Chapter 1

Preface

Copyright

GOOSY Copyright

The GOOSY software package has been developed at GSI for scientific applications. Any distribution or usage of GOOSY without permission of GSI is not allowed. To get the permission, please contact at GSI Mathias Richter (tel. 2394 or E-Mail "M.Richter@gsi.de") or Hans-Georg Essel (tel. 2491 or E-Mail "H.Essel@gsi.de").

MBS Copyright

The MBS software package has been developed at GSI for scientific applications. Any distribution or usage of MBS without permission of GSI is not allowed. To get the permission, please contact at GSI Mathias Richter (tel. 2394 or E-Mail "M.Richter@gsi.de") or Hans-Georg Essel (tel. 2491 or E-Mail "H.Essel@gsi.de"), or Nikolaus Kurz (tel. 2979 or E-Mail "N.Kurz@gsi.de").

Conventions used in this Document

Examples in this manual show both system output (prompts, messages, and displays) and user input, which are all written in typewriter style. Names and keywords are also in typewriter style. Items to be replaced by actual values are enclosed in <>.

The change bars mark changes between MBS and SBS.

Registered Trademarks are not explicitly noted.
1.1 MBS Authors and Advisory Service

The authors of MBS and their main fields for advisory services are (phone GSI: + 6159 71 -):

**R. Barth**  Fastbus SMI library (Tel. 2554)

**Y. Du**  Event routines, DECunix port, AIX port (1995-1996)

**H.G. Essel**  Multitasking, CI, message logger, ESONE, GOOSY connections (Tel. 2491)

**R. Fritzsche**  TCP library.

**H. Göringer**  Event server, PAW connections (Tel. 2553)

**J. Hoffmann**  CVC hardware, trigger (Tel. 2494)

**F. Humbert**  Eson library (1996)

**N. Kurz**  Lynx, setup, readout, collector, multi-branch systems (Tel. 2979)


**W. Ott**  Message logger, transport, stream server, taping (Tel. 2979)

**D. Schall**  CVC hardware, trigger (left GSI in 1996)

The authors thank Margareta Hellström and Piotr Koczon for useful hints and for reviewing this manual resulting in substantial enhancements. Further suggestions to enhance the manual are welcome. The manuals are accessible also through WWW:

http://www-gsi-vms.gsi.de/img/doc or

http://www-gsi-vms.gsi.de/daq/home

H.Essel@gsi.de.
Chapter 2

MBS Keyword Summary
Keywords

In the following the MBS command keywords are listed with their occurrence in the commands.

ACCESS

REMOTE ACCESS

ACQUISITION

START ACQUISITION
STOP ACQUISITION

ARECEIVER

SET ARECEIVER FLUSHTIME
START ARECEIVER
STOP ARECEIVER

ATTACH

ATTACH BASE

BASE

ATTACH BASE
CREATE BASE
DELETE BASE
DETACH BASE
DUMP BASE
RESTORE BASE
CAMAC

CAMAC CNAF
CAMAC FILE

CHANNEL

SET HISTOGRAM CHANNEL

CLEAR

CLEAR DAQ.STATUS COUNTER
CLEAR DAQ.STATUS PROCTAB
CLEAR DAQ.STATUS STATUS
CLEAR HISTOGRAM
CLEAR PIPES
CLEAR POLYGON
CLEAR TRIG.MOD
CLEAR WINDOW

CLIENT

DISCONNECT CLIENT STREAM_SERV
DISCONNECT CLIENT TRANSPORT

CLOSE

CLOSE FILE

CNAF

CAMAC CNAF
CNAF
COMMANDS

DEFINE COMMANDS

COMMENT

COMMENT

CONDITION

CREATE CONDITION INCLUDE
CREATE CONDITION POLYGON
CREATE CONDITION WINDOW

CONNECT

CONNECT DISPATCHER
CONNECT RFIO
CONNECT TRANSPORT

COUNTER

CLEAR DAQ, STATUS COUNTER

CREATE

CREATE BASE
CREATE CONDITION INCLUDE
CREATE CONDITION POLYGON
CREATE CONDITION WINDOW
CREATE FILES
CREATE HISTOGRAM
CREATE POLYGON
CREATE WINDOW
CVC_CAM_IRQ

DISABLE CVC_CAM_IRQ
ENABLE CVC_CAM_IRQ

CVC_IRQ_MASK

DABC

ENABLE DABC

DAQ_STATUS

CLEAR DAQ_STATUS COUNTER
CLEAR DAQ_STATUS PROCTAB
CLEAR DAQ_STATUS STATUS

DEFINE

DEFINE COMMANDS

DELAYED_EB

DISABLE DELAYED_EB
ENABLE DELAYED_EB

DELETE

DELETE BASE
DELETE HISTOGRAM

DEL_EB_COL
DISABLE DEL_EB_COL
ENABLE DEL_EB_COL

DETACH

DETACH BASE

DISABLE

DISABLE CVC_CAM_IRQ
DISABLE DELAYED_EB
DISABLE DEL_EB_COL
DISABLE EVENT_COPY
DISABLE HISTOGRAM
DISABLE RECEIVE_DATA
DISABLE TCP
DISABLE TRIG_MOD

DISCONNECT

DISCONNECT CLIENT_STREAM_SERV
DISCONNECT CLIENT_TRANSPORT
DISCONNECT DISPATCHER
DISCONNECT RFIO
DISCONNECT TRANSPORT

DISMOUNT

DISMOUNT_TAPE

DISPATCHER

CONNECT DISPATCHER
DISCONNECT DISPATCHER
SET DISPATCHER
SET VERBOSE DISPATCHER
DR_FLUSH_TIME

SET DR_FLUSH_TIME

DUMP

DUMP BASE
DUMP HISTOGRAM
DUMP WINDOW

ENABLE

ENABLE CVC_CAM_IRQ
ENABLE DABC
ENABLE DELAYED_EB
ENABLE DEL_EB_COL
ENABLE EVENT_COPY
ENABLE HISTOGRAM
ENABLE IRQ
ENABLE RECEIVE_DATA
ENABLE TCP
ENABLE TRIG_MOD

ENVIRONMENT

ERROR

ON ERROR

ESONE_SERV

SET MAXCLIENTS ESONE_SERV
SET PRINT ESONE_SERV
SET VERBOSE ESONE_SERV
EVENT

TYPE EVENT

EVENT_COPY

DISABLE EVENT_COPY
ENABLE EVENT_COPY

EVENT_SERV

SET EVENT_SERV
SET MAXCLIENTS EVENT_SERV
SET VERBOSE EVENT_SERV
START EVENT_SERV

FILE

CAMAC FILE
CLOSE FILE
OPEN FILE

FILEHEADER

SET FILEHEADER

FILES

CREATE FILES

FLUSHTIME

SET ARECEIVER FLUSHTIME
SET FLUSHTIME
SET RIRECEIVER FLUSHTIME
SET TORECEIVER FLUSHTIME
GLOBAL

SET VERBOSE GLOBAL

HELP

HELP

HISTOGRAM

CLEAR HISTOGRAM
CREATE HISTOGRAM
DELETE HISTOGRAM
DISABLE HISTOGRAM
DUMP HISTOGRAM
ENABLE HISTOGRAM
PROTECT HISTOGRAM
SET HISTOGRAM CHANNEL
SET HISTOGRAM TEXT
SET VERBOSE HISTOGRAM

INCLUDE

CREATE CONDITION INCLUDE

INITIALIZE

INITIALIZE TAPE

INPUT

IRQ

ENABLE IRQ
LOAD

LOAD ML_SETUP
LOAD MO_SETUP
LOAD READOUT
LOAD SETUP
LOAD SLAVE_READOUT

MAXCLIENTS

SET MAXCLIENTS ESONE_SERV
SET MAXCLIENTS EVENT_SERV

MESSAGE

START MESSAGE

ML_SETUP

LOAD ML_SETUP

MOUNT

MOUNT TAPE

MO_SETUP

LOAD MO_SETUP

NEWS

NEWS

NODES
ON

ON ERROR

OPEN

OPEN FILE

OUTPUT

START OUTPUT
STOP OUTPUT

PIPES

CLEAR PIPES

POLYGON

CLEAR POLYGON
CREATE CONDITION POLYGON
CREATE POLYGON
RESTORE POLYGON

PRINT

SET PRINT ESONE_SERV

PROCTAB

CLEAR DAQ_STATUS PROCTAB

PROMPT

SET VERBOSE PROMPT
PROTECT

  PROTECT HISTOGRAM

PSHELL

  PSHELL

RATE

READOUT

  LOAD READOUT

RECEIVE_DATA

  DISABLE RECEIVE_DATA
  ENABLE RECEIVE_DATA

REMOTE

  REMOTE ACCESS
  REMOTE RESET
  REMOTE SHOW

REMOVE

  REMOVE SEGMENTS

RESET

  REMOTE RESET
  RESET
RESTORE

RESTORE BASE
RESTORE POLYGON
RESTORE WINDOW

RFIO

CONNECT RFIO
DISCONNECT RFIO

RIRECEIVER

SET RIRECEIVER FLUSHTIME
START RIRECEIVER
STOP RIRECEIVER

SEGMENTS

REMOVE SEGMENTS

SET

SET ARECEIVER FLUSHTIME
SET DISPATCHER
SET DR_FLUSHTIME
SET EVENT_SERV
SET FILEHEADER
SET FLUSHTIME
SET HISTOGRAM CHANNEL
SET HISTOGRAM TEXT
SET MAXCLIENTS_ESONE_SERV
SET MAXCLIENTS_EVENT_SERV
SET PRINT_ESONE_SERV
SET RIRECEIVER_FLUSHTIME
SET STREAM_SERV
SET TASK
SET TORECEIVER FLUSHTIME
SET TRIG_MOD
SET USER VALUE
SET VERBOSE DISPATCHER
SET VERBOSE ESONE_SERV
SET VERBOSE EVENT_SERV
SET VERBOSE GLOBAL
SET VERBOSE HISTOGRAM
SET VERBOSE PROMPT
SET WINDOW
SET XDISPLAY

SETUP

LOAD SETUP

SHOW

REMOTE SHOW
SHOW

SLAVE_READOUT

LOAD SLAVE_READOUT

START

START ACQUISITION
START ARECEIVER
START EVENT_SERV
START MESSAGE
START OUTPUT
START RIRECEIVER
START TASK

STATUS
CLEAR DAQ, STATUS, STATUS

STOP

STOP ACQUISITION
STOP ARECEIVER
STOP OUTPUT
STOP RIRECEIVER
STOP TASK

STREAM

TYPE STREAM

STREAM_SERV

DISCONNECT CLIENT STREAM_SERV
SET STREAM_SERV

TAPE

DISMOUNT TAPE
INITIALIZE TAPE
MOUNT TAPE

TASK

SET TASK
START TASK
STOP TASK

TCP

DISABLE TCP
ENABLE TCP
TEXT

SET HISTOGRAM TEXT

TORECEIVER

SET TORECEIVER FLUSHTIME

TRANSPORT

CONNECT TRANSPORT
DISCONNECT CLIENT TRANSPORT
DISCONNECT TRANSPORT

TRIG_MOD

CLEAR TRIG_MOD
DISABLE TRIG_MOD
ENABLE TRIG_MOD
SET TRIG_MOD

TYPE

TYPE EVENT
TYPE STREAM

USER

SET USER VALUE

VALUE

SET USER VALUE
VERBOSE

SET VERBOSE DISPATCHER
SET VERBOSE ESONE_SERV
SET VERBOSE EVENT_SERV
SET VERBOSE GLOBAL
SET VERBOSE HISTOGRAM
SET VERBOSE PROMPT

VOID

VOID

VRA16D16

VRA16D16

VRA24D16

VRA24D16

VRA24D32

VRA24D32

VRA32D16

VRA32D16

VRA32D32

VRA32D32
VWA16D16

VWA16D16

VWA24D16

VWA24D16

VWA24D32

VWA24D32

VWA32D16

VWA32D16

VWA32D32

VWA32D32

WINDOW

CLEAR WINDOW
CREATE CONDITION WINDOW
CREATE WINDOW
DUMP WINDOW
RESTORE WINDOW
SET WINDOW

XDISPLAY

SET XDISPLAY
XSHELL

XSHELL
Chapter 3

MBS Command Summary

ATTACH BASE

name

(m_histogram) (m_histogram) Attaches existing histogram data base.

CAMAC CNAF


c n a f [d r] -LOG -NOPRINT

(m_esone_serv) (m_esone_serv) Executes local CAMAC cnaf.

CAMAC FILE

filename -LOG -NOPRINT

(m_esone_serv) (m_esone_serv) Executes local CAMAC cnaf.

