHC05400 | APPC-Maintask | ATTACH | |
| ZHC05500 | APPC-Send-Subtask | ATTACH | |
| ZHC05600 | APPC-Receive-Subtask | ATTACH | |
| ZHC05800 | Exit Driver for TMS-Exit ZHC02500 | ATTACH | |
| ZHC05900 | Read/Update Archive | ATTACH | |
| ZHC06000 | Write HCC-MESSAGE-LOG | ATTACH | |
| ZHC06100 | ROBTEST 1 | ATTACH | |
| ZHC06200 | ROBTEST 2 | ATTACH | |
| ZHC06300 | ROBTEST 3 | ATTACH | |
| ZHC06400 | ROBTEST 4 | ATTACH | |
| ZHC06500 | EXCP-/VTAM LU 2 Interface to AML | ATTACH | |
| ZHC06900 | Subfunction processor | LOAD | |
| ZHC07000 | Read/Write Exchange log | ATTACH | |
| | associated alias modules ZHC07100-7700 | ATTACH | |
| ZHC08000 | Display tape header labels (independent) | MSTR | |
| ZHC08100 | Display and Test HCC-Messages | LOAD | |
| ZHC08200 | Unfold FULLSCREEN for printing the map | LOAD | |
| ZHC08300 | ABBASEND EXCP subtask | ATTACH | AC(1) |
| ZHC08400 | Read/Update Archive (ABBASEND) | LOAD | |
| ZHC08600 | TMSCHECK - TSO test for TMS user exit | LOAD | |
| ZHC08800 | Foreground compilation (ASM-H) | TSO | |
| ZHC08900 | Foreground link (IEWL) | LINK | |
| ZHC09000 | Attention exit | LOAD | RENT IMPL. |
| ZHC10100 | HACC0-Maintask | XCTL | |
| ZHC10300 | ABBASEND | INIT/TSO | AC(1) |
| ZHC10400 | ABBASEND new with output | LOAD | |
| | message modules for GERMAN language | | |
| ZHCG1400 | HCC-Command Syntax-Checker | LOAD | |
| | Optional: | | |
| ZHC20200 | SSI initialization for HACPARM2 | MSTR | |
| ZHC20300 | Independent call for ZHC20200 | MSTR | |
| ZHC20400 | IGG026DU,ZHC026DU | SVC | |
| ZHC20500 | IGG026DU,ARCATEX,ZHC026DU (Coexistence) | SVC | |
| | ISPF-Module: | | |
| ZHC40000 | Move a Command to HCC | | |
| ZHC40100 | Display HCC-Buffers | | |
| ZHC40200 | Initialize HCC/ISPF | | |
| ZHC4300 | Display HCC-Buffers | | |
| ZHC41000 | TSF Program to do authorized work in ISPF | | |
| ZHC42000 | Call HACPARM1 Checker | | AC(1) |

| MODUL | FUNCTION | FETCH | LINK-ATTR. |
|---|---|---|---|
| | HCC ARCHIVE ADMINISTRATION (HAA) | | |
| ZHC30000 | HAA command processor | TSO | |
| ZHC30100 | General information | LOAD | |
| ZHC30200 | Volser administration | LOAD | |
| ZHC30300 | Coordinate generation | LOAD | |
| ZHC30400 | Save/restore/copy | LOAD | |
| ZHC30500 | Statistics | LOAD | |
| ZHC31100 | Subfunction: CALL IDCAMS to retrieve catalog | LOAD | |
| ZHC31200 | Subfunction: generate volser records | LOAD | |
| ZHC31300 | Subfunction: unload archive | LOAD | |
| ZHC31400 | Subfunction: reload archive | LOAD | |
| ZHC31500 | Subfunction: generate coords. from HACPARM1 | LOAD | |
| ZHC31600 | Subfunction: merge existent and new records | LOAD | |
| ZHC31700 | Subfunction: execute SORT/MERGE program | LOAD | |
| ZHC31800 | Subfunction: copy archive | LOAD | |
| ZHC31900 | Subfunction: modify volser entries | LOAD | |
| ZHC32000 | Subfunction: display volser entries | LOAD | |
| ZHC32100 | Subfunction: generate print-out coordinates | LOAD | |
| ZHC32200 | Subfunction: convert archive version 1 to 2 | LOAD | |
| ZHC32300 | Subfunction: print ejected tapes | LOAD | |
| ZHC32400 | Subfunction: print inserted tapes | LOAD | |
| ZHC32500 | Subfunction: print scratch tapes | LOAD | |
| ZHC32600 | Subfunction: print swap statistic | LOAD | |

## 7.2 MACRO SUMMARY

This Section gives a summary of HCC Macros and MVS Macros used. For local HCC compilations, the customer has to provide the MVS distribution library.

| MACRO | DESCRIPTION |
|---|---|
| HACAER | Robot error action codes for each rob command |
| HACAM | Switch AMODE focus 31->24 24->31 |
| HACAPP | Data Area for APPC Send/Recv. Subtasks |
| HACASR | Archive record |
| HACBIN | Binary search parmlist |
| HACCMS | Command stack |
| HACCOM | HCC-AML EXCP communication definitions |
| HACCOO | Coordinate AIX |
| HACCUU | Unit statistic and recovery log entry (primary HCC) |
| HACCVT | HCC communications vector table |
| HACCXP | Queue entry for cross memory services |
| HACDGR | Scratch tape access for generic dsname groups |
| HACDMP | Unhex and issue dump data (if trace on) |
| HACDOM | Delete descriptor code 2 WTO messages |
| HACDQR | Common data queue entry robot communication |
| HACEDT | Esoteric device table |
| HACEJD | EJECT-device description |
| HACEQU | Register equates |
| HACESD | EJECT-selection by dsnames |
| HACESV | EJECT-selection by volumes |
| HACEXC | Exchange log record |
| HACEXL | Exchange log pointer list |
| HACEX1 | Incoming parmlist for the TMS-user-exit (ZHC02500) |
| HACFMD | Foreign mount pointer table |
| HACFPD | FPMA entry |
| HACFTA | HACC command function entry |
| HACFUS | Function Stack entry description |
| HACFVD | Free volume tab pointer list description |
| HACHCO | HOST communication table entry description |
| HACICC | Intercommunication constants |
| HACICE | Intercommunication table extensions |
| HACICT | Intercommunication table |
| HACJBP | Job priority control |
| HACLBL | Record description HDR1/HDR2 of standard tape header labels |
| HACLDV | Library device coordinates |
| HACLGD | Message log record (in LOG-datasets) |
| HACLOG | Message-log record (internal) |
| HACMGN | Message generation |
| HACMPD | MPMA entry |
| HACMSG | Message issue |
| HACMnn | Special messages |
| HACOMF | MVS-message format (HACXIT-structure) |
| HACPAT | Patch area, literal pool |
| HACPGM | Program Table entry description |
| HACQDR | ROBOT system queue-pointer-list |
| HACQDS | AML-system queue-pointer-list |

| MACRO | DESCRIPTION |
|---|---|
| HACRCA | MOUNT/KEEP-recovery area |
| HACRCD | Rack device definition |
| HACRMF | Robot communication record format |
| HACRQD | Robqueue description |
| HACRTD | Pointer list for robtest-subtasks |
| HACSEC | Security interface |
| HACSLU | FPMA selection by unit rotation |
| HACSRT | Parmlist incore-sort |
| HACSSN | Globals: SSI-name,HACCVT-identifier,HACEDT-id. |
| HACSUE | Subroutine entry |
| HACSUX | Subroutine exit |
| HACTMD | Time/date conversion parmlist |
| HACTME | Call time/date conversion routine |
| HACTSC | TSO user security |
| HACTWD | Tower device definition |
| HACUPE | Subroutine Entry/Exit Macros |
| HACUPX | |
| HACVGR | Scratch tape access for volser groups |
| HACVID | Insert device definition |
| HACWBD | Library device definition (waste box) |
| HACXCV | EBCDIC/HEX conversion |
| HACXIT | SSI communication exit parmlist in HACCVT |
| HACXME | ABS Output Buffer Data Entry Description |
| HACXMH | ABS Output Buffer Header Description |
| HACXPD | EXCP-parm fields for subtasks ZHC06500/66/6700/6800 |
| HACXSB | EXCP SEND BUFFER |
| | HAA-COMMAND |
| HAAICT | Intercommunication table |
| HAAVFY | Coordinates checking |
| HAAVGT | Volser records generation |
| | TSO, VTAM and 327x-EXCP |
| $BA | Define buffer address |
| $IC | Insert cursor address |
| $RA | Repeat character |
| $SBA | Define and set buffer address |
| $SF | Define and set attribute character |
| $SF11 | Define and set attribute character and set buffer control |

| MACRO | MVS MACROS |
|---|---|
| CVT | |
| IEECUCM | |
| IEESMCA | |
| IEFJESCT | |
| IEFJSCVT | |
| IEFJSCB | |
| IEZJSCB | |
| IEFUCBOB | |
| IEFJSSOB | |
| IEZCOM | |
| IEZCIB | |
| IHAPSA | |
| IHAASVT | |
| IHAASCB | |
| IHAORE | |
| IHASDWA | |
| IHAWQE | |
| IKJTCB | |
| LOCASCB | |
| MGCR | |
| SYSEVENT | |

