# Mark5B disk data format

Taken from Mark5B System User's Manual (9 June 2006) Section 5.

Data on disk is divided into equal-length disk frames (DF). Each DF has a header of 4 32-bit words followed by 2500 32-bit words of data (10000 bytes). The DF boundary is aligned with the UT second tick.

## Disk Frame Header

| Word 0 | Synchronization word (*0xABADDEED*) | |
|---|---|---|
| Word 1 | Bits 31-16: User specifed | |
| | Bit 15: T - tvg data (test data if set) | |
| | Bits 14-0: DF # within second | |
| Word 2-3 | VLBA BCD Time code and 16 bit CRC | |

| Sync word (32 bit) | | |
|---|---|---|
| User-specifed (16 bits) | T | Frame# within second (starting at 0) |
| VLBA BCD Time code word 1 ('JJJSSSSS') | | |
| VLBA BCD Time code word 2 (16 bit) ('.SSSS') | CRC 16 bit | |

### Disk Data Format

Raw VLBI data is packed within a 32 bit word, with the earliest time sample corresponding to the least significant bit.

# Mark5 control commands

## Mark5 Control commands

The Mark5A protocol uses two TCP ports:

- m5data 2630/tcp # Mark5 data
- m5drive 2620/tcp # Mark5 control

Commands are sent to the **m5drive** port as newline terminated strings. The replies are newline terminated also.

Data is sent to the **m5data** port.

## Mark5 commands

## play_rate=data:<rate>

Used to set the output data rate of the Mark5 at the station.

Expected response:

[=!=]play_rate = 0 ;

## play=off

Used to make the Mark5 stop playing if it was playing back data from disk.

## net_protocol=<protocol>:<sockbuf size>:<workbuf size>

Used to set the network data-transport protocol. The <sockbuf size> argument is the socket send buffer size. You should probably use this value in a setsockopt(…, SOL_SOCKET, SO_SBDBUF, …) call on the socket used to send the data.

Expected response:

[=!=]net_protocol = 0 ;

## mtu=<mtu size>

Set the mtu for UDP transfers

Expected response:

[=!=] !mtu = 0 : $mtu bytes;

## in2net=connect:<ip>

Used to make the Mark5 at the station connect to the Mark5 at JIVE. The <ip> argument will be the IP address of the Mark5 at JIVE, which should at that point be ready to accept a connection from the Mark5 at the station. The connection should be made to the m5data port. No data will be sent yet.

Expected response:

[=!=]in2net = 0 ;

## in2net=on

Used to make the Mark5 at the station start sending data.

Expected response:

[=!=]in2net = 1 ;

## in2net=disconnect

Used to make the Mark5 at the station stop sending data and terminate the connection.

Expected response:

[=!=]in2net = 0 ;

## in2net=off

??

Expected response:

[=!=]in2net = 0 ;

## mode=<data mode>:<data submode>

Used to set the playback mode of the Mark5 at the station.

Expected response:

[=!=]mode = 0 ;

## status?

Used to query the status of the (remote) Mark5.

Expected response:

[=!=]status? 0 : 0x00000001 ; (when not sending data)
[=!=]status? 0 : 0x00010001 ; (when sending data)

## play?

Request play state (disk only??)

Expected response:

[=!=]status? 0 : 0x00000001 ; (when not sending data)
[=!=]status? 0 : 0x00010001 ; (when sending data)

# clock_set=<rate>:int:<rate>

To set the Mark5b clock rate to "rate"

[=!=]clock_set = 0 ;

# Setting TVG test

To setup the Mark5 to send the TVG test vector, run:

mode=tvg:8 play_rate=data:4

This will send 8 tracks at 4 Mbps, ie a total of 32 Mbps.

From:
https://www.atnf.csiro.au/vlbi/dokuwiki/ - **ATNF VLBI Wiki**

Permanent link:
**https://www.atnf.csiro.au/vlbi/dokuwiki/doku.php/lbaops/mark5evlbi**

Last update: **2015/12/18 16:39**