CLEAR DAQ.STATUS COUNTER

–

(m_util) (m_util) Clears status counters in the global daq status segment

CLEAR DAQ.STATUS PROCTAB

–

(m_util) (m_util) Clears daq process table in the global daq status segment

CLEAR DAQ.STATUS STATUS

–

(m_util) (m_util) Clears daq status bits in the global daq status segment

CLEAR HISTOGRAM

name

(m_histogram) (m_histogram) Clears histogram.

CLEAR PIPES

–

(m_read_meb) (m_read_meb) Clears the subevent pipes (queue).

CLEAR POLYGON

name

(m_collector) (m_collector) Clear polygon condition counters.

CLEAR TRIG.MOD

–

(m_util) (m_util) Resets the trigger module.

CLEAR WINDOW

name

(m_collector) (m_collector) Clear window condition counters.
CLOSE FILE  [tape]
           (m_transport) (m_transport) Closes file.

CNAF   c n a f  [d r]
       (m_camac) (m_camac) Executes local CAMAC cnaf.

COMMENT  line -ERROR -INFO -COMMAND
          (m_dispatch) (m_dispatch) Writes line to log file.

CONNECT DISPATCHER  [node] -ALL
                     (m_prompt) (m_prompt) Connects prompter to remote dispatcher.

CONNECT RFIO  node [mode path fraction maxfiles convention]
              (m_transport) (m_transport) connect to RFIO archive or disk server

CONNECT TRANSPORT  name
                   (m_to) (m_to) connect to m_transport

CREATE BASE  name histogram size
             (m_histogram) (m_histogram) Creates histogram data base.

CREATE CONDITION INCLUDE  name -POLYGON -WINDOWS
                           (m_collector) (m_collector) Creates include files.

CREATE CONDITION POLYGON  polygons -NEW
                          (m_collector) (m_collector) Initialize polygon conditions.

CREATE CONDITION WINDOW  windows -NEW
                         (m_collector) (m_collector) Initialize window conditions.

CREATE FILES  project [-ANALYSIS -STARTUP -CREATE]
            (m_histogram) (m_lea) Create include files and procedure.

CREATE HISTOGRAM  name dim type 1channels 1low 1up [1binsize 2channels 2low
                   2up 2binsize]
                 (m_histogram) (m_histogram) Creates histogram.

CREATE POLYGON  name [description datafile]
                (m_collector) (m_collector) Create polygon condition.

CREATE WINDOW  name [description x1 x2 y1 y2]
              (m_collector) (m_collector) Create window condition.

DEFINE COMMANDS  task file -LOG
                   (m_dispatch) (m_dispatch) Defines commands for task.

DELETE BASE  name
              (m_histogram) (m_histogram) Deletes histogram data base.
DELETE HISTOGRAM  name
(m_histogram) (m_histogram) Deletes histogram.

DETACH BASE  name
(m_histogram) (m_histogram) Detaches histogram data base.

DISABLE CVC_CAM_IRQ  –
(m_util) (m_util) Disables CAMAC LAM and VSB interrupts

DISABLE DELAYED_EB  –
(m_ds) (m_ds) disables delayed event building

DISABLE DEL_EB_COL  –
(m_collector) (m_collector) disables delayed event building

DISABLE EVENT_COPY  –
(m_collector) (m_collector) Disables event copy during data acquisiton

DISABLE HISTOGRAM  –
(m_collector) (m_collector) Disables histogramming during data acquisition

DISABLE RECEIVE_DATA  –
(m_dr) (m_dr) Disables data receive of a dr node

DISABLE TCP  –
(m_transport) (m_transport) Disables tcp connection.

DISABLE TRIG_MOD  –
(m_util) (m_util) Disables trigger module on trigger bus

DISCONNECT CLIENT_STREAM_SERV  [waitsec]
(m_stream_serv) (m_stream_serv) Tell client to close socket.

DISCONNECT CLIENT_TRANSPORT  [waitsec]
(m_transport) (m_transport) Tell client to close socket.

DISCONNECT DISPATCHER  [node] -ALL -KILL
(m_prompt) (m_prompt) Disconnects prompter from remote dispatcher.

DISCONNECT RFIO  –
(m_transport) (m_transport) disconnect from RFIO server

DISCONNECT TRANSPORT  name
(m_to) (m_to) disconnect from m_transport

DISMOUNT TAPE  [tape] -UNLOAD
(m_transport) (m_transport) Dismounts tape.
**DUMP BASE**  base file -COMPRESS -OVERWRITE
(m_histogram) (m_histogram) Dumps base into file.

**DUMP HISTOGRAM**  name file -SEPARATE
(m_histogram) (m_histogram) Dumps histogram in file or files.

**DUMP WINDOW**  file
(m_collector) (m_collector) Dump window conditions.

**ENABLE CVC_CAM_IRQ**  –
(m_util) (m_util) Enables CAMAC LAM and VSB interrupts

**ENABLE DABC**  –
(m_transport) (m_transport) Set mode to run with DABC. Transport waits for TCP client.

**ENABLE DELAYED EB**  lo_mark hi_mark
(m_ds) (m_ds) enables delayed event building

**ENABLE DEL_EB_COL**  lo_mark hi_mark
(m_collector) (m_collector) enables delayed event building

**ENABLE EVENT_COPY**  –
(m_collector) (m_collector) Enables event copy during data acquisition

**ENABLE HISTOGRAM**  basename
(m_collector) (m_collector) Enables histogramming during data acquisition

**ENABLE IRQ**  –
(m_util) (m_util) Disables trigger module to send IRQ or LAM

**ENABLE RECEIVE_DATA**  –
(m_dr) (m_dr) Enables data receive of a dr node

**ENABLE TCP**  -INCLUSIVE -EXCLUSIVE
(m_transport) (m_transport) Enables tcp connection.

**ENABLE TRIG_MOD**  –
(m_util) (m_util) Enables trigger module on trigger bus

**HELP**  [k1 k2 k3 k4 k5 library] -MBS
(m_prompt) (m_prompt) Outputs help information by keywords.

**INITIALIZE TAPE**  label [tape]
(m_transport) (m_transport) Initializes tape.
LOAD ML_SETUP  usf_file
(m_util) (m_util) Loads setup file of multi-layer multi-branch daq system

LOAD MO_SETUP  usf_file
(m_util) (m_util) Loads setup file multi-output/collector mbs system

LOAD READOUT  usf_file
(m_read_meb) (m_read_meb) Loads readout table.

LOAD SETUP  usf_file [crate_nr]
(m_util) (m_util) Loads setup file.

LOAD SLAVE_READOUT  usf_file
(m_read_cam_slav) (m_read_cam_slav) Loads readout table.

MOUNT TAPE  [tape]
(m_transport) (m_transport) Mounts tape.

NEWS  [facility item path] -ALL
(m_dispatch) (m_dispatch) Outputs news.

ON ERROR  -CONTINUE -BREAK
(m_dispatch) (m_dispatch) Sets error handling in procedures.

OPEN FILE  name [tape size number first inhead outhead]
(m_transport) (m_transport) Opens file on tape.

PROTECT HISTOGRAM  name -UNPROTECT
(m_histogram) (m_histogram) [Un]protect histogram from being cleared.

PSHELL  command [a1 a2 a3 a4]
(m_prompt) (m_prompt) Executes shell command line.

REMOTE ACCESS  [nodelist]
(m_prompt) (m_prompt) Grants command access from nodes.

REMOTE RESET  [node] -ALL
(m_prompt) (m_prompt) Cleans up remote nodes. Remote program is m_remote.

REMOTE SHOW  [node] [task] -LOCAL -TASKS -DAQ -NET
(m_prompt) (m_prompt) Shows remote info.

REMOVE SEGMENTS
(m_util) (m_util) Removes critical segments

RESET  [node] [task] -LOCAL
(m_remote) (m_remote) Resets remote node. Executed by alias remote.
RESTORE BASE  base file -OVERWRITE -NEW
(m_histogram)  (m_histogram) Restores base from file.

RESTORE POLYGON  file -NEW -OVER -APPEND
(m_collector)  (m_collector) Restore polygon condition from file.

RESTORE WINDOW  file -NEW -OVER -APPEND
(m_collector)  (m_collector) Restore window condition from file.

SET ARECEIVER FLUSHTIME  time
(m_ar)  (m_ar) Sets stream flushtime

SET DISPATCHER  [node]
(m_prompt)  (m_prompt) Sets terminal to remote dispatcher.

SET DR_FLUSHTIME  time
(m_dr)  (m_dr) Sets stream flushtime

SET EVENT_SERV  [scale]  [events]  [maxclnt] -ALL
(m_event_serv)  (m_event_serv) Sets m_event_serv parameters.

SET FILEHEADER  string [line] -RUNID -EXPERIMENT
(m_transport)  (m_transport) Specify fileheader information.

SET FLUSHTIME  time
(m_collector)  (m_collector) Sets stream flushtime

SET HISTOGRAM CHANNEL  name value xchan [ychan] -INCREMENT
(m_histogram)  (m_histogram) Sets channel content of histogram.

SET HISTOGRAM TEXT  name [text] -TITLE -XTXT -YTXT -CONT
(m_histogram)  (m_histogram) Sets lettering text field of histogram.

SET MAXCLIENTS ESONE_SERV  maxclnt
(m_esone_serv)  (m_esone_serv) Sets maximum number of clients for m_esone_serv.

SET MAXCLIENTS EVENT_SERV  maxclnt
(m_event_serv)  (m_event_serv) Sets maximum number of clients for m_event_serv.

SET PRINT ESONE_SERV  -ON -OFF
(m_esone_serv)  (m_esone_serv) Sets terminal output Eson data for Esone Server.

SET RIRECEIVER FLUSHTIME  time
(m_rirece)  (m_rirece) Sets stream flushtime

SET STREAM_SERV  [scale] -[NO]SYNC -[NO]KEEP -SCALED_KEEP -CLEAR
(m_stream_serv)  (m_stream_serv) Selects scaledown of streams.
MBS Command Summary

**SET TASK** task pid -CLEAR
(m_dispatch) (m_dispatch) Sets task id.

**SET TORECEIVER FLUSHTIME** time
(m_to) (m_to) set stream flush time

**SET TRIG_MOD** -SLAVE
(m_util) (m_util) Sets trigger module.

**SET USER VALUE** index values -clear -show
(m_util) (m_util) Set user values in daqst.

**SET VERBOSE DISPATCHER** -ON -OFF
(m_dispatch) (m_dispatch) Sets verbosity for dispatcher.

**SET VERBOSE ESONE_SERV** -ON -OFF
(m_esone_serv) (m_esone_serv) Sets verbosity for Esone Server.

**SET VERBOSE EVENT_SERV** -ON -OFF
(m_event_ser) (m_event_ser) Sets verbosity for m_event_ser.

**SET VERBOSE GLOBAL** -ON -OFF -NEUTRAL
(m_util) (m_util) Sets verbosity for all tasks.

**SET VERBOSE HISTOGRAM** -ON -OFF
(m_histogram) (m_histogram) Sets verbosity for m_histogram.

**SET VERBOSE PROMPT** -ON -OFF
(m_prompt) (m_prompt) Sets verbosity for m_prompt.

**SET WINDOW** name [x1 x2 y1 y2] -CURSOR
(m_collector) (m_collector) Set limits of window condition.

**SET XDISPLAY** node
(m_prompt) (m_prompt) Set name of remote display.

**SHOW** [node] [task] -LOCAL -TASKS -DAQ -NET
(m_remote) (m_remote) Shows remote info. Executed by alias remote.

**SHOW ACQUISITION** [seconds] -SETUP -CRATES -SERVER -RATE -LOG
(m_util) (m_util) Shows acquisition.

**SHOW BASE** [name] -FULL
(m_histogram) (m_histogram) Shows information about histogram data base.

**SHOW COMMANDS** [task] -FULL -ALL
(m_dispatch) (m_dispatch) Shows known commands.
SHOW CVC_IRQ_MASK
   (m_util) (m_util) Reads irq mask of the CVC irq controller

SHOW DISPATCHER [node]
   (m_prompt) (m_prompt) Shows connections to remote dispatchers.

SHOW ENVIRONMENT
   (m_dispatch) (m_dispatch) Shows environment parameters.

SHOW ESONE_SERV -FULL -LOG -CLIENT
   (m_esone_serv) (m_esone_serv) Shows status of m_esone_serv.

SHOW EVENT_SERV -FULL -LOG -CLIENT
   (m_event_serv) (m_event_serv) Shows status of m_event_serv.

SHOW FILE [tape]
   (m_transport) (m_transport) Shows file output information.