# 8 RECORD FORMATS

The following HCC records are listed in detail:

- ARCHIVE MIRROR
- MESSAGE LOG
- EXCP READ/WRITE

## 8.1 ARCHIVE VOLSER

COMMON NAME: ARCHIVE VOLSER / INCORE RECORD
MACRO ID: ZHCASR
RECORD SIZE: 8-84 BYTES,VARIABLE LENGTH
DATA FORMAT: VSAM-KSDS,KEYLEN=6,RKP=0

```
DEC  HEX   TYPE      LENGTH  NAME     DESCRIPTION
-------------------------------------------------------------------------
 0    0    STRUCTURE   84   ARENTRY  ARCHIVE VOLSER RECORD
 0    0    STRUCTURE   14   ARVLENT  INCORE VOLSER ENTRY  OD-SIDE A
 0    0    CHARACTER    6   ARVOL    VOLSER (KEY)
-------------------------------------------------------------------------
 6    6    BITSTRING    1   ARSTA    Volume Status for Cartridges
           1... ....        ARXCLT   - CLEANING TAPE else DATATAPE
           .1.. ....        ARXINA   - IN ARCHIVE else EJECTED
           ..1. ....        ARXEJD   - EJECT TO BE DONE
           ...1 ....        ARXSCR   - SCRATCH TAPE
           .... 1...        ARXOD    - OD Bit must be OFF
           .... .1..        ARXIDEV  - IN VOLUME INSERT DEVICE
           .... ..1.        ARXFPMA  - IN FPMA
           .... ...1        ARXHOME  - ON HOME POSITION
-------------------------------------------------------------------------
           Redefinitions of ARSTA for optical Disk
-------------------------------------------------------------------------
 6    6    BITSTRING    1   AROSTA   Volume Status for optical Disk
           1... ....        AROXMOU  - mounted on OAD else HOME or in JB
                                       (only useful if Bit ARXALIAS is OFF)
           .1.. ....        AROXINA  - IN ARCHIVE else EJECTED
           ..1. ....        AROXEJD  - EJECT TO BE DONE
           ...1 ....        AROXSCR  - SCRATCH TAPE
           .... 1...        AROXOD   - OD Bit must be ON
           .... .1..        AROXIDEV - IN VOLUME INSERT DEVICE
           .... ..1.        AROJB    - Volume in Jukebox
                                       (only useful if Bit ARXALIAS is OFF)
           .... ...1        AROXHOME - ON HOME POSITION
                                       (only useful if Bit ARXALIAS is OFF)

-------------------------------------  Comment  --------------------------
           valid combinations of AROXMOU ,AROXHOME, AROXJB and AROXINA :
           These combinations are only valid if AROXOD Bit is ON

           AROXMOU      AROXHOME      AROXINA      AROXJB      Status
              1            1            1            0          OM
              0            1            1            0          OB
              0            1            1            1          OJ
              0            1            0            0          OE

---------------------------------  End of Comment  ----------------------
```

| DEC | HEX | TYPE | LENGTH | NAME | DESCRIPTION |
|---|---|---|---|---|---|
| 7 | 7 | BITSTRING | 1 | ARSTA1 | RECOVERY STATUS |
| | | 1... .... | | ARXVIR | - IS VIRTUAL TAPE |
| | | .1.. .... | | ARXMVD | - MOVE TO BE DONE |
| | | ..1. .... | | ARXMVH | - MOVE HOME |
| | | ...1 .... | | ARXALIAS | - SECOND SIDE FOR OPTICAL DISC |
| | | .... 1... | | ARXMP | - Manual Pool - MAPS |
| | | .... .1.. | | ARXHWS | - in HWS (HACC WORK STORAGE) |
| | | .... ..1. | | ARXINIT | - INITIAL VOLSER ENTRY |
| | | .... ...1 | | ARXINVC | - INVALID COORDINATE |

**Offsets**

| DEC | HEX | TYPE | LENGTH | NAME | DESCRIPTION |
|---|---|---|---|---|---|
| 8 | 8 | BITSTRING | 1 | ARFLG | STATUS FLAG USED BY MAINTASK |
| | | 1... .... | | ARXINQ | - VOLUME WAS PLACED INTO ROBQUE |
| | | .1.. .... | | ARXIUS | - VOLUME IN USE(ON UNIT,IN EJECT..) |
| | | ..1. .... | | ARXART | - ALLOCATED FOR ROBTEST FUNCTIONS |
| | | ...1 .... | | ARXFKES | - FORCE KES DUE TO ROB UNACCESSABILITY |
| | | .... 1... | | ARXCLC | - CURRENT CLEANING TAPE |
| | | .... .1.. | | ARXPML | - PMA LOAD PENDING |
| | | .... ..1. | | ARXPMU | - PMA UNLOAD PENDING |
| 9 | 9 | HEX | 1 | ARSYS | ABBA SYSTEM TO WHICH TAPE BELONGS |
| | | .... ...1 | | ARXSYS1 | - SYSTEM 1 |
| | | .... ..1. | | ARXSYS2 | - SYSTEM 2 |
| 10 | A | HEX | 1 | ARROB | ROBOT WHICH CAN ACCESS THE TAPE |
| | | .... ...1 | | ARXROB1 | - ROBOT 1 |
| | | .... ..1. | | ARXROB2 | - ROBOT 2 |
| 11 | B | HEX | 1 | AREJDEV | EJECT-DEVICE-NO (RECOVERY ONLY) |
| 12 | C | UNSIGNED | 4 | ARHCO | HOME-COORDINATE |
| 12 | C | UNSIGNED | 1 | ARHTOW | - TOWER NO./SIDE NO. |
| 13 | D | UNSIGNED | 1 | ARHSEG | - SEGMENT NO. |
| 14 | E | UNSIGNED | 1 | ARHROW | - ROW IN SEGMENT |
| 15 | F | UNSIGNED | 1 | ARHPOS | - POSITION IN ROW |
| 16 | 10 | UNSIGNED | 4 | ARTCO | TEMPORARY COORDINATE |
| 16 | 10 | UNSIGNED | 1 | ARTTOW | - TOWER NO./SIDE NO. |
| 17 | 11 | UNSIGNED | 1 | ARTSEG | - SEGMENT NO. |
| 18 | 12 | UNSIGNED | 1 | ARTROW | - ROW IN SEGMENT |
| 19 | 13 | UNSIGNED | 1 | ARTPOS | - POSITION IN ROW |
| | | B-Side redefinitions of ARHCO and ARTCO for optical Disk | | | |
| 12 | C | CHARACTER | 6 | AROVOL | VOLSER of A-Side |
| 18 | 12 | CHARACTER | 2 | ARORES1 | free |
| | | A-Side redefinition of ARTCO for optical Disk | | | |
| 16 | 10 | UNSIGNED | 1 | AROINDL | Index into LIBTAB, if Status=J |
| 17 | 11 | UNSIGNED | 3 | AROINDV | Index into ASRTAB for B-VOLSER |
| 19 | 13 | X'14' | | ARLI | *-ARVLENT - Length of Incore-Record |
| 198 | C6 | CHARACTER | 2 | FUSRES2 | unused |
| 200 | C8 | STRUCTURE | 48 | FUSSUBT | Subtask Work-area |
| 200 | C8 | STRUCTURE | 48 | FUS59W | Workarea for ZHC05900 |
| 200 | C8 | UNSIGNED | 1 | FUS59REC | ASR-Rectype |
| 201 | C9 | BITSTRING | 1 | FUS59ACT | ASR-Rectype |
| | | 1... .... | | FUS59UPD | Perform an update operation |
| | | .1.. .... | | FUS59GET | Read a record |
| 202 | CA | UNSIGNED | 2 | | Free for further use |
| 204 | CC | CHARACTER | 44 | FUS59PTR | Free for use with Subtask-work |
| | | X'30' | | FUS59LEN | *-FUS59 - Length of ZHC05900 Workarea |
| | | X'F8' | | FUSCL1 | *-FUSQUE - Used Length of FUS-Entry |
| | | X'108' | | FUSCL2 | (512-FUSCL1) - remaining Length |