SHOW HISTOGRAM name [1offset] [1channels] [2offset] [2channels] -DATA -FULL
   (m_histogram) (m_histogram) Shows histogram info and content.

SHOW INPUT NODES
   (m_to) (m_to) show input nodes connected via tcp sockets

SHOW MESSAGE
   (m_dispatch) (m_dispatch) Shows status of internal message file.

SHOW ML_SET UP
   (m_util) (m_util) Shows multilayer setup parameters

SHOW POLYGON name -DATA
   (m_collector) (m_collector) Show polygon conditions.

SHOW RATE [seconds] -OFF -ON
   (m_util) (m_util) Shows acquisition rate.

SHOW SETUP
   (m_util) (m_util) Shows setup parameters

SHOW STATUS
   (m_util) (m_util) Shows daqst parameters

SHOW STREAM_SERV -CLEAR
   (m_stream_serv) (m_stream_serv) Shows modes and counters.

SHOW TAPE [tape]
   (m_transport) (m_transport) Shows tape information.
MBS Command Summary

SHOW TASK [task] -FULL -ALL
(m_dispatch) (m_dispatch) Shows known tasks.

SHOW TRIG MOD
(m_util) (m_util) Shows current setup of trigger module.

SHOW WINDOW name
(m_collector) (m_collector) Show window conditions.

START ACQUISITION
(m_util) (m_util) Starts acquisition.

START ARECEIVER
(m_ar) (m_ar) Enables data receive of a ar node

START EVENT SERV [scale] [events] [maxclnt]
(m_event_serv) (m_event_serv) Starts m_event_serv.

START MESSAGE
(m_dispatch) (m_dispatch) Starts the message logger.

START OUTPUT
(m_to) (m_to) starts time sorting and event output

START RIRECEIVER
(m_rirecc) (m_rirecc) Enables data receive of a rirec node

START TASK task [file]
(m_dispatch) (m_dispatch) Starts task.

STOP ACQUISITION
(m_util) (m_util) Stops acquisition.

STOP ARECEIVER
(m_ar) (m_ar) Disables data receive of a ar node

STOP OUTPUT
(m_to) (m_to) stops time sorting and event output

STOP RIRECEIVER
(m_rirecc) (m_rirecc) Disables data receive of a rirec node

STOP TASK [task] [pid] -ALL -KILL -ZOMBIE
(m_dispatch) (m_dispatch) Stops task by sending command "exit"

TYPE EVENT [events id control crate offset items] -SAMPLE -VERBOSE -DECIMAL
-BUFFER
(m_transport) (m_transport) Prints events.
<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>TYPE STREAM</td>
<td>buffers -VERBOSE -HEADER</td>
</tr>
<tr>
<td></td>
<td>(m_transport) (m_transport) Prints buffers of stream.</td>
</tr>
<tr>
<td>VOID</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>(m_daq_rate) (m_daq_rate) place holder, do not execute.</td>
</tr>
<tr>
<td>VRA16D16</td>
<td>addr [size repeat]</td>
</tr>
<tr>
<td></td>
<td>(m_vme_serv) (m_vme_serv) VME read A16 D16</td>
</tr>
<tr>
<td>VRA24D16</td>
<td>addr [size repeat]</td>
</tr>
<tr>
<td></td>
<td>(m_vme_serv) (m_vme_serv) VME read A24 D16</td>
</tr>
<tr>
<td>VRA24D32</td>
<td>addr [size repeat]</td>
</tr>
<tr>
<td></td>
<td>(m_vme_serv) (m_vme_serv) VME read A24 D32</td>
</tr>
<tr>
<td>VRA32D16</td>
<td>addr [size repeat]</td>
</tr>
<tr>
<td></td>
<td>(m_vme_serv) (m_vme_serv) VME read A32 D16</td>
</tr>
<tr>
<td>VRA32D32</td>
<td>addr [size repeat]</td>
</tr>
<tr>
<td></td>
<td>(m_vme_serv) (m_vme_serv) VME read A32 D32</td>
</tr>
<tr>
<td>VWA16D16</td>
<td>addr data [size repeat]</td>
</tr>
<tr>
<td></td>
<td>(m_vme_serv) (m_vme_serv) VME write A16 D16</td>
</tr>
<tr>
<td>VWA24D16</td>
<td>addr data [size repeat]</td>
</tr>
<tr>
<td></td>
<td>(m_vme_serv) (m_vme_serv) VME write A24 D16</td>
</tr>
<tr>
<td>VWA24D32</td>
<td>addr data [size repeat]</td>
</tr>
<tr>
<td></td>
<td>(m_vme_serv) (m_vme_serv) VME write A24 D32</td>
</tr>
<tr>
<td>VWA32D16</td>
<td>addr data [size repeat]</td>
</tr>
<tr>
<td></td>
<td>(m_vme_serv) (m_vme_serv) VME write A32 D16</td>
</tr>
<tr>
<td>VWA32D32</td>
<td>addr data [size repeat]</td>
</tr>
<tr>
<td></td>
<td>(m_vme_serv) (m_vme_serv) VME write A32 D32</td>
</tr>
<tr>
<td>XSHELL</td>
<td>command [a1 a2 a3 a4]</td>
</tr>
<tr>
<td></td>
<td>(m_dispatch) (m_dispatch) Executes shell command line.</td>
</tr>
</tbody>
</table>
Chapter 4

MBS Command Description
ATTACH

ATTACH BASE

ATTACH BASE name

PURPOSE (m_histogram) Attaches existing histogram data base.

PARAMETERS

name name of segment

Description

Function Create or attach histogram data base.
Routine f_his_cmd_attbas
Task m_histogram
CAMAC

CAMAC CNAF

CAMAC CNAF c n a f [d r] -LOG -NOPRINT

PURPOSE
(m_esone_serv) Executes local CAMAC cnaf.

PARAMETERS

  c  crate
  n  station
  a  address
  f  function
  d  data
  r  repetition
-LOG  write results to log file
-NOPRINT  no terminal output

Description

  Function  Execute local CAMAC cnaf.
  Routine  f_es_cmd_cnaf
  Task  m_esone_serv
CAMAC FILE

CAMAC FILE filenam -LOG -NOPRINT

PURPOSE
(m_esone_serv) Executes local CAMAC cnaf.

PARAMETERS

filenam
filename with cnaf data:

lines with c n a f d r like arguments of

CAMAC CNAF COMMAND

-LOG
write results to log file

-NOPRINT
no terminal output

Description

Function
Execute local CAMAC cnaf.

Routine f_es_cmd_file

Task m_esone_serv
CLEAR

CLEAR_DAQ_STATUS_COUNTER

PURPOSE (m_util) Clears status counters in the global daq status segment
PARAMETERS none

Description
Function Clear status counters in the global daq status segment
Routine f_ut_clear_daqst
Task m_util

CLEAR_DAQ_STATUS_PROCTAB

PURPOSE (m_util) Clears daq process table in the global daq status segment
PARAMETERS none
Description

Function: Clear daq process table in the global daq status segment
Routine: f_ut_clear_daqst
Task: m_util

CLEAR DAQ_STATUS STATUS

PUBLIC DAQ_STATUS

PURPOSE: (m_util) Clears daq status bits in the global daq status segment

PARAMETERS

none

Description

Function: Clear daq status bits in the global daq status segment
Routine: f_ut_clear_daqst
Task: m_util

CLEAR HISTOGRAM

PUBLIC HISTOGRAM name

PURPOSE: (m_histogram) Clears histogram.

PARAMETERS

name: name of histogram, also wildcard.
Description

Function  Clear histogram.
Routine    f_his_cmd_clrhis
Task       m_hisogram

CLEAR PIPES

PURPOSE (m_read_meb) Clears the subevent pipes (queue).

PARAMETERS

Description

Function  Clear the subevent pipes (queue).
(not yet implemented)
Routine
Task      m_read_meb

CLEAR POLYGON

PURPOSE (m_collector) Clear polygon condition counters.

PARAMETERS
name      name of condition
**CLEAR TRIG_MOD**

**PURPOSE**

(m_util) Resets the trigger module.

**PARAMETERS**

none

---

**CLEAR WINDOW**

**PURPOSE**

(m_collector) Clear window condition counters.

**PARAMETERS**

name name of condition
### Description

<table>
<thead>
<tr>
<th>Function</th>
<th>Routine</th>
<th>Task</th>
</tr>
</thead>
<tbody>
<tr>
<td>Clear window condition.</td>
<td>f_win_cmd_c</td>
<td>m_collector</td>
</tr>
</tbody>
</table>
CLOSE FILE

CLOSE FILE [tape]

PURPOSE
(m_transport) Closes file.

PARAMETERS

tape
Optional tape number

Description

Function
Closes file on tape. The acquisition is not stopped!

Routine
f_

Task
m_transport
CNAF

CNAF c n a f [d r]

PURPOSE  (m_camac) Executes local CAMAC cnaf.

PARAMETERS

- **c**: crate
- **n**: station
- **a**: address
- **f**: function
- **d**: data
- **r**: repetition

Description

- **Function**: Execute local CAMAC cnaf.
- **Routine**: f_cnaf
- **Task**: m_camac
COMMENT

COMMENT

COMMENT line -ERROR -INFO -COMMAND

PURPOSE (m_dispatch) Writes line to log file.

PARAMETERS

command Line inserted in log file.

-ERROR Mark as error line

-INFO Mark as info line

-COMMAND Mark as command line

Description

Function Use F_ERROR to write line to log file and terminal.

Example COMM "Neue Schicht"

Routine f_disp_comment

Task m_dispatch
CONNECT

CONNECT DISPATCHER

CONNECT DISPATCHER [node] -ALL

PURPOSE (m_prompt) Connects prompter to remote dispatcher.

PARAMETERS

node Name of remote node.

-ALL All known nodes

Description

Function Opens TCP link to remote dispatcher which is started. Message client is started. The remote nodes must be declared as trustedhost and trusteduser in the .rhosts file on users directory.

Routine f_ifa_new_connect

Task m_prompt

CONNECT RFIO

CONNECT RFIO node [mode path fraction maxfiles convention]

DISKSERVER -ARCHIVESERVER

PURPOSE (m_transport) connect to RFIO archive or disk server

PARAMETERS

Node node name of RFIO server
Path
Destination where to be copied: "lustre/..." => fully qualified path name; if not existing: will be created (see below). "RC" => read cache

Fraction
Copy each ith file to Path if tape migration fails: ignore Fraction, copy each file.

MaxFiles
For dataFS only: 0: no file number limit > 0: max no. of files to be written to directory. Files already existing are ignored. If MaxFiles is reached, a new directory will be created. Running number is appended to directory name, initially specified ...

-DISKSERVER
Server is private disk server

-ARCHIVESERVER
Server is public tape archive server

-COPYTOPATH
Copy to Path after file written to Write Cache (for high data rates, don’t interfere writing to cache).

-PARALLEL
For lustre only: write each data buffer in parallel to Write Cache and to Path (for low data rates, first buffers quickly available in lustre)

-DIRDATE
Hades rule for creation of initial/new directories: path name contains time stamp initially specified ...

Description
Function
Establish connection to archive. Subsequent open/close file commands leave the connection connected. Open command must use -rfio switch. Without -rfio writing to local disk is possible. Connection must be terminated by command DISCONNECT RFIO

Routine
f_conn_rfio

Task
m_transport
CONNECT TRANSPORT

CONNECT TRANSPORT name

PURPOSE (m_to) connect to m_transport

PARAMETERS

name node name where m_transport runs

Description

Function connect to m_transport
Routine f_to_connect_transport
Task m_to
CREATE

CREATE BASE

CREATE BASE name histogram size

PURPOSE (m_histogram) Creates histogram data base.

PARAMETERS

name name of segment
histogram maximum number of histograms
size size of segment in kByte

Description

Function Create or attach histogram data base.
Routine f_his_cmd_crebas
Task m_histogram

CREATE CONDITION INCLUDE

CREATE CONDITION INCLUDE name -POLYGON -WINDOWS

PURPOSE (m_collector) Creates include files.

PARAMETERS

name name postfix for files
-POLYGON write file POLDEF_name.h
-WINDOWS write file WINDEF_name.h
CREATE

Description

Function Creates include files. In these files all names WIN_name or POLY_name from the current window or polygon directory are defined to the index of the condition. At runtime it is checked that the names are defined properly.

Routine f_col_cmd crefil

Task m_collector

CREATE CONDITION POLYGON

CREATE CONDITION POLYGON polygons -NEW

PURPOSE (m_collector) Initialize polygon conditions.

PARAMETERS

polygons Number of conditions

-NEW Remove current polygons

Description

Function Initialize polygon condition.

Routine f_poly_cmd_i

Task m_collector

CREATE CONDITION WINDOW

CREATE CONDITION WINDOW windows -NEW

PURPOSE (m_collector) Initialize window conditions.
PARAMETERS

windows  Number of conditions
-NEW  Remove current windows

Description

Function  Initialize window condition.
Routine  f_win_cmd_i
Task  m_collector

CREATE FILES

CREATE FILES project [-ANALYSIS -STARTUP -CREATE]

PURPOSE  (m_lea) Create include files and procedure.
PARAMETERS

project  used to name the files.
-STARTUP  not used
-ANALYSIS  create analysis function
-CREATE  not used

Description

Function  Create three files:
hisdef_project.h  Definition include file
hisini_project.h  Initialization include file
hiscre_project.scom  Procedure to create histograms
**CREATE**

f_mbs_anal_project.c  User function when -ANALYSIS was given.
Routine  f_his_cmd_crefil
Task  m_histogram

**CREATE HISTOGRAM**

CREATE HISTOGRAM name dim type 1channels 1low 1up [1binsize 2channels 2low 2up 2binsize]

**PURPOSE**
(m_histogram) Creates histogram.

**PARAMETERS**

- **name**  name of histogram
- **dim**  dimension of histogram (dim=1,2)
- **type**  type of histogram 'r' : real or 'i' : integer
- **chan1**  number of channels 1st dimension
- **lo1**  lower limit of histogram 1st dimension
- **up1**  upper limit of histogram 1st dimension
- **chan2**  number of channels 2nd dimension
- **lo2**  lower limit of histogram 2nd dimension
- **up2**  upper limit of histogram 2nd dimension

**Description**

- **Function**  Create histogram.
- **Routine**  f_his_cmd_crehis
- **Task**  m_histogram
CREATE POLYGON

CREATE POLYGON name [description datafile]

PURPOSE  (m_collector) Create polygon condition.
PARAMETERS
  name     name
  description  string with description
  datafile  optional file with data points as generated by polygon editor.

Description
  Function  Create polygon condition. polygon is closed or set to values from data file.
  Routine   f_poly.cmd_n
  Task      m_collector

CREATE WINDOW

CREATE WINDOW name [description x1 x2 y1 y2]

PURPOSE  (m_collector) Create window condition.
PARAMETERS
  name     name
  description  string with description (in ””)
  x1,x2,y1,y2  optional values
### Description

<table>
<thead>
<tr>
<th><strong>Function</strong></th>
<th>Create window condition. Window is closed or set to the specified values.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Routine</strong></td>
<td>f_win_cmd_n</td>
</tr>
<tr>
<td><strong>Task</strong></td>
<td>m_collector</td>
</tr>
</tbody>
</table>
DEFINE

DEFINE COMMANDS

**DEFINE COMMANDS task file -LOG**

**PURPOSE**
(m_dispatch) Defines commands for task.

**PARAMETERS**

- **task**
  Task (program) name or * for all tasks

- **file**
  name of command definition file (cdf)

- **-LOG**
  Output definitions

**Description**

**Function**
Tasks, commands and arguments can be defined in text files like /mbs/deve/set/mbscom.cdf. The definitions of such a file can be read by this command. The task which has to execute the commands, must specify functions for each command. This is done by calls to routines f_cmd_cdf_def and f_cmd_set_entry. See command interface description.

**Routine**

f_disp_def_cmd

**Task**
m_dispatch
DELETE

DELETE BASE

DELETE BASE name

PURPOSE  (m_histogram) Deletes histogram data base.
PARAMETERS
  name  name of segment

Description
  Function  Delete histogram data base.
  Routine  f_his_cmd_delbas
  Task  m_histogram

DELETE HISTOGRAM

DELETE HISTOGRAM name

PURPOSE  (m_histogram) Deletes histogram.
PARAMETERS
  name  name of histogram, also wildcard.
## Description

<table>
<thead>
<tr>
<th>Function</th>
<th>Delete histogram in histogram manager.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Routine</td>
<td>f_his_cmd_delhis</td>
</tr>
<tr>
<td>Task</td>
<td>m_histogram</td>
</tr>
</tbody>
</table>
DETACH

DETACH BASE

DETACH BASE name

PURPOSE  (m_histogram) Detaches histogram data base.

PARAMETERS

  name  name of segment

Description

  Function  Detach histogram data base.
  Routine  f_his_cmd_detbas
  Task  m_histogram
DISABLE

DISABLE CVC_CAM_IRQ

DISABLE CVC_CAM_IRQ

**PURPOSE**  
(m_util) Disables CAMAC LAM and VSB interrupts

**PARAMETERS**  
none

**Description**

**Function**  
Disable CAMAC LAM and VSB interrupts of the CVC

**Routine**  
f_ut_cvc_cam_irq

**Task**  
m_util

DISABLE DELAYED_EB

DISABLE DELAYED_EB

**PURPOSE**  
(m_ds) disables delayed event building

**PARAMETERS**
DISABLE

Description
  Function  disables delayed event building in spill
  Routine   f_ds_dis_del_eb
  Task      m_ds

DISABLE DEL_EB_COL

DISABLE DEL_EB_COL

PURPOSE  (m_collector) disables delayed event building
PARAMETERS

Description
  Function  disables delayed event building in spill
  Routine   f_col_dis_del_eb
  Task      m_collector

DISABLE EVENT_COPY

DISABLE EVENT_COPY

PURPOSE  (m_collector) Disables event copy during data acquisition
PARAMETERS
Description

Function  disable event copy during data acquisition
Routine    f_col_cmd_dis_evt_copy
Task       m_collector

DISABLE HISTOGRAM

DISABLE HISTOGRAM

PURPOSE (m_collector) Disables histogramming during data acquisition
PARAMETERS
basename

Description

Function  disable histogramming during data acquisition
Routine    f_col_cmd_dis_hist
Task       m_collector

DISABLE RECEIVE_DATA

DISABLE RECEIVE_DATA

PURPOSE (m_dr) Disables data receive of a dr node
PARAMETERS
Description

Function    Disables data receive of a dr node
Routine      f_dr.cmd.dis_receive_data
Task         m_dr

DISABLE TCP

DISABLE TCP

PURPOSE    (m_transport) Disables tcp connection.
PARAMETERS

Description

Function    disables any tcp connection from GOOSY transport manager
Routine      f_
Task         m_transport

DISABLE TRIG_MOD

DISABLE TRIG_MOD

PURPOSE    (m_util) Disables trigger module on trigger bus
PARAMETERS
    none
## Description

<table>
<thead>
<tr>
<th>Function</th>
<th>Routine</th>
<th>Task</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td><code>f_ut_op_trig_mod</code></td>
<td><code>m_util</code></td>
</tr>
</tbody>
</table>

Disables trigger module on the trigger bus to take NOT part in a multi trigger module environment.
DISCONNECT

DISCONNECT CLIENT STREAM_SERV

PURPOSE (m_stream_serv) Tell client to close socket.
PARAMETERS
waitsec After setting the bit to send a termination buffer, wait. This is needed in command procedures to wait for next command, i.e. stopping stream server.

Description
Function Triggers to send a buffer with l_evt=-1 to client. Client then should close socket.
Routine f_str_disconnect
Task m_stream_serv

DISCONNECT CLIENT TRANSPORT

PURPOSE (m_transport) Tell client to close socket.
PARAMETERS
waitsec After setting the bit to send a termination buffer, wait. This is needed in command procedures to wait for next command, i.e. stopping transport.
Description

Function
Triggers to send a buffer with l_evt=-1 to client. Client then should close socket.

Routine
f_tra_disconnect

Task
m_transport

DISCONNECT DISPATCHER

PURPOSE (m_prompt) Disconnects prompter from remote dispatcher.

PARAMETERS

node
Name of remote node.

-ALL
All known nodes

-KILL
Stop dispatcher task

Description

Function

Routine
f_ifa_disconnect

Task
m_prompt

DISCONNECT RFIO

PURPOSE (m_transport) disconnect from RFIO server
PARAMETERS

Description

Function | disconnect from RFIO server
Routine   | f_disconn_rfi
Task      | m_transport

DISCONNECT TRANSPORT

DISCONNECT TRANSPORT name

PURPOSE | (m_to) disconnect from m_transport
PARAMETERS

name | node name where m_transport runs

Description

Function | disconnect from m_transport
Routine   | f_to_disco_transport
Task      | m_to
DISMOUNT

DISMOUNT TAPE

DISMOUNT TAPE [tape] -UNLOAD

PURPOSE (m_transport) Dismounts tape.

PARAMETERS

tape Optional tape number

-UNLOAD unloads tape after rewind

Description

Function Dismount tape.

Routine f_

Task m_transport
DUMP

DUMP BASE

DUMP BASE base file -COMPRESS -OVERWRITE

PURPOSE (m_histogram) Dumps base into file.

PARAMETERS

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>base</td>
<td>name of segment = name of base</td>
</tr>
<tr>
<td>file</td>
<td>name of file</td>
</tr>
<tr>
<td>-COMPRESS</td>
<td>compress base</td>
</tr>
<tr>
<td>-OVERWRITE</td>
<td>overwrite file</td>
</tr>
</tbody>
</table>

Description

<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dump base</td>
<td>Dump base into file.</td>
</tr>
</tbody>
</table>

Routine  f_his_cmd_dmpbas

Task      m_histogram

DUMP HISTOGRAM

DUMP HISTOGRAM name file -SEPARATE

PURPOSE (m_histogram) Dumps histogram in file or files.

PARAMETERS

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>name of histogram, also wildcard.</td>
</tr>
<tr>
<td>file</td>
<td>file name</td>
</tr>
</tbody>
</table>
-SEPARATE  when using wildcard, file name equal histogram name

### Description

**Function**
Dump histogram in file or files.

**Routine**
f_his_cmd_dmphis

**Task**
m_histogram

---

### DUMP WINDOW

**DUMP WINDOW file**

**PURPOSE**
(m_collector) Dump window conditions.

**PARAMETERS**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>file</td>
<td>File name</td>
</tr>
</tbody>
</table>

### Description

**Function**
Dump window conditions to text file.

**Routine**
f_win_cmd_d

**Task**
m_collector
ENABLE

ENABLE CVC_CAM_IRQ

PURPOSE  (m_util) Enables CAMAC LAM and VSB interrupts
PARAMETERS
none

Description

Function  Enable CAMAC LAM and VSB interrupts.
this is normally done during boot time of the CVC
Routine  f_ut_cvc_cam_irq
Task  m_util

ENABLE DABC

PURPOSE  (m_transport) Set mode to run with DABC. Transport waits for TCP client. Without client blocks. Sends buffers with variable length (type 100).
PARAMETERS

<table>
<thead>
<tr>
<th>parameter</th>
<th>description</th>
</tr>
</thead>
<tbody>
<tr>
<td>type</td>
<td>Buffer type, default 100</td>
</tr>
</tbody>
</table>
## Description

<table>
<thead>
<tr>
<th>Function</th>
<th>enable_dabc</th>
</tr>
</thead>
<tbody>
<tr>
<td>Task</td>
<td>m_transport</td>
</tr>
</tbody>
</table>

### ENABLE DELAYED_EB

<table>
<thead>
<tr>
<th>Command</th>
<th>enable_dabc</th>
</tr>
</thead>
<tbody>
<tr>
<td>Task</td>
<td>m_transport</td>
</tr>
</tbody>
</table>

#### ENABLE DELAYED_EB lo_mark hi_mark

<table>
<thead>
<tr>
<th>PURPOSE</th>
<th>(m_ds) enables delayed event building</th>
</tr>
</thead>
<tbody>
<tr>
<td>PARAMETERS</td>
<td></td>
</tr>
<tr>
<td>lo_mark</td>
<td>low water mark in percent</td>
</tr>
<tr>
<td>hi_mark</td>
<td>high water mark in percent</td>
</tr>
</tbody>
</table>

## Description

<table>
<thead>
<tr>
<th>Function</th>
<th>enables delayed event building in spill</th>
</tr>
</thead>
<tbody>
<tr>
<td>Routine</td>
<td>f_ds_ena_del Eb</td>
</tr>
<tr>
<td>Task</td>
<td>m_ds</td>
</tr>
</tbody>
</table>

### ENABLE DEL_EB_COL

<table>
<thead>
<tr>
<th>Command</th>
<th>enable_dabc</th>
</tr>
</thead>
<tbody>
<tr>
<td>Task</td>
<td>m_ds</td>
</tr>
</tbody>
</table>