## 8.2 MESSAGE LOG

COMMON NAME: MESSAGE LOG RECORD
MACRO ID: HACLGD
RECORD SIZE: 132 BYTES (LIST OUTPUT FORMAT)
DATA ACCESS: BSAM

| DEC | HEX | TYPE | LENGTH | NAME | DESCRIPTION |
|---|---|---|---|---|---|
| 0 | 0 | STRUCTURE | 132 | LRECORD | LOG RECORD |
| 0 | 0 | CHARACTER | 3 | LRTYP | MESSAGE TYPE |
| | | | | | - 'ROB' ROBOT COMMUNICATION |
| | | | | | - 'CMD' HACC-COMMAND |
| | | | | | - 'MVS' MVS-MESSAGE |
| | | | | | - 'MSG' HACC-MESSAGE |
| | | | | | - 'SWI' LOG-SWITCH INDICATOR |
| 3 | 3 | CHARACTER | 1 | * | BLANK-CHARACTER |
| 4 | 4 | CHARACTER | 5 | LRDATE | DATE 'YYDDD' |
| 9 | 9 | CHARACTER | 1 | * | ' '-CHARACTER |
| 10 | A | CHARACTER | 8 | LRTIME | TIME 'HH/MM/SS' |
| 18 | 12 | CHARACTER | 1 | * | BLANK-CHARACTER |
| 19 | 13 | CHARACTER | 4 | LRCPU | CPUID ORIGINATING THE LOG ENTRY |
| 23 | 17 | CHARACTER | 1 | * | BLANK-CHARACTER |
| 24 | 18 | CHARACTER | 8 | LRJOB | JOBNAME RELATED TO THE LOG ENTRY |
| 32 | 20 | CHARACTER | 1 | * | BLANK-CHARACTER |
| 33 | 21 | CHARACTER | 87 | LRMSG | MESSAGE TEXT |
| 120 | 78 | CHARACTER | 1 | * | BLANK-CHARACTER |
| 121 | 79 | CHARACTER | 2 | LREPID | WQE-REPLYID (OR BLANK IF ABSENT) |
| 123 | 7B | CHARACTER | 1 | * | BLANK-CHARACTER |
| 124 | 7C | CHARACTER | 8 | LRECSQ | RECOVERY-SEQ.NO (OR BLANK IF ABSEN) |
| | | 1... .1.. | 8 | LRECL | *-LRECORD,LENGTH OF LOG-RECORD |

## 8.3 EXCP READ/WRITE

COMMON NAME: ROBCOM MESSAGE FORMAT
MACRO ID: HACRMF
RECORD SIZE: 80 BYTES,FIXED
DATA FORMAT: SEQUENTIAL

| DEC | HEX | TYPE | LENGTH | NAME | DESCRIPTION |
|---|---|---|---|---|---|
| 0 | 0 | STRUCTURE | 80 | ABAMSG | ROBCOM MESSAGE RECORD |
| message header | | | | | |
| 0 | 0 | STRUCTURE | 9 | ABHDR | MESSAGE HEADER |
| 0 | 0 | CHARACTER | 1 | ABASM | START MESSAGE ID |
| | | | | | - '<' BEGIN BRACKET |
| 1 | 1 | CHARACTER | 2 | ABRE | MESSAGE RECEIVER ("EMPFÄNGER") |
| | | | | | - 'Hn' HACC0 ID (n=1-N) |
| | | | | | - 'Mn' MAJOR ID (n=1-N) |
| | | | | | - 'An' ABBA SYSTEM ID (n=1-N) |
| | | | | | - 'T1' COMMUNICATION ID |
| 3 | 3 | CHARACTER | 2 | ABSE | MESSAGE ISSUER ("ABSENDER") |
| | | | | | - 'Hn' HACC0 ID (n=1-N) |
| | | | | | - 'Mn' MAJOR ID (n=1-N) |
| | | | | | - 'An' ABBA SYSTEM ID (n=1-N) |
| | | | | | - 'T1' COMMUNICATION ID |
| 5 | 5 | CHARACTER | 1 | ABMT | MESSAGE TYPE |
| | | | | | - 'A' ABBA COMMUNICATION |
| | | | | | - 'H' HOST-HOST COMMUNICATION |
| | | | | | - 'T' TOKEN-HOST COMMUNICATION |
| 6 | 6 | CHARACTER | 2 | ABRT | RECORD TYPE |
| | | | | | - '00' ABBA |
| 8 | 8 | CHARACTER | 1 | * | DELIMETER VALUE ',' |
| | | .... 1..1 | | ABHL | *-ABHDR, LENGTH OF HEADER |
| ROBCOM data entry | | | | | |
| 9 | 9 | STRUCTURE | 70 | ABADATA | DATA-ENTRY MESSAGE TYPE 'A' |
| 9 | 9 | CHARACTER | 4 | ABASNR | SEQUENCE NO (MAJOR) |
| 13 | D | CHARACTER | 1 | * | ','-CHARACTER |
| 14 | E | CHARACTER | 4 | ABACOM | COMMAND |
| 18 | 12 | CHARACTER | 1 | * | ','-CHARACTER |
| 19 | 13 | CHARACTER | 1 | ABARETF | RETURN FLAG (ACKNOWLEDGEMENT) |
| | | | | | - 'P' POSITIVE |
| | | | | | - 'N' NEGATIVE |
| | | | | | - 'E' END OF FUNCTION |
| | | | | | - 'W' WARNING |
| 20 | 14 | CHARACTER | 1 | * | ','-CHARACTER |
| 21 | 15 | CHARACTER | 4 | ABARETC | COMMAND RETURN CODE |
| 25 | 19 | CHARACTER | 1 | * | ','-CHARACTER |

| DEC | HEX | TYPE | LENGTH | NAME | DESCRIPTION |
|---|---|---|---|---|---|
| 26 | 1A | CHARACTER | 1 | ABASYS | ABBA SYSTEM NUMBER |
| | | | | | - '1' SYSTEM 1 |
| | | | | | - '2' SYSTEM 2 |
| 27 | 1B | CHARACTER | 1 | * | ','-CHARACTER |
| | | | | | - 'W' WASTE BOX |
| 28 | 1C | CHARACTER | 1 | ABAROB | ROBOT NUMBER |
| | | | | | - '1' ROBOT 1 |
| | | | | | - '2' ROBOT 2 |
| 29 | 1D | CHARACTER | 1 | * | ','-CHARACTER |
| 30 | 1C | STRUCTURE | 3 | ABAUNIT | ABBA UNIT |
| 30 | 1C | CHARACTER | 1 | ABAUTYP | DEVICE TYPE |
| | | | | | - 'D' TAPE UNIT |
| | | | | | - 'I' INSERT DEVICE |
| | | | | | - 'E' EJECT DEVICE |
| | | | | | - 'F' EXTERNAL MOUNT AREA |
| 31 | 1E | CHARACTER | 2 | ABAUNO | UNIT NUMBER |
| 33 | 21 | CHARACTER | 1 | * | ','-CHARACTER |
| 34 | 22 | CHARACTER | 6 | ABAVOL | VOLUME SERIAL NUMBER |
| 40 | 28 | CHARACTER | 1 | * | ','-CHARACTER |
| 41 | 29 | STRUCTURE | 2 | ABAVST | VOLSER STATUS |
| 41 | 29 | CHARACTER | 1 | ABAVSTL | LOGICAL STATUS |
| | | | | | - 'C' CLEANING CARTRIDGE |
| | | | | | - 'M' DATA CARTRIDGE |
| | | | | | - '0' INITIAL STATUS |
| 42 | 2A | CHARACTER | 1 | ABAVSTP | PHYSICAL STATUS |
| | | | | | - 'B' IN ARCHIVE |
| | | | | | - 'E' EJECTED |
| | | | | | - 'D' IN TAPE DEVICE |
| | | | | | - 'L' INITIAL STATUS |
| 43 | 2B | CHARACTER | 1 | * | ','-CHARACTER |
| 44 | 2C | CHARACTER | 8 | ABAPRM1 | SOURCE COORDINATE |
| 52 | 34 | CHARACTER | 1 | * | ','-CHARACTER |
| 53 | 35 | CHARACTER | 8 | ABAPRM2 | TARGET COORDINATE |
| 61 | 3D | CHARACTER | 1 | * | ','-CHARACTER |
| 62 | 3E | STRUCTURE | 9 | ABATIME | TIME STAMP |
| 62 | 3E | CHARACTER | 2 | ABATDD | DAY OF DATE (DD) |
| 64 | 40 | CHARACTER | 1 | * | '/'-CHARACTER |
| 65 | 41 | CHARACTER | 2 | ABATHH | HOURS (HH) |
| 67 | 43 | CHARACTER | 2 | ABATMM | MINUTES (MM) |
| 69 | 45 | CHARACTER | 2 | ABATSS | SECONDS (SS) |
| 71 | 47 | CHARACTER | 1 | * | ','-CHARACTER |
| 72 | 48 | CHARACTER | 2 | ABHID | HACC0 COMPLEX ID |
| 74 | 4A | CHARACTER | 1 | * | ','-CHARACTER |
| 75 | 4B | CHARACTER | 4 | ABSNRO | SEQUENCE NUMBER (ORIGINAL) |
| | | .1.. .11. | | ABADL | *-ABADATA, LENGTH OF DATA |

end of message

| | | | | | |
|---|---|---|---|---|---|
| 79 | 4F | CHARACTER | 1 | ABAEM | END MESSAGE |
| | | | | | - '>' END BRACKET |
| | | .1.1 .... | | ABAL | *-ABAMSG, TOTAL LENGTH |