#### ENABLE DEL_EB_COL lo_mark hi_mark

<table>
<thead>
<tr>
<th>PURPOSE</th>
<th>(m_collector) enables delayed event building</th>
</tr>
</thead>
<tbody>
<tr>
<td>PARAMETERS</td>
<td></td>
</tr>
<tr>
<td>lo_mark</td>
<td>low water mark in percent</td>
</tr>
<tr>
<td>hi_mark</td>
<td>high water mark in percent</td>
</tr>
</tbody>
</table>
### Description

**Function**
- enables delayed event building in spill

**Routine**
- `f_col_ena_del_eb`

**Task**
- `m_collector`

---

### ENABLE EVENT_COPY

**PURPOSE**
- (m_collector) Enables event copy during data acquisition

**PARAMETERS**

### Description

**Function**
- enable event copy during data acquisition

**Routine**
- `f_col_cmd_ena_evt_copy`

**Task**
- `m_collector`

---

### ENABLE HISTOGRAM

**PURPOSE**
- (m_collector) Enables histogramming during data acquisition

**PARAMETERS**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>basename</code></td>
<td>histogram base (segment) name</td>
</tr>
</tbody>
</table>
Description

**Function**: enable histogramming during data acquisition

**Routine**: f_col_cmd_ena_hist

**Task**: m_collector

---

**ENABLE IRQ**

**ENABLE IRQ**

**PURPOSE**: (m_util) Disables trigger module to send IRQ or LAM

**PARAMETERS**: none

---

**Description**

**Function**: disable trigger module to send an IRQ or LAM on occurrence of a trigger

**Routine**: f_ut_op_trig_mod

**Task**: m_util

---

**ENABLE RECEIVE_DATA**

**ENABLE RECEIVE_DATA**

**PURPOSE**: (m_dr) Enables data receive of a dr node

**PARAMETERS**: none
### Description

**Function**
Enables data receive of a dr node

**Routine**
f_dr_cmd_ena_receive_data

**Task**
m_dr

### ENABLE TCP

<table>
<thead>
<tr>
<th>Purpose</th>
<th>(m_transport) Enables tcp connection.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parameters</td>
<td></td>
</tr>
<tr>
<td>- INCLUSIVE</td>
<td>enable simultaneous output on tape and tcp</td>
</tr>
<tr>
<td>- EXCLUSIVE</td>
<td>enable exclusive output on tape or tcp</td>
</tr>
</tbody>
</table>

### Description

**Function**
enables tcp connection from GOOSY transport manager or other clients. Default is to write both, tape and TCP, if a client connects! To stop writing tape, when a client connects

**Routine**
f_

**Task**
m_transport

### ENABLE TRIG_MOD

<table>
<thead>
<tr>
<th>Purpose</th>
<th>(m_util) Enables trigger module on trigger bus</th>
</tr>
</thead>
</table>
PARAMETERS

none

Description

Function enables trigger module on the trigger bus to take part in a multi trigger module environment
Routine f_ut_op_trig_mod
Task m_util
HELP [k1 k2 k3 k4 k5 library] -MBS

PURPOSE (m_prompt) Outputs help information by keywords.

PARAMETERS

- key# Optional keys.
- library Help library in VAX text format. A .hlp is appended. Default is /mbs/deve/lib/mbs.hlp
- -MBS Use MBS library.

Description

- Function Help similar to VMS.
- Routine f_ifa_help
- Task m_prompt
INITIALIZE

INITIALIZE TAPE

INITIALIZE TAPE label [tape]

**PURPOSE**

(m_transport) Initializes tape.

**PARAMETERS**

- **label**
  - ANSI label of tape.
- **tape**
  - Optional tape number

**Description**

**Function**

Writes ANSI label to tape. Any information on the tape is lost.

**Routine**

f_

**Task**

m_transport
LOAD

LOAD ML_SETUP

LOAD ML_SETUP usf_file

PURPOSE (m_util) Loads setup file of multi-layer multi-branch daq system

PARAMETERS
usf_file User setup file.

Description
Function load content of mb ml setup file into memory.
Routine f_ut_load_mml_setup
Task m_util

LOAD MO_SETUP

LOAD MO_SETUP usf_file

PURPOSE (m_util) Loads setup file multi-output/collector mbs system

PARAMETERS
usf_file User setup file.
Description

**Function**  
load content of mo setup file into memory.

**Routine**  
f_ut_load_mo_setup

**Task**  
m_util

---

**LOAD READOUT**

```plaintext
LOAD READOUT usf_file
```

**PURPOSE**  
(m_read_meb) Loads readout table.

**PARAMETERS**

usf_file  
Definition file.

---

**Description**

**Function**  
Load readout table for master readout into memory.

This must be done at least once but as often
the user wants since the startup of the
readout task. The previous readout table will be
completely overwritten.

**See also**  
Command LOAD SLAVE_READOUT.

**Routine**  
f_ut_load_readout

**Task**  
m_read_meb
LOAD SETUP

LOAD SETUP usf_file [crate_nr]

PURPOSE (m_util) Loads setup file.

PARAMETERS

usf_file User setup file.
crate_nr Optional crate number (slave readout)

Description

Function load content of setup file into memory.

When executing on a slave, the crate number
of the slave must be specified.

Routine f_ut_load_setup
Task m_util

LOAD SLAVE READOUT

LOAD SLAVE_READOUT usf_file

PURPOSE (m_read_cam_slav) Loads readout table.

PARAMETERS

usf_file Definition file.
## Description

**Function** Load readout table for slave readout into memory.

This must be done at least once but as often

the user wants since the startup of the

readout task. The previous readout table will be

completely overwritten.

**See also** Command LOAD READOUT.

**Routine** f_ut_load_readout

**Task** m_read_cam_slav
MOUNT

MOUNT TAPE

MOUNT TAPE [tape]

PURPOSE (m_transport) Mounts tape.

PARAMETERS

tape Optional tape number

Description

Function Mount tape and skip files up to logical end of tape. The tape must be initialized.

Routine f_

Task m_transport
NEWS

NEWS [facility item path] -ALL

<table>
<thead>
<tr>
<th>Purpose</th>
<th>(m_dispatch) Outputs news.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parameters</td>
<td></td>
</tr>
<tr>
<td>facility</td>
<td>Optional name of facility, i.e mbs. If not specified, a listing of available facilities is printed. Facilities are files with .inf postfix.</td>
</tr>
<tr>
<td>item</td>
<td>Optional number of item to be printed.</td>
</tr>
<tr>
<td>path</td>
<td>Optional path to look for facility files. Default is /mbs/deve/lib.</td>
</tr>
<tr>
<td>-ALL</td>
<td>If specified, all items are listed, if not, only unseen ones. When called from shell, switch -a lists all items, i.e. news mbs -a.</td>
</tr>
</tbody>
</table>

Description

Function  News similar to VMS.
Routine   f_disp_info
Task      m_dispatch
ON ERROR

ON ERROR -CONTINUE -BREAK

PURPOSE (m_dispatch) Sets error handling in procedures.

PARAMETERS

-CONTINUE continue on error.
-BREAK break on error (default)

Description

Function Set error response in procedures. Valid inside all procedures. If set in a procedure, will be valid in all procedures called from it, but not above.

Routine f_disp_onerror

Task m_dispatch
OPEN

OPEN FILE

OPEN FILE name [tape size number first inhead outhead]

AUTO -PROMPT -EDIT -DISK -RFIO -MULTI -FIRST

PURPOSE
(m_transport) Opens file on tape.

PARAMETERS

name File name. if no extension ist given, .lmd ist appended

tape Optional tape number

size file size in MB, default value in auto mode is 50MB

-AUTO automatic file creation. Names of the form namexxx.lmd are created with consecutive numbers xxx. The first number is either given with the first=yyy parameter or read from the file filenum.set

number number of automatically generated files

first sets the first file number to start with. If first is not specified, the first file number is taken from the file filenum.set, which contains always the last used file number

inhead reads a goosy file header from the file specified Information from this file is used in the GOOSY file header written to tape except of Filename, Tape label and file creation date

outhead writes goosy file header data to the specified file

-PROMPT prompt for goosy file header data

-EDIT this switch can be used together with the inhead command option to edit a goosy file header read from a file

-DISK Write to local disk file.

-RFIO Write to RFIO server. Must be connected first with command connect rfio -DISK or -ARCHIVE.
-MULTI
Set this if more than one Transport node is running. filenum.set is used to get first file number in -AUTO mode.

-FIRST
Create new filenum.set file if -MULTI is given. This is necessary if first file number argument should be used.

Description

Function
When the file is opened, incoming buffers are written. Normally one first opens the file and start the acquisition then. Opening and closing files does not affect the acquisition status.

Routine
f_

Task
m_transport
PROTECT

PROTECT HISTOGRAM

PROTECT HISTOGRAM name -UNPROTECT

PURPOSE
(m_histogram) [Un]protect histogram from being cleared.

PARAMETERS
name name of histogram, also wildcard.
-UNPROTECT Enable histogram clearing

Description
Function [Un]protect histogram from being cleared.
Routine f_his_cmd_prohis
Task m_histogram
PSHELL command [a1 a2 a3 a4]

<table>
<thead>
<tr>
<th>PURPOSE</th>
<th>(m_prompt) Executes shell command line.</th>
</tr>
</thead>
<tbody>
<tr>
<td>PARAMETERS</td>
<td></td>
</tr>
<tr>
<td>command</td>
<td>Command to be executed.</td>
</tr>
<tr>
<td>a#</td>
<td>Further arguments</td>
</tr>
</tbody>
</table>

Description

<table>
<thead>
<tr>
<th>Function</th>
<th>Use function system() to fork a command.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Example</td>
<td>PSH ps -af</td>
</tr>
<tr>
<td>Routine</td>
<td>f_ifa_shell</td>
</tr>
<tr>
<td>Task</td>
<td>m_prompt</td>
</tr>
</tbody>
</table>
REMOTE

REMOTE ACCESS

REMOTE ACCESS [nodelist]

PURPOSE (m_prompt) Grants command access from nodes.

PARAMETERS

nodelist Nodelist

Description

Function Grant nodes command access by f_pr_send.
Routine f_pr_access
Task m_prompt

REMOTE RESET

REMOTE RESET [node] -ALL

PURPOSE (m_prompt) Cleans up remote nodes. Remote program is m_remote.

PARAMETERS

node Reset one node
-ALL Reset all known nodes.
See also RESET command of program remote.
Description

Function

Resets nodes by following steps: 1. Disconnect all dispatchers. 2. Kill all MBS tasks of user (remote only). m_dispatch m_col_vme m_collector m_esone_serv m_event_serv m_read_cam_slav m_read_web m_stream_serv m_transport m_util m_msg_log 3. Create new msg files (remote only) 4. Clear DAQ status (remote only) After RESET REM -ALL the tasks on the local node are still there. One must use standalone program m_remote "remote reset" to cleanup the local node, too. Other remote commands:

remote show

show remote tasks, message files and netstat.

remote reset

reset specified nodes.

Routine

f_ifa_reset

Task

m_prompt

REMOTE SHOW

REMOTE SHOW [node] [task] -LOCAL -TASKS -DAQ -NET

PURPOSE

(m_prompt) Shows remote info.

PARAMETERS

node

Node name or @file (one name per line)

task

optional task for ps command

-LOCAL

local node

-TASKS

show task list

-DAQ

show DAQ status and message status

-NET

show net status

See also

SHOW command of program remote
## Description

<table>
<thead>
<tr>
<th>Function</th>
<th>Shows remote info</th>
</tr>
</thead>
<tbody>
<tr>
<td>Routine</td>
<td>f_ifa_show_rem</td>
</tr>
<tr>
<td>Task</td>
<td>m_prompt</td>
</tr>
</tbody>
</table>
REMOVE

REMOVE SEGMENTS

PURPOSE (m_util) Removes critical segments

PARAMETERS

none

Description

Function remove critical segments
Routine f_ut_remove_segments
Task m_util
RESET

RESET

RESET [node] [task] -LOCAL

PURPOSE (m_remote) Resets remote node. Executed by alias remote.

PARAMETERS

node Node name or @file (one name per line)
task optional task to kill. If specied, only this task is killed. DAQ status and message files are NOT cleared. If not specified, the following tasks are killed. m_dispatch m_col_vme m_collector m_esone_serv m_event_serv m_read_cam_slav m_read_meb m_stream_serv m_transport m_util m_msg_log

-LOCAL local node

See also REMOTE RESET command of prompter

Description

Function Kills MBS tasks, creates new message files and clears DAQ status.

Example shell> remote reset -l

Routine f_remote.c

Task m_remote
RESTORE

RESTORE BASE

RESTORE BASE base file -OVERWRITE -NEW

PURPOSE (m_histogram) Restores base from file.

PARAMETERS

base name of segment = name of base
file name of file

-OVERWRITE Overwrite existing base. Size and structure must be the same as current
-NEW Create new base after deleting old

Description

Function Restore base from file.
Routine f_his_cmd_resbas
Task m_histogram

RESTORE POLYGON

RESTORE POLYGON file -NEW -OVER -APPEND

PURPOSE (m_collector) Restore polygon condition from file.

PARAMETERS

file File name

-NEW Remove current polygons, create new.
-OVER
  Overwrite current (there must be enough slots)

-APPEND
  Append to current (there must be enough slots)

Description
Function
  Restore polygon condition.
Routine
  f_poly_cmd_r
Task
  m_collector

RESTORE WINDOW

RESTORE WINDOW file -NEW -OVER -APPEND

PURPOSE
(m_collector) Restore window condition from file.

PARAMETERS
file
  File name
-NEW
  Remove current windows, create new.
-OVER
  Overwrite current (there must be enough slots)
-APPEND
  Append to current (there must be enough slots)

Description
Function
  Restore window condition.
Routine
  f_win_cmd_r
Task
  m_collector
SET

SET ARECEIVER FLUSHTIME

SET ARECEIVER FLUSHTIME time

PURPOSE
(m_ar) Sets stream flushtime

PARAMETERS

time
stream flushtime in seconds

Description

Function
set stream flushtime

Routine
f_dr_set_flush_time

Task
m_ar

SET DISPATCHER

SET DISPATCHER [node]

PURPOSE
(m_prompt) Sets terminal to remote dispatcher.

PARAMETERS

node
Name of remote node. If ommitted use local.
Description

Function
Set prompt to node name, and route terminal IO to/from node through TCP. Note, that for single commands the node can be specified by a node:: prefix, i.e. cvc10::sho task To send commands to all connected nodes, use *::command

Routine f_ifa_set_host

Task m_prompt

SET DR_FLUSHTIME

SET DR_FLUSHTIME time

PURPOSE (m_dr) Sets stream flush time

PARAMETERS

time stream flush time in seconds

Description

Function set stream flush time

Routine f_dr_set_flush_time

Task m_dr

SET EVENT_SERV

SET EVENT_SERV [scale] [events] [maxclnt] -ALL

PURPOSE (m_event_serv) Sets m_event_serv parameters.

PARAMETERS
scale  scaledown streams to analyse. 1: take all
events  maximum number of events to copy from a stream
maxclnt  maximum number of clients
-ALL  take all events per selected stream

Description

Function  Set max events/stream in m_event_serv. This command can be used to reduce the load produced by the event server. The event server processes only the number of events specified per stream and frees the stream.

Function  Set max number of clients on m_event_serv.

Function  Similar to SET EVENTS this command may reduce the load produced by the event server. Specified number of streams is skipped without processing.

Routine  f_ev_cmd_setmaxevt
Task  m_event_serv

SET FILEHEADER

SET FILEHEADER string [line] -RUNID -EXPERIMENT

COMMENT -LABEL -FILENAME -USERNAME -CLEAR

PURPOSE  (m_transport) Specify fileheader information.

PARAMETERS

string  Text according qualifier
-EXPERIMENT  Experiment
-RUNID  Run id.
-COMMENT  Comment line (line=# optionally overwrites line).
-LABEL Label (default)
-FILENAME Filename (default)
-USERNAME Username (default)
-CLEAR disables this info.

Description
Function Specify fileheader information for next 'open file' command. This information can be input interacively by 'open file -prompt'.
Routine f_
Task m_transport

SET FLUSHTIME

SET FLUSHTIME time

PURPOSE (m_collector) Sets stream flush time
PARAMETERS
  time stream flush time in seconds

Description
Function set stream flush time
Routine f_col_set_flush_time
Task m_collector
SET HISTOGRAM CHANNEL

SET HISTOGRAM CHANNEL name value xchan [ychan] -INCREMENT

PURPOSE  (m_histogram) Sets channel content of histogram.
PARAMETERS
  name    histogram name
  value   value or increment
  xchan   x-channel (1st dim)
  ychan   y-channel (2nd dim)
-INCREMENT Add value.

Description
  Function  Set or increment channel content of histogram.
  Routine   f_his_cmd_setcha
  Task      m_histogram

SET HISTOGRAM TEXT

SET HISTOGRAM TEXT name [text] -TITLE -XTXT -YTXT -CONT

PURPOSE  (m_histogram) Sets lettering text field of histogram.
PARAMETERS
  name    histogram name
  text    value or increment
  -TITLE  Lettering on histogram title
-XTXT Lettering on x-axis
-YTXT Lettering on y-axis
-CONT Lettering on data content

Description
Function Sets lettering text field of histogram.
Routine f_his_cmd_settxt
Task m_histogram

SET MAXCLIENTS ESONE_SERV

SET MAXCLIENTS ESONE_SERV maxclnt

PURPOSE (m_esone_serv) Sets maximum number of clients for m_esone_serv.
PARAMETERS
maxclnt maximum number of clients

Description
Function Set max number of clients on m_esone_serv.
Routine f_es_cmd_setmaxclnt
Task m_esone_serv
**SET MAXCLIENTS EVENT_SERV**

```plaintext
SET MAXCLIENTS EVENT_SERV maxclnt
```

**PURPOSE**
(m_event_serv) Sets maximum number of clients for m_event_serv.

**PARAMETERS**
- `maxclnt` maximum number of clients

**Description**

- **Function**
  Set max number of clients on m_event_serv.
- **Routine**
  f_ev_cmd_setmaxclnt
- **Task**
  m_event_serv

**SET PRINT ESONE_SERV**

```plaintext
SET PRINT ESONE_SERV -ON -OFF
```

**PURPOSE**
(m_esone_serv) Sets terminal output Esone data for Esone Server.

**PARAMETERS**
- `-ON` print C N A F on.
- `-OFF` print C N A F off (default)

**Description**

- **Function**
  Set terminal output Esone data for m_esone_serv.
- **Routine**
  f_es_cmd_setprint
- **Task**
  m_esone_serv
### SET RIRECEIVER FLUSHTIME

**SET RIRECEIVER FLUSHTIME time**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>time</td>
<td>stream flushtime in seconds</td>
</tr>
</tbody>
</table>

**Purpose**
(m_rire) Sets stream flushtime

**Description**
- **Function**: set stream flushtime
- **Routine**: f_dr_set_flush_time
- **Task**: m_rire

### SET STREAM_SERV

**SET STREAM_SERV [scale] -[NO]SYNC -[NO]KEEP -SCALED_KEEP -CLEAR**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>scale</td>
<td>scaledown streams to client. 1: send all, when requested. Default after startup is 2!</td>
</tr>
<tr>
<td>-[NO]SYNC</td>
<td>synchronous mode: send all scaled, wait for request.</td>
</tr>
<tr>
<td>-[NO]KEEP</td>
<td>Keep streams when scale=1 and when there are &gt; 3 free streams.</td>
</tr>
<tr>
<td>-SCALED_KEEP</td>
<td>Keep scaled down streams when there are &gt; 3 free streams.</td>
</tr>
<tr>
<td>-CLEAR</td>
<td>Clear counters.</td>
</tr>
</tbody>
</table>

**Remarks**
Mode after startup: SET STREAM 2 -NOKEEP -NOSYNC
Description

Function

This command may reduce the load produced by the stream server. Specified number (scale) of streams is skipped without processing. There are several strategies to avoid the problem that between spills all streams are passed, and the clients do not get any more streams, even if both, client and mbs, are idle. The general mechanism is that the server checks if there are enough (3) free streams. If yes it yields and gives the client the chance to request the stream. Between spills the client would then get all streams, but also, if the streams are always free (slow data rate). This behavior is modified in two variants: When -KEEP is used, free streams are only checked, if scale=1. Otherwise no streams are kept and the client will not get streams between spills. When -SCALE is used, for any nth stream the number of free streams is checked. This is a compromise to give the client the chance to get streams between spills, but still use scaling for not slowing down the acquisition. Which method is the best depends very much on the special situation. One should play with the parameters to get the desired behavior (transfer rate against DAQ slow down). The default mode after startup is SET STREAM 2 -NOKEEP -NOSYNC

REMARK

-SYNC switches off any keep mechanism.

NOTE

Arguments and switches not specified are not changed.

Routine

f_str_setstr

Task

m_stream_serv

SET TASK

SET TASK task pid -CLEAR

PURPOSE

(m_dispatch) Sets task id.

PARAMETERS

<table>
<thead>
<tr>
<th>task</th>
<th>Task (program) name</th>
</tr>
</thead>
<tbody>
<tr>
<td>pid</td>
<td>pid of task. Normally tasks write their pid in s_daqst control structure and m_dispatch reads it from there. If not, SET TASK can be used to set the pid which is used for command channels.</td>
</tr>
</tbody>
</table>
**-CLEAR**

Removes tasks from the command list which are marked not active in the daq status table. NOTE: This qualifier MUST NOT given as long as there are tasks starting! It should be used when tasks abort abnormally.

**Description**

**Function**

This command is not needed under normal conditions. It could be used, if the task is known to the dispatcher but is started elsewhere. Then its pid can be set in the dispatcher and the task accepts command from dispatcher. The task MUST NOT, however, stopped by STOP TASK command.

**Routine**

f_disp_set_tsk

**Task**

m_dispatch

---

**SET TORECEIVER FLUSHTIME**

**SET TORECEIVER FLUSHTIME time**

**PURPOSE**

(m_to) set stream flush time

**PARAMETERS**

* time
  
  flush time in seconds, 0 = no flushing

**Description**

**Function**

set stream flush time

**Routine**

f_to_set_flush_time

**Task**

m_to
SET TRIG_MOD

**SET TRIG_MOD -SLAVE**

**PURPOSE**
(m_util) Sets trigger module.

**PARAMETERS**

- **-SLAVE**
  Trigger module is set to slave.

- **-MULTI**
  Master Trigger module is part of a system which utilizes the trigger bus

**Description**

**Function**
Set the trigger module parameters.

**Routine**
f_ut_op_trig_mod

**Task**
m_util

SET USER VALUE

**SET USER VALUE index values -clear -show**

**PURPOSE**
(m_util) Set user values in daqst.

**PARAMETERS**

- **index**
  0..15: index in array.

- **value**
  value array: n or (n1,n2,...)

- **-clear**
  clear all values first

- **-show**
  show all values
Description

Function    Set user values in daqst.
Routine     f_ut_set_user_val
Task        m_util

SET VERBOSE DISPATCHER

SET VERBOSE DISPATCHER -ON -OFF

PURPOSE    (m_dispatch) Sets verbosity for dispatcher.
PARAMETERS
-ON         verbosity on.
-OFF        verbosity off (default)

Description

Function    Set verbosity of messages for dispatcher.
Routine     fDisp_Verbose
Task        m_dispatch

SET VERBOSE ESONE_SERV

SET VERBOSE ESONE_SERV -ON -OFF

PURPOSE    (m_esone_serv) Sets verbosity for Esone Server.
PARAMETERS
-ON         verbosity on.
-OFF        verbosity off (default)
Description

Function: Set verbosity of messages for m_event_serv.
Routine: f_es_cmd_setverb
Task: m_esone_serv

SET VERBOSE EVENT_SERV

SET VERBOSE EVENT_SERV -ON -OFF

PURPOSE: (m_event_serv) Sets verbosity for m_event_serv.
PARAMETERS
-ON: verbosity on.
-OFF: verbosity off (default)

Description

Function: Set verbosity of messages for m_event_serv.
Routine: f_ev_cmd_setverb
Task: m_event_serv

SET VERBOSE GLOBAL

SET VERBOSE GLOBAL -ON -OFF -NEUTRAL

PURPOSE: (m_util) Sets verbosity for all tasks
PARAMETERS
-ON: verbosity on.
GSI Multi-Branch System Reference Manual: MBS Command Description

- **OFF**  
  verbosity off (default)

- **NEUTRAL**  
  verbosity is like set in specific task

**Description**

**Function**  
Set verbosity of all tasks

**Routine**  
`f_ut_set_gleob_vebose`

**Task**  
`m_util`

---

**SET VERBOSE HISTOGRAM**

**SET VERBOSE HISTOGRAM -ON -OFF**

**PURPOSE**  
(m_histogram) Sets verbosity for m_histogram.

**PARAMETERS**

- **-ON**  
  verbosity on.

- **-OFF**  
  verbosity off (default)

**Description**

**Function**  
Set verbosity of messages for m_histogram.

**Routine**  
`f_his.cmd_setverb`

**Task**  
`m_histogram`
SET VERBOSE PROMPT

SET VERBOSE PROMPT -ON -OFF

PURPOSE  (m_prompt) Sets verbosity for m_prompt.