# 9 HCC MESSAGES

## 9.1 MESSAGE FORMAT

```
HACnnni text
```

| | |
|---|---|
| HAC | HCC identification |
| nnn | Message number (000-999) |
| i | Indicator I=information, A=action |
| text | Message text |

Important messages are issued on the operator console with descriptor code 2 (highlight, action). The routing code for the operator console can be selected by the ROUTCD parameter of the HACPARM1 member.

The language of the message texts can be selected with the LANG parameter of the HACPARM1 member.

## 9.2 MESSAGES

Detailed information can be found in the MESSAGES AND CODES GUIDE for HCC/MVS 3.0.

# 10 COMPILE and LINK

It is sometimes necessary to compile modules during installation, for example, the TMS exit. This can be done either as a standard batch compilation or directly under TSO, whereby ASSEMBLER H (IEV90) must already be installed.

## 10.1 FOREGROUND

The following commands can be used:

**Compile and Link:**

```
HACASML member [RENT] [REUS] [TEST] [AC(1)] [CNTL]
               [MAC1(dsn1)] [MAC2(dsn2)] [MAC3(dsn3)] ...
               [LELIB1(dsnl1)] [LELIB2(dsnl2)] ...
               [LOAD(loadlib)] [LIB(sourcelib)]
```

**Compile:**

```
HACASM  member [RENT] [REUS] [TEST]
               [MAC1(dsn1)] [MAC2(dsn2)] [MAC3(dsn3)] ...
               [LIB(sourcelib)]
```

**Link:**

```
HACLINK member [RENT] [REUS] [TEST] [AC(1)] [CNTL]
               [LELIB1(dsnl1)] [LELIB2(dsnl2)] ...
               [LOAD(loadlib)]
```

| | |
|---|---|
| member | Sequential dataset with the name userid.member.ASM or userid.member.OBJ |
| MACn | Macro libraries (n=1-6 MACLIB concatenations)<br>Defaults: MAC1(SYS1.MACLIB), MAC3(SYS1.AMODGEN) |
| LELIBn | Autocall load library for unresolved references<br>Default: none |
| LOAD | Load library (SYSLMOD)<br>Default: LOAD(userid.LOAD), this dataset is automatically created if it does not exist |
| CNTL | Input prompt for any additional Linkage Editor statements |
| LIB | Source code input library, default: userid.member.ASM |

Example: compile and link ZHCU2500 into the steplib hlq.ZHC....HACCLOAD

```
HACASML ZHCU2500 REUS MAC2(HLQ.V2R3M1.HACCSAMP) MAC3(HLQ.V2R3M1.UCC1.mac)
LOAD(HLQ.V2R3M1.HACCLOAD) LIB(HLQ.V2R3M1.HACCSRC)
```

The list output name is userid.member.SYSPRINT.

## 10.2 BATCH COMPILE / LINK

The following job can be used to install the TMS exit (and possibly other modules as well) when a standard compile procedure is not available:

```
//jobname  JOB ...
//*------- COMPILE ------------------------------------------------
//ASM      EXEC PGM=IEV90,PARM='LIST,DECK,LINECNT(66)'
//SYSLIB   DD   DSN=SYS1.MACLIB,DISP=SHR
//         DD   DSN=SYS1.MODGEN,DISP=SHR
//         DD   DSN=HLQ.V2R3M1.HACCSAMP,DISP=SHR     <- HACC dlib
//         DD   DSN=HLQ.V2R3M1.UCC1.MAC,DISP=SHR     <- UCC1 maclib
//SYSUT1   DD   UNIT=SYSDA,SPACE=(CYL,(1,1)
//SYSPRINT DD   SYSOUT=*
//SYSPUNCH DD   DSN=&&OBJ,DISP=(,PASS),UNIT=SYSDA,SPACE=(TRK,(1,1))
//SYSIN    DD   DSN=input-member,DISP=SHR
//*------- LINK ---------------------------------------------------
//LINK     EXEC PGM=IEWL,PARM='LIST,REUS,XREF,LET,(SIZE=512K,100K)'
//SYSPRINT DD   SYSOUT=*
//SYSUT1   DD   UNIT=SYSDA,SPACE=(CYL,(1,1)
//SYSLMOD  DD   DSN=HLQ.V2R3M1.HACCLOAD,DISP=SHR<- HACC STEPLIB
//SYSLIN   DD   DSN=&&OBJ,DISP=(OLD,DELETE)
//         DD   *
  SETCODE AC(1) (when required, see module summary)
  NAME member(R)
/*
```

# 11 DISPLAY AND INITIALIZE HEADER

The independent utility ZHC08000 serves to display header information from IBM standard HDR1 and HDR2 labels, and to initialize these headers.

## 11.1 JCL START PROCEDURE

The following JCL statements must be included in the SYS1.PROCLIB or the chained user.PROCLIB, whereby the member name is arbitrary (LCK for header display and TLI for initialization in the example):

### Member name LCK:

```
//LCK      PROC V=ANYTAPE,U=unitname,FILENO=1
//ST010    EXEC PGM=ZHC08000,PARM='VOL=&V,UNIT=&U,FILENO=&FILENO'
//STEPLIB  DD   DSN=HLQ.V2R3M1.HACCLOAD,DISP=SHR
//SYSUDUMP DD   SYSOUT=A
```

### Member name TLI:

```
//TLI      PROC V=ANYTAPE,U=unitname
//ST010    EXEC PGM=ZHC08000,PARM='VOL=&V,UNIT=&U,TLI'
//STEPLIB  DD   DSN=HLQ.V2R3M1.HACCLOAD,DISP=SHR
//SYSUDUMP DD   SYSOUT=A
```

The PARM entries for these functions differ for safety reasons: FILENO and TLI are mutually exclusive.

## 11.2 START COMMANDS

| | |
|---|---|
| **S LCK,V=volser[,U=unit,FILENO=n]** | **Display header** |
| **S TLI[,U=unit]** | **Initialize** |

| | |
|---|---|
| volser | VOLSER |
| unit | MVS unit address, esoteric or generic name (ROB0, 3420, 3480, 3490 ...) |
| n | FILE Number of the file on the tape |

## 11.3 START PARAMETER TCPU

```
TCPU=xxxx
```

xxxx SMF-CPUId (different ID to the one of the system on which HCC runs)

This parameter supports using a different system than the one defined. This is required to simulate a HCC Major/Minor complex.

# 12 HCC/APPC displays

## 12.1 HCC-HCC examples:

All examples refer to the following HACNET statements:

```
HACNET HID=H1,LU=A5OB6200,...
HACNET HID=H2,LU=A5OB6201,...
```

The following displays referring to the LU can appear on the console:

### 1.) Status of the LU (ACB) of the local HCC when the local HCC is inactive:

```
D NET,E,ID=A5OB6200
IST097I DISPLAY ACCEPTED
IST075I NAME = A5OB6200, TYPE = APPL 332
IST486I STATUS= CONCT, DESIRED STATE= CONCT
IST977I MDLTAB=***NA*** ASLTAB=***NA***
IST861I MODETAB=MTOBIS USSTAB=***NA*** LOGTAB=***NA***
IST934I DLOGMOD=***NA*** USS LANGTAB=***NA***
IST597I CAPABILITY-PLU INHIBITED,SLU INHIBITED,SESSION LIMIT NONE
IST231I APPL MAJOR NODE = A05OBIS2
IST654I I/O TRACE = OFF, BUFFER TRACE = OFF
IST271I JOBNAME = ***NA***, STEPNAME = ***NA***, DSPNAME = ***NA***
IST171I ACTIVE SESSIONS = 0000000000, SESSION REQUESTS = 0000000000
IST172I NO SESSIONS EXIST
IST314I END
```

### 2.) Status of the LU (ACB) of the local HCC when the local HCC is active and the partner inactive

```
D NET,E,ID=A5OB6200
IST097I DISPLAY ACCEPTED
IST075I NAME = A5OB6200, TYPE = APPL 399
IST486I STATUS= ACTIV, DESIRED STATE= ACTIV
IST977I MDLTAB=***NA*** ASLTAB=***NA***
IST861I MODETAB=MTOBIS USSTAB=***NA*** LOGTAB=***NA***
IST934I DLOGMOD=***NA*** USS LANGTAB=***NA***
IST597I CAPABILITY-PLU ENABLED  ,SLU ENABLED  ,SESSION LIMIT NONE
IST231I APPL MAJOR NODE = A05OBIS2
IST654I I/O TRACE = OFF, BUFFER TRACE = OFF
IST271I JOBNAME = HACWD601, STEPNAME = WD, DSPNAME = IST111B0
IST171I ACTIVE SESSIONS = 0000000000, SESSION REQUESTS = 0000000000
IST172I NO SESSIONS EXIST
IST314I END
```

### 3.) Status of the LU (ACB) of the local HCC when the local HCC and the partner are active

```
D NET,E,ID=A5OB6200
IST097I DISPLAY ACCEPTED
IST075I NAME = A5OB6200, TYPE = APPL 593
IST486I STATUS= ACT/S, DESIRED STATE= ACTIV
IST977I MDLTAB=***NA*** ASLTAB=***NA***
IST861I MODETAB=MTOBIS USSTAB=***NA*** LOGTAB=***NA***
IST934I DLOGMOD=***NA*** USS LANGTAB=***NA***
IST597I CAPABILITY-PLU ENABLED  ,SLU ENABLED  ,SESSION LIMIT NONE
IST231I APPL MAJOR NODE = A05OBIS2
IST654I I/O TRACE = OFF, BUFFER TRACE = OFF
IST271I JOBNAME = HACWD601, STEPNAME = WD, DSPNAME = IST111B0
IST171I ACTIVE SESSIONS = 0000000003, SESSION REQUESTS = 0000000000
IST206I SESSIONS:
IST634I NAME     STATUS         SID          SEND RECV VR TP NETID
IST635I A5OB6201 ACTIV-S    C88363FE61B666AF 0005 0000  0  0 NETICR
IST635I A5OB6201 ACTIV-P    C88363FE61B666AB 0000 0004  0  0 NETICR
IST635I A5OB6201 ACTIV-P    C88363FE61B666A7 0003 0002  0  0 NETICR
IST314I END
```

### 4.) A further important display appears as from VTAM 3.4:

```
D NET,CONVID,ID=A5OB6200
IST097I DISPLAY ACCEPTED
IST1040I CONVERSATION(S) FOUND FOR A5OB6200 703
IST1007I PARTNER = A5OB6201, LOGMODE = OBISL62P
IST1008I CONVID = 0300000E, STATUS = SEND, ETIME = 1
IST1009I SID = C88363FE61B666AF
IST924I -------------------------------------------------------------
IST1007I PARTNER = A5OB6201, LOGMODE = OBISL62P
IST1008I CONVID = 03000007, STATUS = RECEIVE, ETIME = 1
IST1009I SID = C88363FE61B666AB
IST924I -------------------------------------------------------------
IST314I END
```

This display shows ALL conversations for a LU (parameter ID=). This can be compared to the HCC command **D SESSION**.

The SENDCNV and RECVCNV fields must contain values to indicate that everything is in order.

```
HAC036I *-------------------------- SESSION  DISPLAY ---------------------*
HAC036I * STATE OF ZHC05400 : ACTIVE
HAC036I * HID MID TYP LUNAME   SENDCNV  RECVCNV  SENDMODE SENDSESS RECVSESS
HAC036I * H1  M1  LCL A5OB6200
HAC036I * H2  00  REM A5OB6201 0300000E 03000007 OBISL62P 61B666AF 61B666AB
HAC036I * H3  00  REM A3OB6200
HAC036I * H4  00  REM A3OB6201
HAC036I * H5  00  REM A3OB6202
HAC036I *-------------------------- END OF DISPLAY ----------------------*
```

MODETAB examples:
- Source

```
COLS      1...!....10...!..16.........          .......................70.72
                   MTOBIS   MODETAB
          OBISL62S MODEENT LOGMODE=OBISL62S,                              *
                           TYPE=0,                                        *
                           FMPROF=X'13',                  BYTE 1          *
                           TSPROF=X'07',                  BYTE 2          *
                           PRIPROT=X'B0',                 BYTE 3          *
                           SECPROT=X'B0',                 BYTE 4          *
                           COMPROT=X'50B1',               BYTES 5 AND 6   *
                           SSNDPAC=X'00',                 BYTES 7         *
                           SRCVPAC=X'00',                 BYTES 8         *
                           PSNDPAC=X'00',                 BYTES 9         *
                           RUSIZES=X'85C5',               BYTES 10 and 11*
                           PSERVIC=X'0602000000000000000002C00' 13 - 24
          OBISL62P MODEENT LOGMODE=OBISL62P,                              *
                           TYPE=0,                                        *
                           FMPROF=X'13',                  BYTE 1          *
                           TSPROF=X'07',                  BYTE 2          *
                           PRIPROT=X'B0',                 BYTE 3          *
                           SECPROT=X'B0',                 BYTE 4          *
                           COMPROT=X'50B1',               BYTES 5 AND 6   *
                           SSNDPAC=X'00',                 BYTES 7         *
                           SRCVPAC=X'00',                 BYTES 8         *
                           PSNDPAC=X'00',                 BYTES 9         *
                           RUSIZES=X'85C5',               BYTES 10 and 11*
                           PSERVIC=X'0602000000000000000023 00'   13-24
                   MODEEND
                   END
```

The parameter RUSIZES is arbitrary and has the following format:
RUSIZES=X'85C5' represents: 85 = Send-RU-size = 8x2**5 = 256
C5 = Recv-RU-size = 12x2**5 = 384

The lower value is always used for the new session when a different RUSIZE has been selected for the partner.

- Load module MTOBIS in VTAMLIB concatenation:

```
Offset   Data                                  Comment

000000   C0300002 00000000                     MODETAB-MACRO
000008   D6C2C9E2 D3F6F2E2                     MODEENT-MACRO(Mode:OBISL62S)
000010   001307B0 B050B100 0085C500 00060200  (0602 = LU6.2 Mode)
000020   00000000 0000002C 00000840 40404040  (2C = Single Session Mode)
000030   40404000 00000000
000038   D6C2C9E2 D3F6F2D7                     MODEENT-MACRO(Mode:OBISL62P)
000040   001307B0 B050B100 0085C500 00060200  (0602 = LU6.2 Mode)
000050   00000000 00000023 00000840 40404040  (23 = Parallel Session Mode)
000060   40404000 00000000
```

- Printout MTOBIS in hex-on format:

```
..Ø......MTOBIS  .......Ç
28000001DEDCCE4400000006
0000010043629200000008
-----------------------------------------------------------------------------
-------------------
Ø³.............................................................................
...................8F0000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000A1000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000
-----------------------------------------------------------------------------
-------------------Ø..566528408 ..l..
810FFFFFFFFF400922
01256652840803332F
-----------------------------------------------------------------------------
-------------------Ø.dØ..566896201 ..l..
818800FFFFFFFFF400922
04401056689620102132F
 ----------------------------------------------------------------------------
-------------------..........  ..Ç...Ç
00000000000040060006
D0000400600000080108
-----------------------------------------------------------------------------
-------------------ä.......OBISL62S...¢¢&£..eE................
.....OBISL62P...¢¢&£..eE................
C3000000DCCEDFFE010BB5B008C0000000000002000444444440000 0DCCEDFFD010BB5B008C
0000000000002000444
0002000062923622037000100550062000000000C008000000000000062923627037000100 55
0062000000003008000
-----------------------------------------------------------------------------
-------------------
```

HCC-AMU examples:

All examples refer to the following HACNET statements:

```
HACNET HID=H1,LU=A5OB6200,...
HACNET HID=A1,LU=TOBPL52,LU2=TOBPL53
```

The following displays referring to the LU can appear on the console:

### 1.) Status of the LU (ACB) of the local HCC when the AMU is active

```
D NET,E,ID=A5OB6200
IST097I DISPLAY ACCEPTED
IST075I NAME = A5OB6200, TYPE = APPL 207
IST486I STATUS= ACT/S, DESIRED STATE= ACTIV
IST977I MDLTAB=***NA*** ASLTAB=***NA***
IST861I MODETAB=MTOBIS USSTAB=***NA*** LOGTAB=***NA***
IST934I DLOGMOD=***NA*** USS LANGTAB=***NA***
IST597I CAPABILITY-PLU ENABLED  ,SLU ENABLED  ,SESSION LIMIT NONE
IST231I APPL MAJOR NODE = A05OBIS2
IST654I I/O TRACE = OFF, BUFFER TRACE = OFF
IST271I JOBNAME = ***NA***, STEPNAME = ***NA***, DSPNAME = ISTA4F2F
IST171I ACTIVE SESSIONS = 0000000002, SESSION REQUESTS = 0000000000
IST206I SESSIONS:
IST634I NAME     STATUS         SID          SEND RECV VR TP NETID
IST635I TOBPL52  ACTIV-S    C88363FE659FA7E1 001B 0001  4  0 NETICR
IST635I TOBPL53  ACTIV-S    D08363FE67A71271 0001 001B  4  0 NETICR
IST314I END
```

## 2.) Conversation display for the local HCC

```
D NET,CONVID,ID=A5OB6200
IST097I DISPLAY ACCEPTED
IST1040I CONVERSATION(S) FOUND FOR A5OB6200 291
IST1007I PARTNER = TOBPL52, LOGMODE = OBISL62S
IST1008I CONVID = 02000022, STATUS = SEND, ETIME = 8
IST1009I SID = C88363FE659FA7E1
IST924I -------------------------------------------------------------
IST1007I PARTNER = TOBPL53, LOGMODE = OBISL62S
IST1008I CONVID = 02000021, STATUS = RECEIVE, ETIME = 8
IST1009I SID = D08363FE67A71271
IST924I -------------------------------------------------------------
IST314I END
```

This display shows ALL conversations for a LU (parameter ID=). This is can be compared to the HCC command **D SESSION**.

```
D SESSION
*-------------------------- SESSION  DISPLAY -------------------------*
* STATE OF ZHC05400 : ACTIVE                                           *
* HID MID TYP LUNAME   SENDCNV  RECVCNV  SENDMODE SENDSESS RECVSESS    *
* H1  M1  LCL A5OB6200                                                 *
* A1  00  REM TOBPL52  02000022 02000021 OBISL62S 659FA7E1 67A71271    *
*-------------------------- END OF DISPLAY ---------------------------*
```

Only the LU used by the HCC to send to the AMU (HACNET HID=A1,LU=sendlu,...).is shown under LUNAME during a SINGLE SESSION connection to the AMU. The name of the LU used by the HCC to receive data from the AMU (HACNET HID=A1,LU2=recvlu,...) is not displayed.

### 3.) Status of the LU in the AMU during an active AMU-HCC connection

```
D NET,E,ID=TOBPL52
IST097I DISPLAY ACCEPTED
IST486I STATUS= ACT/S----Y, DESIRED STATE= ACTIV
IST977I MDLTAB=***NA*** ASLTAB=***NA***
IST861I MODETAB=***NA*** USSTAB=***NA*** LOGTAB=***NA***
IST934I DLOGMOD=***NA*** USS LANGTAB=***NA***
IST597I CAPABILITY-PLU ENABLED  ,SLU ENABLED  ,SESSION LIMIT NONE
IST231I CDRSC MAJOR NODE = ISTCDRDY
IST479I CDRM NAME = CIC03, VERIFY OWNER = NO
IST082I DEVTYPE = CDRSC
IST654I I/O TRACE = OFF, BUFFER TRACE = OFF
IST171I ACTIVE SESSIONS = 0000000001, SESSION REQUESTS = 0000000000
IST206I SESSIONS:
IST634I NAME     STATUS         SID          SEND RECV VR TP NETID
IST635I A5OB6200 ACTIV-P    C88363FE659FA7E1 0001 001B  4  0 NETICR
IST314I END
```

For VTAM < 3.4, the SEND/RECV counts in the previous display show whether the inquired LU has LU SEND or RECV status with the HCC. The LU (an AMU-LU) in this example has SEND=0001 RECV=001B, for example, it has sent one message to A5OB6200 and has received 27 messages from A5OB6200. Seen from the HCC, this is the LU being sent to (HACNET HID=A1,LU=TOBPL52,...).

## 12.2 VTAM TRACE creation example

The following shows how to create a VTAM trace:

- Precondition: HACPARM:

```
HACNET HID=H1,LU=A5OB6200,LM=OBISL62S
HACNET HID=A1,LU=TOBPL52,LU2=TOBPL53,LM=OBISL62S
```

- Method:

  + activate the VTAM Trace function for the desired LUs

```
F NET,TRACE,TYPE=BUF,ID=A5OB6200
F NET,TRACE,TYPE=BUF,ID=TOBPL52
F NET,TRACE,TYPE=BUF,ID=TOBPL53
```

  + Start a prepared GTF start procedure:

```
S HACTR601.TRC
```

Dataset: SYS.OBIS.PROCLIB(HACTR601)

```
//GTFOBIS PROC
//IEFPROC EXEC PGM=AHLGTF,PARM='MODE=EXT,DEBUG=NO,TIME=YES',
//  TIME=1440,REGION=2880K
//IEFRDER DD   DSNAME=OBISD.SNA.TRACE,DISP=OLD
//SYSLIB  DD   DSNAME=SYS1.OBIS.PROCLIB(VTAMTRC),DISP=SHR
```

Dataset: SYS.OBIS.PROCLIB(VTAMTRC)

```
TRACE=RNIO,USR
GO
```

- Start a HCC
  + Start the AMU
  + Await all significant STARTUP messages from HCC /AMU

  + Terminate GTF trace:

```
P TRC
```

  + Terminate the activated VTAM TRACE function

```
F NET,NOTRACE,TYPE=BUF,ID=A5OB6200
F NET,NOTRACE,TYPE=BUF,ID=TOBPL52
F NET,NOTRACE,TYPE=BUF,ID=TOBPL53
```

- Evaluate the TRACE:

In this example, the TRACE data are recorded in the file **OBISD.SNA.TRACE.** This file can be made "readable" with a TRACE formatting function (for example, ACFTAP).

```
//OBI1WD1V JOB 'E001135F00001',OBIS,CLASS=5,MSGCLASS=T
//ACFTAP  EXEC PGM=ACFTAP,REGION=1024K
//STEPLIB  DD  DSN=NCP.SSP360.SSPLIB,DISP=SHR
//SORTIN   DD  DSN=OBI1WD1.SSP.SORTIN,UNIT=SYSDA,SPACE=(CYL,(10,5)),
//             DISP=(NEW,DELETE),DCB=(RECFM=F,LRECL=364,BLKSIZE=364)
//SORTOUT  DD  DSN=OBI1WD1.SSP.SORTOUT,UNIT=SYSDA,SPACE=(CYL,(10,5))
//             DISP=(NEW,DELETE),DCB=(RECFM=F,LRECL=364,BLKSIZE=364)
//SORTWK01 DD  DSN=&&SORTWK01,SPACE=(CYL,(10,5),,CONTIG),UNIT=SYSDA,
//             DISP=(NEW,DELETE)
//SYSSDPRT DD  SYSOUT=*
//SYSSSPRT DD  SYSOUT=*
//SYSNEPRT DD  SYSOUT=*
//SYSDTPRT DD  SYSOUT=*
//SYSLUPRT DD  SYSOUT=*
//SYSXPRT  DD  SYSOUT=*
//SYSPRINT DD  SYSOUT=*
//SYSOUT   DD  SYSOUT=*
//SYSTRACE DD  DISP=SHR,DSN=OBISD.SNA.TRACE
//SYSIN    DD  *
INPUT=ALL
STIME=00.00.00
ETIME=23.59.00
SDPRT=YES
SSPRT=YES
DTPRT=YES
NEPRT=YES
SUMMARY=YES
LONGPIU=YES
PRINT=YES
DUMP=YES
UNIT=DISK
WRAP=YES
GO
QUIT
/*
//* INPUT=LINE        (ALL_BUFFER_PIU_LINE_RNIO_IO_SCAN) DEFAULT=LINE
//*                    VIT (VTAM INTERNAL TRACE ) SSP V2R2
//* SDPRT=YES         SNA DETAIL       (NO_YES)          DEFAULT=NO
//* SSPRT=YES         SNA SUMMARY      (NO_YES)          DEFAULT=YES
//* DTPRT=YES         DATA TRAFFIC     (NO_YES)          DEFAULT=NO
//* NEPRT=NO          NETWORK ERROR    (NO_YES)          DEFAULT=YES
//* SUMMARY=NO        INPUT RECORDS    (NO_YES_ALL_EVERY)DEFAULT=NO
//* DUMP=NO           STANDARD FORMAT  (NO_YES)          DEFAULT=NO
//* PRINT=YES         TH,RH,RU         (NO_YES)          DEFAULT=YES
//* STIME=   ETIME=   SELECT TIME      (HH:MM:SS)
//* START=1  END=9999 SELECT RECORDS   (COUNT)
```