PARAMETERS

-ON        verbosity on.
-OFF       verbosity off (default)

Description

Function    Set verbosity of messages for m_prompt.
Routine     f_prVerbose
Task        m_prompt

SET WINDOW

SET WINDOW name [x1 x2 y1 y2] -CURSOR

PURPOSE  (m_collector) Set limits of window condition.

PARAMETERS

name        name of condition
x1,x2,y1,y2  values
-CURSOR     Take values from the last cursor input
Description

Function       Set limits of window condition.
Routine        f_win_cmd_s
Task           m_collector

SET XDISPLAY

SET XDISPLAY node

PURPOSE        (m_prompt) Set name of remote display.
PARAMETERS
node           Node where a remote display runs.

Description

Function       Set name of remote display.
Routine        f_pr_set_disp
Task           m_prompt
### SHOW

**SHOW**

SHOW [node] [task] -LOCAL -TASKS -DAQ -NET

<table>
<thead>
<tr>
<th>PURPOSE</th>
<th>(m_remote) Shows remote info. Executed by alias remote.</th>
</tr>
</thead>
<tbody>
<tr>
<td>PARAMETERS</td>
<td></td>
</tr>
<tr>
<td>node</td>
<td>Node name or @file (one name per line)</td>
</tr>
<tr>
<td>task</td>
<td>optional task for ps command</td>
</tr>
<tr>
<td>-LOCAL</td>
<td>local node</td>
</tr>
<tr>
<td>-TASKS</td>
<td>show task list</td>
</tr>
<tr>
<td>-DAQ</td>
<td>show DAQ status and message status</td>
</tr>
<tr>
<td>-NET</td>
<td>show net status</td>
</tr>
<tr>
<td>See also</td>
<td>REMOTE SHOW command of prompter</td>
</tr>
</tbody>
</table>

**Description**

**Function**

Shows remote info.

**Example**

shell> remote show -l

**Routine**

f\_remote\_s

**Task**

m\_remote
SHOW ACQUISITION

SHOW ACQUISITION [seconds] -SETUP -CRATES -SERVER -RATE -LOG

PURPOSE
(m_util) Shows acquisition.

PARAMETERS

seconds    wait n seconds to show rates (def=1).

-SETUP      Show setup

-CRATES     Show crates

-SERVER     Show data transfers

-RATE       Show data rates

-LOG        write output also to log file

Description

Function      Show acquisition. When no switch is given, all are output.

Routine       f_ut_show_acq

Task          m_util

SHOW BASE

SHOW BASE [name] -FULL

PURPOSE
(m_histogram) Shows information about histogram data base.

PARAMETERS

name         name of segment
SHOW

Description

<table>
<thead>
<tr>
<th>Function</th>
<th>Show information about histogram data base.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Routine</td>
<td>f_his_cmd_shobas</td>
</tr>
<tr>
<td>Task</td>
<td>m_histogram</td>
</tr>
</tbody>
</table>

SHOW COMMANDS

SHOW COMMANDS [task] -FULL -ALL

PURPOSE  (m_dispatch) Shows known commands.

PARAMETERS

- **task**: Optional task name to show commands
- **-FULL**: Commands with arguments.
- **-ALL**: All commands with arguments

Description

<table>
<thead>
<tr>
<th>Function</th>
<th>Show all known commands (including the commands of inactive tasks by -ALL).</th>
</tr>
</thead>
<tbody>
<tr>
<td>Routine</td>
<td>f_disp_sho_cmd</td>
</tr>
<tr>
<td>Task</td>
<td>m_dispatch</td>
</tr>
</tbody>
</table>

SHOW CVC_IRQ_MASK

PURPOSE  (m_util) Reads irq mask of the CVC irq controller
PARAMETERS

none

Description

Function Read irq mask of the CVC irq controller
Routine f_ut_cvc_cam_irq
Task m_util

SHOW DISPATCHER

SHOW DISPATCHER [node]

PURPOSE (m_prompt) Shows connections to remote dispatchers.

PARAMETERS

node Name of remote node.

Description

Function Show connections to remote dispatchers.
Routine f_ifa_show_connect
Task m_prompt

SHOW ENVIRONMENT

SHOW ENVIRONMENT

PURPOSE (m_dispatch) Shows environment parameters.
PARAMETERS

Description

Function: Show environment parameters of dispatcher.
Routine: f_disp_sho_env
Task: m_dispatch

SHOW ESONE_SERV

SHOW ESONE_SERV -FULL -LOG -CLIENT

PURPOSE: (m_esone_serv) Shows status of m_esone_serv.

PARAMETERS

-FULL: full output
-LOG: output to log file
-CLIENT: sho status of each client

Description

Function: Show status of m_event_serv.
Routine: f_es_cmd_showsts
Task: m_esone_serv
SHOW EVENT_SERV

SHOW EVENT_SERV -FULL -LOG -CLIENT

PURPOSE (m_event_serv) Shows status of m_event_serv.

PARAMETERS
   -FULL full output
   -LOG output to log file
   -CLIENT show client info

Description
   Function Show status of m_event_serv.
   Routine f_ev_cmd_showsts
   Task m_event_serv

SHOW FILE

SHOW FILE [tape]

PURPOSE (m_transport) Shows file output information.

PARAMETERS
   tape Optional tape number
Description

Function Shows current tape device, current tape label, file name and number of kBytes written already to tape and/or file.

Routine f_show_tape

Task m_transport

SHOW HISTOGRAM

SHOW HISTOGRAM name [1offset] [1channels] [2offset] [2channels] -DATA -FULL

PURPOSE (m_histogram) Shows histogram info and content.

PARAMETERS

name name of histogram, also wildcard.
-DATA show histogram data
-FULL show full histogram parameters

Description

Function Show histogram info and content.

Routine f_his_cmd_shohis

Task m_histogram

SHOW INPUT NODES

SHOW INPUT NODES

PURPOSE (m_to) show input nodes connected via tcp sockets
PARAMETERS

Description

Function  stops time sorting and event output
Routine    f_to_show_input_nodes
Task       m_to

SHOW MESSAGE

PURPOSE  (m_dispatch) Shows status of internal message file.
PARAMETERS

Description

Function  There are two message files: One for commands and one for messages. These files are used for intertask communication.
Routine   f_disp_sho_mes
Task      m_dispatch

SHOW ML_SETUP

PURPOSE  (m_util) Shows multilayer setup parameters
PARAMETERS
none

Description
Function  parameters from memory
Routine    f_ut_print_ml_setup
Task       m_util

SHOW POLYGON

SHOW POLYGON name -DATA

PURPOSE  (m_collector) Show polygon conditions.
PARAMETERS
name     name of condition

Description
Function  Show polygon conditions.
Routine    f_poly_cmd_p
Task       m_collector

SHOW RATE

SHOW RATE [seconds] -OFF -ON

PURPOSE  (m_util) Shows acquisition rate.
PARAMETERS
seconds          If 0, show counters. If > 0, wait n seconds and show rates. Also sets interval for rate program.
-OFF            switches rate program off
-ON             switches rate program on

Description
Function           Show counters and rates.
Routine            f_ut_rate
Task               m_util

SHOW SETUP

PURPOSE          (m_util) Shows setup parameters
PARAMETERS        none

Description
Function            SHOW SETUP parameters from memory
Routine             f_ut_print_setup
Task                m_util
SHOW STATUS

PURPOSE (m_util) Shows daqst parameters

PARAMETERS

none

Description

Function daqst parameters from memory
Routine f_ut_print_daqst
Task m_util

SHOW STREAM_SERV

SHOW STREAM_SERV -CLEAR

PURPOSE (m_stream_serv) Shows modes and counters.

PARAMETERS

-CLEAR Clear counters.

Description

Function This command shows counters and modes of the stream server. The counters are incremented only if a client is connected.
Routine f_str_shostr
Task m_stream_serv
SHOW TAPE

SHOW TAPE [tape]

PURPOSE (m_transport) Shows tape information.

PARAMETERS

  tape    Optional tape number

Description

  Function    Shows current tape device, current tape label, file name and number
               of kBytes written already to tape.
  Routine     f_show_tape
  Task        m_transport

SHOW TASK

SHOW TASK [task] -FULL -ALL

PURPOSE (m_dispatch) Shows known tasks.

PARAMETERS

  task    Optional task (program) name. When specified, show commands of this
          task.
  -FULL   Show commands with arguments
  -ALL    Show also inactive tasks
**SHOW**

**Description**

**Function**

Outputs list of known tasks. When the task is marked "running" it accepts commands and could be terminated by command STOP TASK. Otherwise it is not started or runs "detached", i.e. does not execute commands.

**Routine**

f_disp_sho_tsk

**Task**

m_dispatch

---

**SHOW TRIG_MOD**

**PURPOSE**

(m_util) Shows current setup of trigger module.

**PARAMETERS**

none

---

**Description**

**Function**

returns current setup of trigger module.

**Routine**

f_ut_op_trig_mod

**Task**

m_util

---

**SHOW WINDOW**

**PURPOSE**

(m_collector) Show window conditions.

**PARAMETERS**

name name of condition
### Description

<table>
<thead>
<tr>
<th><strong>Function</strong></th>
<th>Show window conditions.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Routine</strong></td>
<td>f_win_cmd_p</td>
</tr>
<tr>
<td><strong>Task</strong></td>
<td>m_collector</td>
</tr>
</tbody>
</table>
START

START ACQUISITION

START ACQUISITION

**PURPOSE**  (m_util) Starts acquisition.

**PARAMETERS**

xxx

Description

**Function**  This command assumes that the trigger module is setup properly (by command SET TRIG.MOD or after STOP ACQUIS). Readout tables must be loaded. A software trigger number 14 is delivered and the deadtime is reset. The readout tasks get the trigger 14 as first event. They may do specific actions, collect some data and write a start event.

**Routine**  f_ut_op_trig_mod

**Task**  m_util

START ARECEIVER

START ARECEIVER

**PURPOSE**  (m_ar) Enables data receive of a ar node

**PARAMETERS**
Description

Function Enables data receive of a ar node
Routine f_dr_cmd_ena_receive_data
Task m_ar

START EVENT_SERV

START EVENT_SERV [scale] [events] [maxclnt]

[NO]PORTS -VERB

PURPOSE (m_event_serv) Starts m_event_serv.

PARAMETERS

scale scaledown streams to analyse. 1: take all
events maximum number of events to copy from a stream.
maxclnt max number of clients
-NOPORTS do not send information to portserver. (default)
-PORTS send information to portserver.
-VERB verbosity on.

Description

Function abc
Routine f_ev_cmd_start
Task m_event_serv
START MESSAGE

PURPOSE (m_dispatch) Starts the message logger.

Description

Function The message logger is required before any other task can be started or any command can be executed.
Routine f_disp_sta_msg
Task m_dispatch

START OUTPUT

PURPOSE (m_to) starts time sorting and event output

PARAMETERS

Description

Function starts time sorting and event output
Routine f_to_start_data_out
Task m_to
START RIRECEIVER

**PURPOSE**

(m_irec) Enables data receive of a rirec node

**PARAMETERS**

**Description**

- **Function**: Enables data receive of a rirec node
- **Routine**: `f_dr_cmd_ena_receive_data`
- **Task**: `m_irec`

START TASK

**PURPOSE**

(m_dispatch) Starts task.

**PARAMETERS**

- **task**: Task (program) name. Optionally with path.
  - Default path is `/MBSROOT/bin_HOSTTYPE`.
- **file**: Optional path

**Description**

- **Function**: Start a task (program) by forking. When the task is known to the dispatcher through command definitions, the program is forked directly and the status is set to running. These tasks can be terminated by command STOP TASK. When the task is not known, it is forked through an intermediate task which terminates. The new task is entered to the
task list, but not marked running but detached thus indicating that it
does not accept commands. These tasks can be stopped by command
STOP TASK dummy pid -KILL. It is assumed that they exit when their
job is done.

<table>
<thead>
<tr>
<th>Routine</th>
<th>f_disp_sta_tsk</th>
</tr>
</thead>
<tbody>
<tr>
<td>Task</td>
<td>m_dispatch</td>
</tr>
</tbody>
</table>
STOP

STOP ACQUISITION

PURPOSE (m_util) Stops acquisition.

PARAMETERS
none

Description

Function This command delivers software trigger 15. Then the readout tasks recognise this event as last event. The master readout keeps the dead-time thus blocking further triggers. The collector frees the current buffer stream when event type 15 occurs. The status of the system is set to STOP PENDING until the transport acknowledges the last event and finishes the acquisition stop.