## 12.3 VTAM trace example

- Actions executed with console displays:

```
F NET,TRACE,TYPE=BUF,ID=A5OB6200
IST097I MODIFY ACCEPTED
IST513I TRACE INITIATED FOR NODE A5OB6200

F NET,TRACE,TYPE=BUF,ID=TOBPL52
IST097I MODIFY ACCEPTED
IST511I TRACE REQUEST FAILED - TOBPL52 INVALID

F NET,TRACE,TYPE=BUF,ID=TOBPL53
IST097I MODIFY ACCEPTED
IST511I TRACE REQUEST FAILED - TOBPL53 INVALID
```

(These LUs are defined in a different system and can first be displayed after definition as dynamic resources for this system)

```
S HACTR601.TRC
ICH70001I ICRUNDEF LAST ACCESS AT 10:44:50 ON FRIDAY, AUGUST 27, 1993
$HASP100 HACTR601 ON STCINRDR
IEF693I PROCEDURE HACTR601 IS ASSIGNED TO USER ICRUNDEF, GROUP IN$SP
$HASP373 HACTR601 STARTED
IEF403I HACTR601 - STARTED
AHL121I  TRACE OPTION INPUT INDICATED FROM MEMBER VTAMTRC  OF PDS
SYS1.OBIS.PROCLIB
TRACE=RNIO,USR
AHL103I  TRACE OPTIONS SELECTED --USR,RNIO
71 AHL125A  RESPECIFY TRACE OPTIONS OR REPLY U
R 71,U
IEE600I REPLY TO 71 IS;U
U
AHL906I THE OUTPUT BLOCK SIZE OF    13200 WILL BE USED FOR OUTPUT 537
        DATA SETS:
          OBISD.SNA.TRACE
AHL080I GTF STORAGE USED FOR GTF DATA: 538
        GTFBLOCK STORAGE       51K BYTES (BLOK=        40K)
        PRIVATE STORAGE      1031K BYTES
        SADMP HISTORY          51K BYTES (SADMP=       40K)
        SDUMP HISTORY          51K BYTES (SDUMP=       40K)
        ABEND DUMP DATA         0K BYTES (ABDUMP=       0K)
        AHL031I GTF INITIALIZATION COMPLETE
```

HCC-JOBLOG:

```
J E S 2  J O B  L O G   -- S Y S T E M  S 5 0 0 --  N O D E  E S A T N J E
0
IEF693I PROCEDURE HACWD60S IS ASSIGNED TO USER ICRUNDEF, GROUP IN$SP
$HASP373 HACWD60S STARTED
IEF403I HACWD60S - STARTED
HAC021I HACC START: VERSION=V2R03MX1, SMFCPUID=S500
HAC036I YOU SELECTED THE FOLLOWING STARTPARMS:
HAC036I SYS=PRI,FMTU=N,SIM=N,SSI=N,HOT=N,HID=H1,M1=H1,A1ID=A1,A2ID=A2,SUP=N
HAC204I PARM INPUT: MEMBER=HACPARMS,DSN=OBI1WD1.APPC.HACCCNTL
HAC415I Start of SEND subtask for A1 completed
HAC117I TOTAL NO. ARCHIVE TAPES 06010
HAC314I NO OF "VIRTUAL" TAPES   00000
HAC118I NO. EJECTED CARTRIDGES  00250
HAC166I TAPES TO BE EJECTED     00000
HAC119I NO. CLEAN-CARTRIDGES    00010
HAC246I TOTAL NO. SCRATCH TAPES 00098
HAC247I AVAILABLE SCRATCH TAPES 00098
HAC000I ========================================================
HAC016I ===          HACC READY FOR COMMUNICATION            ===
HAC000I ========================================================
HAC046I MESSAGE TRACE ON
HAC036I LOGSTART
HAC036I VARY ACT,HID=A1
HAC082I CLEANING CARTRIDGE CL0001 IS NOW IN USE FOR SYS=1,ROB=1
HAC082I CLEANING CARTRIDGE CL0002 IS NOW IN USE FOR SYS=1,ROB=2
HAC243I MESSAGE LOG IS ACTIVE, DSN=OBI1WD1.ZHCLG200
HAC421I APPC Error: FUNCTION SEND CON RC: 00  R0: 11 RCPRI:0004 RCSEC: 000A
HAC443I APPC ERROR: ALLOCATION_ERROR
HAC443I             TP NOT AVAILABLE,RETRY
```

This return code indicates that the PC on which the AMU runs is active but the AMU itself has not been started.

```
HAC036I CX SOL,A1,H1
HAC036I <A1H1H03,H1,V2R03MX1,25.06.93,H1                                  >
HAC407I There is no SEND conversation for A1 and CMD:
        <A1H1H03,H1,V2R03MX1,02.06.93,H1
```

The AMU has now been started and has received an ALLOCATE request from the HCC

```
HAC445I Conversation request received from A1; Partner_Lu_Name is TOBPL53
HAC415I Start of RECV subtask for A1 completed
HAC036I VARY ACT,HID=A1
HAC036I CX SOL,A1,H1
HAC036I <A1H1H03,H1,V2R03MX1,02.06.93,H1                                  >
HAC036I <H1A1H04,H1,V2R03MX1,02.06.93,H1                 27/110344,  ,    >
HAC036I CX SOL,A1,H1
HAC036I <A1H1H03,H1,V2R03MX1,02.06.93,H1                                  >
HAC036I <H1A1H04,H1,V2R03MX1,02.06.93,H1                 27/110355,  .    >
```

```
P TRC
AHL006I GTF ACKNOWLEDGES STOP COMMAND
AHL904I THE FOLLOWING TRACE DATASETS CONTAIN TRACE DATA : 775
          OBISD.SNA.TRACE
          JOBNAME  PROCSTEP STEPNAME PROGRAMM COND.CDE EXCPS
ICRACTRT HACTR601 TRC      IEFPROC  AHLGTF     0000      47
IEF404I HACTR601 - ENDED
$HASP395 HACTR601 ENDED
IEA989I SLIP TRAP ID=033E MATCHED
$HASP250 HACTR601 IS PURGED


F NET,NOTRACE,TYPE=BUF,ID=A5OB6200
IST097I MODIFY ACCEPTED
IST512I TRACE TERMINATED FOR NODE = A5OB6200
```

# 13 TRACE

## 13.1 Function:

A TRACE table in ZHC00100 is initialized to x'00' during STARTUP. All programs to be traced must include the HACTRE macro (or this must be present at all significant points). Creating so-called TRACEPOINTS.

Commands activate and deactivate these TRACEPOINTS (SET TRACE,ON/OFF commands).

Various TRACE LEVELS are available:

- Level 1: Run trace only showing names allocated by CSECT in HACTRE.
- Level 2: Register trace showing all register contents.
- Level 3: Register detail trace showing the storage area pointed to by a register.

Commands start and stop the trace (SET TRACE,START/STOP)

## 13.2 Command set:

```
SET TRACE,ON
     [,MOD=modulename]
     [,NUM=number]
     [,LVL={1/2/3}]
     [,REGS=regnum[,O=offset]]
```

**modulename** (Optional, see example for exceptions)
Name of the CSECT to be traced.

**number** (Optional, see example for exceptions)
Internal number within a CSECT allocated by the HACTRE macro in the module.

**LVL** (Optional, default : LVL=1)
- is a run trace
- register trace
- register detail trace

**REGS** (Optional, no default)
REGS specifies the registers to be used during a LVL=3 trace. Regnum is a register number (0-15), offset is an offset to be added to the value in regnum. The length of the data to be displayed can be specified in the HACTRE macro.

SET TRACE,ON,... either makes a new entry in the Tracepoint table or modifies an existing entry.

```
SET TRACE,OFF
     [,MOD=modulename]
     [,NUM=number]
     [,LVL={1/2/3}]
```

**modulename** (Optional, see example for exceptions)
Name of the CSECT to be traced.

**number** (Optional, see example for exceptions)
Internal number within a CSECT allocated by the HACTRE macro in the module.

**LVL** (Optional, default : LVL=1)
- is a run trace
- register trace
- register detail trace

SET TRACE,OFF,.... modifies an entry in the Tracepoint table (deactivate LVL for **one** entry (MOD=. +NUM=.), for **all entries of a CSECT** (MOD=), for **all entries with a number** (NUM=), or for **all** entries( neither MOD= nor NUM=)).

```
SET TRACE,START
SET TRACE,STOP
```

SET TRACE START/STOP start or stop the trace previously set.

```
D TRACE [,MOD=modulename][,NUM=number]
```

DISPLAY TRACE displays the following:

| | |
|---|---|
| MOD= and NUM= not specified: | All entries in the Tracepoint table |
| Only MOD= specified: | All entries for the CSECT module name |
| Only NUM= specified: | All entries with number |
| MOD= and NUM= specified: | A specific entry |

## 13.3 Calling Trace in a program:

A trace is requested by setting a trace point in a program with the HACTRE macro. Attention: HACTRE can first be used after addressing the HACICE, (see example)

Syntax for HACTRE:

```
HACTRE NUM=nnn[,NAME=mmmmmmmmm]
```

| | |
|---|---|
| **nnn** | Max. 3 digit number unique within the CSECT |
| **mmmmmmmmm** | Max. 8 character arbitrary trace point name |

Example:

Two points in ZHC04500 are to be traced, at the start and end of CSECT ZHC04500.

```
START:
ZHC04500 CSECT
ZHC04500 AMODE 31
ZHC04500 RMODE 24
         .
         .
         .
         USING HACICE,R5
         USING HACICE,R6
         LR R6,R1
         L  R5,ICEPTR
         HACTRE NUM=1,NAME=H45STRT
         .
         .
END:
         .
         .
         HACTRE NUM=2,NAME=H45END
```

ZHC04500 must now be compiled and linked. The TRACEPOINTS in ZHC04500 can now be activated after a module refresh or HCC restart. Use the following commands:

```
SET TRACE,ON,MOD=ZHC04500,NUM=1
SET TRACE,ON,MOD=ZHC04500,NUM=2
```

These commands create two entries in the Trace table. The D TRACE,.. command can be used to display these entries.

Enter the following command to start the Trace function:

```
SET TRACE,START
```

The command creates the message

```
HAC473I The selected Traces have been started
```

The desired trace data are now created and displayed at all the activated trace.

For LVL=1:

```
HAC470I Trace entered for ZHC04500 at H45STRT
```

For LVL=2:

```
HAC471I Register contents:
HAC036I R0: ........ R1: ........ R2: ........ R3: ........
HAC036I R4: ........ R5: ........ R6: ........ R7: ........
HAC036I R8: ........ R9: ........ RA: ........ RB: ........
HAC036I RC: ........ RD: ........ RE: ........ RF: ........
```

For LVL=3:

```
HAC472I Data at Register R6 Offset: 000
HAC036I C8C9C3E3 00001000 00008000 00141A38 HICT............*
HAC036I F0F0F0F0 F0F3F0F0 F0F0F0F0 F0F5F0F0 0000030000000500*
```

# 14 MESSAGE PROCESSING

## 14.1 Messages

The message programs for German language messages are no longer required. The messages are entered sequentially in the modules ZHC03000 -ZHC03400 in various languages with the same message number, whereby the first entry **must** be the English message (default).

The programs ZHCG3000 - ZHCG3400 are no longer required, the old versions are in OBISD.V2R4M0.HACCSRC.SAVE.

## 14.2 Changed macros

**HACMGN**

New parameter : [,L=E,G,A]
Optional parameter, specifies the language in which the message is defined.

L=E English (default)
L=G German
L=A This message is the same for all languages.

New macro entries:

Language code:

```
&I.MGLC   DC   AL1(0)
&I.MXCALL EQU  X'FF' - ALL (ONLY ONE ENTRY)
&I.MXCENG EQU  X'01' - ENGLISH
&I.MXCGER EQU  X'02' - GERMAN
&I.MXCFRE EQU  X'04' - FRENCH
```

Parameter: `&L=E` (default: English)

Message number: `&I.MGNO DC  AL2(0)`

# 15 MSG Command

MSG serves to test MVS messages to be processed in MVS MESSAGE SCAN (ZHC01800).

```
MSG message
```

**`message`** Any valid MVS message that can be processed by HCC.

Example:

```
MSG IEF233E M 92A,000001
```

Modules affected:

New module: H015115
Changed modules: ZHC01500, ZHCG1400, ZHC01400

# 16 WAIT COMMAND

WAIT is an extension (ABBASEND feature) to generate a ‘WAIT’ before the next command is processed.

```
WAIT,hhmmssth
```

**hhmmssth** Defines the time to be waited

| | | |
|---|---|---|
| **hh** | = | Hours |
| **mm** | = | Minutes |
| **ss** | = | Seconds |
| **t** | = | Tenths of seconds |
| **h** | = | Hundredths of seconds |

Modules affected:

| | |
|---|---|
| New modules: | - |
| Changed module: | ZHC10300 |

# 17 REFRESH MODULE COMMAND

REFRESH MODUL loads a fresh copy of a HCC module without HCC Stop. Further parameters for this command are described in the COMMAND REFERENCE for HCC/MVS 3.0.

```
REFRESH module
```

**module** HCC module

Example:

```
REFRESH ZHC03000
```

# INDEX


3278 BUFFER CONFIGURATION..... 4-1
AML
COMMAND OVERVIEW ..... 3-3
ERROR MESSAGES.....5-2
AML COMMANDS
ACOM..... 3-5
AOFF..... 3-6
AUTO..... 3-7
BOFF..... 3-8
CLU..... 3-10
DL ..... 3-11
EJ ..... 3-13
EJT ..... 3-15, 3-17, 3-26, 3-38
INC..... 3-18
INV..... 3-20
KE ..... 3-22
KEC..... 3-24
LMV..... 3-27
MAN ..... 3-28
MO3-12, 3-14, 3-16, 3-19, 3-21, 3-23, 3-25, 3-29, 3-32, 3-37, 3-40, 3-42
MV ..... 3-31
ROPO ..... 3-33
ROSA ..... 3-34
ROSO ..... 3-35
SCH..... 3-36
ULC..... 3-39
ULV ..... 3-41
UPC..... 3-43
UPV..... 3-44
VI ..... 3-45
VIC..... 3-46
VICC .....3-47, 3-48
AMU..... 3-1
AR ..... 3-1
ARCHIVE
HACASR Macro ..... 8-2
RECORD FORMAT ..... 8-1
COMMUNICATION
AML COMMANDS..... 3-1
COMMAND SYNTAX..... 3-1
HCC-AML ..... 2-2
HCC-HCC INTERCOMMUNICATION ..... 2-3
HCC-HCC INTERCOMMUNICATION (LU 6.2) 2-6
HCC-TOKEN RING ..... 2-4
RECORD FORMAT ..... 8-6
TOKEN RING..... 2-1
COMPILE AND LINK .....10-1
D SESSION.....12-6
D TRACE.....13-3
HACMGN .....14-1
HCC-JOBLOG .....12-11
HEADER
DISPLAY.....11-1
INITIALIZATION .....11-1
LU (ACB).....12-2
LU DISPLAYS .....12-1
MACRO SUMMARY.....7-5
MACROS
HACASR .....8-2
HACLGD .....8-5
HACRMF.....8-6
MESSAGE LOG
RECORD FORMAT .....8-5
MESSAGE PROCESSING.....14-1
MESSAGES
AML.....5-2
AML.....5-1
AML asynchronous.....5-1
AML synchronous.....5-1
ASYNCHRONOUS .....5-1
HACC .....8-7
HCC .....9-1
MVS.....6-1
SYNCHRONOUS.....5-1
MODETAB .....12-3
MODULE SUMMARY .....7-2
MSG .....15-1
MTOBIS.....12-4
RECORD FORMATS
ARCHIVE.....8-1
RECORD FORMATS
EXCP .....8-6
MESSAGE LOG.....8-5
REFRESH MODULE .....17-1
RUSIZES .....12-3
SET TRACE..... 13-2, 13-3
STATUS VOLSER .....3-5
TRACE.....13-1
TRACE DATA.....12-9
VOLSER STATUS .....3-5
VTAM TRACE ..... 12-8, 12-10
WAIT .....16-1