Routine f_ut_op_trig_mod

Task m_util

STOP ARECEIVER

PURPOSE (m_ar) Disables data receive of a ar node

PARAMETERS
Description

**Function**
Disables data receive of a ar node

**Routine**
f_dr_cmd_dis_receive_data

**Task**
m_ar

STOP OUTPUT

**PURPOSE**
(m_to) stops time sorting and event output

**PARAMETERS**

Description

**Function**
stops time sorting and event output

**Routine**
f_to_stop_data_out

**Task**
m_to

STOP RIRECEIVER

**PURPOSE**
(m_rirec) Disables data receive of a rirec node

**PARAMETERS**
Description

Function
Disables data receive of a rirec node

Routine
f_dr.cmd.dis_receive_data

Task
m_rirecc

STOP TASK

STOP TASK [task] [pid] -ALL -KILL -ZOMBIE

PURPOSE
(m_dispatch) Stops task by sending command "exit"

PARAMETERS

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>task</td>
<td>Task (program) name</td>
</tr>
<tr>
<td>pid</td>
<td>Optional pid used with -KILL if task not known.</td>
</tr>
<tr>
<td>-ALL</td>
<td>Stop all known tasks except message logger.</td>
</tr>
<tr>
<td>-KILL</td>
<td>Stop task by calling kill function with signal SIGTERM.</td>
</tr>
<tr>
<td>-ZOMBIE</td>
<td>Remove status blocks of terminated tasks to let them die (Z to RIP).</td>
</tr>
</tbody>
</table>

Description

Function
Without -KILL: Sends command "exit" to the task. The task command thread exits and the main program exits. Only tasks running commands can be terminated by this way. Other tasks can only be stopped with -KILL.

zombies
When a task terminates by itself, it goes into status Z(ombie), because the father must remove the termination status block. With the -Z switch the dispatcher looks through the task table and tries to remove pending termination status blocks. This is only possible for childs. When the dispatcher has been quit, all childs get the init task as father. Now the restarted dispatcher cannot remove their status block anymore.

Routine
f_disp_sto_tsk
Task: m\_dispatch
TYPE

TYPE EVENT

TYPE EVENT [events id control crate offset items] -SAMPLE -VERBOSE -DECIMAL -BUFFER

PURPOSE (m_transport) Prints events.

PARAMETERS

- events number of events to print, default = 1
- id print only subevents with this procid
- control print only subevents with this controller type
- crate print only subevents with this subcrate
- offset print for each subevent data from offset on
- items print for each subevent only this number of items
- SAMPLE print only first events of every buffer
- VERBOSE print also event data
- DECIMAL default is hex output (longwords)
- BUFFER print buffer headers

Description

Function Prints event headers and event data
Routine f_
Task m_transport
TYPE STREAM

TYPE STREAM buffers -VERBOSE -HEADER

PURPOSE (m_transport) Prints buffers of stream.

PARAMETERS

buffers number of buffers to type
-VERBOSE print also element data (including headers)
-HEADER print also element headers

Description

Function Prints buffer headers and buffer data
Routine f_
Task m_transport
VOID

VOID

VOID

PURPOSE   (m\_daq\_rate) place holder, do not execute.

PARAMETERS

Description

Function   This command must not be executed.

Routine

Task       m\_daq\_rate
VRA16D16

VRA16D16

VRA16D16 addr [size repeat]

LOG -NOPRINT -FC

PURPOSE

(m_vme_serv) VME read A16 D16

PARAMETERS

addr
size
repeat
-LOG
-NOPRINT
-FC

VME address of slave module to read from
size of data to read (in bytes)
repetition count of VME read
log result in mbs log file
don’t print result on terminal
if RIO2 or RIO3 use find_controller function instead of static VME mapping

Description

Function
Routine
Task

VME read A16 D16
f_vs_vme_read
m_vme_serv
VRA24D16

VRA24D16

VRA24D16 addr [size repeat]

LOG -NOPRINT -FC

PURPOSE

(m_vme_serv) VME read A24 D16

PARAMETERS

addr
VME address of slave module to read from

size
size of data to read (in bytes)

repeat
repetition count of VME read

-LOG
log result in mbs log file

-NOPRINT
don’t print result on terminal

-FC
if RIO2 or RIO3 use find_controller function instead of static VME mapping

Description

Function
VME read A24 D16

Routine
f_vs_vme_read

Task
m_vme_serv
VRA24D32

VRA24D32

VRA24D32 addr [size repeat]

LOG -NOPRINT -FC

PURPOSE  (m_vme_serv) VME read A24 D32

PARAMETERS

addr  VME address of slave module to read from
size  size of data to read (in bytes)
repeat  repetition count of VME read
-LOG  log result in mbs log file
-NOPRINT  don’t print result on terminal
-FC  if RIO2 or RIO3 use find_controller function instead of static VME mapping

Description

Function  VME read A24 D32
Routine  f_vs_vme_read
Task  m_vme_serv
VRA32D16

VRA32D16

VRA32D16 addr [size repeat]

LOG -NOPRINT -FC

PURPOSE

(m_vme_serv) VME read A32 D16

PARAMETERS

addr VME address of slave module to read from
size size of data to read (in bytes)
repeat repetition count of VME read

-LOG log result in mbs log file
-NOPRINT don’t print result on terminal
-FC if RIO2 or RIO3 use find_controller function instead of static VME mapping

Description

Function VME read A32 D16
Routine f_vs_vme_read
Task m_vme_serv
VRA32D32

VRA32D32

VRA32D32 addr [size repeat]

LOG -NOPRINT -FC

PURPOSE

(m_vme_serv) VME read A32 D32

PARAMETERS

addr VME address of slave module to read from
size size of data to read (in bytes)
repeat repetition count of VME read
-LOG log result in mbs log file
-NOPRINT don’t print result on terminal
-FC if RIO2 or RIO3 use find_controller function instead of static VME mapping

Description

Function VME read A32 D32
Routine f_vs_vme_read
Task m_vme_serv
VWA16D16

VWA16D16

VWA16D16 addr data [size repeat]

LOG -NOPRINT -FC -INC

PURPOSE (m_vme_serv) VME write A16 D16

PARAMETERS

addr VME address of slave module to write to
data data to write
size size of data to write (in bytes)
repeat repetition count of VME write
-LOG log result in mbs log file
-NOPRINT don’t print result on terminal
-FC if RIO2 or RIO3 use find_controller function instead of static VME mapping
-INC increment data by 1 per write cycle

Description

Function VME write A16 D16
Routine f_vs_vme_write
Task m_vme_serv
VWA24D16

VWA24D16

VWA24D16 addr data [size repeat]

LOG -NOPRINT -FC -INC

PURPOSE  (m_vme_serv) VME write A24 D16

PARAMETERS

- addr  VME address of slave module to write to
- data  data to write
- size  size of data to write (in bytes)
- repeat  repetition count of VME write
- -LOG  log result in mbs log file
- -NOPRINT  don’t print result on terminal
- -FC  if RIO2 or RIO3 use find_controller function instead of static VME mapping
- -INC  increment data by 1 per write cycle

Description

Function  VME write A24 D16
Routine  f_vs_vme_write
Task  m_vme_serv
VWA24D32

VWA24D32

VWA24D32 addr data [size repeat]

LOG -NOPRINT -FC -INC

PURPOSE (m_vme_serv) VME write A24 D32

PARAMETERS

addr VME address of slave module to write to

data data to write

size size of data to write (in bytes)

repeat repetition count of VME write

-LOG log result in mbs log file

-NOPRINT don’t print result on terminal

-FC if RIO2 or RIO3 use find_controller function instead of static VME mapping

-INC increment data by 1 per write cycle

Description

Function VME write A24 D32

Routine f_vs_vme_write

Task m_vme_serv
VWA32D16

VWA32D16 addr data [size repeat]

LOG -NOPRINT -FC -INC

PURPOSE (m_vme_serv) VME write A32 D16

PARAMETERS

addr VME address of slave module to write to
data data to write
size size of data to write (in bytes)
repeat repetition count of VME write
-LOG log result in mbs log file
-NOPRINT don’t print result on terminal
-FC if RIO2 or RIO3 use find_controller function instead of static VME mapping
-INC increment data by 1 per write cycle

Description

Function VME write A32 D16
Routine f_vme_write
Task m_vme_serv
VWA32D32

VWA32D32

VWA32D32 addr data [size repeat]

LOG  -NOPRINT  -FC  -INC

Purpose

(m_vme_serv) VME write A32 D32

Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>addr</td>
<td>VME address of slave module to write to</td>
</tr>
<tr>
<td>data</td>
<td>data to write</td>
</tr>
<tr>
<td>size</td>
<td>size of data to write (in bytes)</td>
</tr>
<tr>
<td>repeat</td>
<td>repetition count of VME write</td>
</tr>
<tr>
<td>-LOG</td>
<td>log result in mbs log file</td>
</tr>
<tr>
<td>-NOPRINT</td>
<td>don’t print result on terminal</td>
</tr>
<tr>
<td>-FC</td>
<td>if RIO2 or RIO3 use find_controller function instead of static VME mapping</td>
</tr>
<tr>
<td>-INC</td>
<td>increment data by 1 per write cycle</td>
</tr>
</tbody>
</table>

Description

Function   VME write A32 D32
Routine    f_vs_vme_write
Task       m_vme_serv
X SHELL

X SHELL

X SHELL command [a1 a2 a3 a4]

PURPOSE  (m_dispatch) Executes shell command line.

PARAMETERS

command  Command to be executed.

a#  Further arguments

Description

Function  Use function system() to fork a command.
Example  X ps -af
Routine  f_disp_shell
Task  m_dispatch
## Contents

1 Preface ........................................................................................................................................1
    1.1 MBS Authors and Advisory Service ......................................................................................2

2 MBS Keyword Summary ............................................................................................................3
    MBS keywords ..........................................................................................................................4

3 MBS Command Summary .........................................................................................................23

4 MBS Command Description ......................................................................................................33
    ATTACH ....................................................................................................................................34
    CAMAC .....................................................................................................................................35
    CLEAR ......................................................................................................................................37
    CLOSE ......................................................................................................................................42
    CNAF ........................................................................................................................................43
    COMMENT .................................................................................................................................44
    CONNECT ..................................................................................................................................45
    CREATE ......................................................................................................................................48
    DEFINE ......................................................................................................................................54
    DELETE ......................................................................................................................................55
    DETACH .....................................................................................................................................57
    DISABLE ....................................................................................................................................58
    DISCONNECT ..............................................................................................................................63
    DISMOUNT ................................................................................................................................66
    DUMP .........................................................................................................................................67
    ENABLE ......................................................................................................................................69
    HELP ........................................................................................................................................75
    INITIALIZE ...............................................................................................................................76
    LOAD .........................................................................................................................................77
    MOUNT .......................................................................................................................................81
    NEWS .........................................................................................................................................82
    ON ............................................................................................................................................83
<table>
<thead>
<tr>
<th>Command</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>OPEN</td>
<td>84</td>
</tr>
<tr>
<td>PROTECT</td>
<td>86</td>
</tr>
<tr>
<td>PSHELL</td>
<td>87</td>
</tr>
<tr>
<td>REMOTE</td>
<td>88</td>
</tr>
<tr>
<td>REMOVE</td>
<td>91</td>
</tr>
<tr>
<td>RESET</td>
<td>92</td>
</tr>
<tr>
<td>RESTORE</td>
<td>93</td>
</tr>
<tr>
<td>SET</td>
<td>95</td>
</tr>
<tr>
<td>SHOW</td>
<td>111</td>
</tr>
<tr>
<td>START</td>
<td>125</td>
</tr>
<tr>
<td>STOP</td>
<td>130</td>
</tr>
<tr>
<td>TYPE</td>
<td>134</td>
</tr>
<tr>
<td>VOID</td>
<td>136</td>
</tr>
<tr>
<td>VRA16D16</td>
<td>137</td>
</tr>
<tr>
<td>VRA24D16</td>
<td>138</td>
</tr>
<tr>
<td>VRA24D32</td>
<td>139</td>
</tr>
<tr>
<td>VRA32D16</td>
<td>140</td>
</tr>
<tr>
<td>VRA32D32</td>
<td>141</td>
</tr>
<tr>
<td>VWA16D16</td>
<td>142</td>
</tr>
<tr>
<td>VWA24D16</td>
<td>143</td>
</tr>
<tr>
<td>VWA24D32</td>
<td>144</td>
</tr>
<tr>
<td>VWA32D16</td>
<td>145</td>
</tr>
<tr>
<td>VWA32D32</td>
<td>146</td>
</tr>
<tr>
<td>XSHELL</td>
<td>147</td>
</tr>
</tbody>
</table>