# eSMA SOFTWARE INTERFACE

**Mark Bentum**
**Version 0.3**
**August 10, 2005**

In this document the software issues of the eSMA will be discussed. The eSMA, the extended Submillimeter Array, is a visiting instrument of the JCMT. Its goal is to connect both the SMA and the CSO, to get an interferometer instrument with 10 antennas in total.

The control of all the antennas, and so of the entire eSMA, will be done out of the SMA control room. The JCMT will act as one of the antennas of the array. That means that only a small set of commands will be needed for the eSMA mode. For visiting and new instruments a simple ASCII interface has been created, that can either be connected over a serial line or to a TCP/IP port. This Telescope Control System serial line interface is described in a separate document, "Description of the JAC Telescope Control System serial line interface," by Nick Rees and Russell Kackley.

# 1. Description of the control software

## 1.1 Pointing the telescope

The first concern for a tracking interferometer is the pointing of the dish. This requires a set of commands from the SMA about the source information. First a target source must be defined and the telescope must slew to the target:

```
SET_TARGET 'NGC6251' 'B1950' 4.33772497 1.44322245 0 0 1950 0 0 0 0 'Galaxy' 0 0 0 <CR>
SLEW 'MAIN'<CR>
```

All these commands are available. Other commands available:

| | |
|---|---|
| GET AIRMASS: | Get the airmass at the current telescope position |
| GET DEMAND: | Get the current telescope demand tracking coordinates |
| GET GUIDING: | Return autoguiding status |
| GET IMAGE SCALE: | Return the image scale at the current focal plane |
| GET OBSERVATORY: | Get observatory information |
| GET OFFSETS: | Get offsets from base in gnomonic coordinates. |
| GET STATE: | Get the telescope STATE values |
| GET SYSTEM: | Get tracking coordinate system |
| GET TARGET: | Get the coordinates of a target object |
| GET TEL BASE: | Get the virtual telescope base position |
| GET TIME: | Get various sorts of time |
| NOD: | Nod telescope to selected chopper beam |
| AOFFSET: | Offset the aperture from base |
| OFFSET: | Offset the main and/or guide telescopes or the aperture from base |
| TOFFSET: | Offset the main telescope from base |
| XOFFSET: | Offset the guide telescope from base |
| SET BASE HERE: | Set the virtual telescope base position to its current position |
| SET GUIDING: | Turn autoguiding on/off |
| SET TARGET: | Specify the coordinates of a target object |
| SLEW: | Slew telescope to the next target |

### 1.2. Setting the receiver

Several parameters need to be send to the receiver (RxA3 at the JCMT at this moment, later on HARP). Although the use of the eSMA will be limited to one receiver at the time (some hardware modifications must be made before eSMA observations can be done) the software function will be general. In the tests up till now, these parameters were set manually. The following list of parameters is required:

a) Receiver (char) - Possible receivers at the JCMT: 'RxA3', 'HARP', 'RxB', 'RxW'
b) Sky frequency (double)
c) IF center frequency (double) – In normal mode the IF center frequency will be 4000 MHz. However, in eSMA mode only the IF-band between 4 and 5 GHz is used, so the IF center frequency should be changed in 4500 MHz. Since the SMA is in control of the eSMA, this value should be set by the SMA control software.
d) Sideband (char : either UPPER or LOWER)

This function is not available yet and should be created. The following command description will be used:

SET_RECEIVER – **Set parameters for one of the receivers at the JCMT**

**Function:** Set the parameters for the selected receiver

**Description:** Set the parameters for the receiver for observation

**Arguments:** SET_RECEIVER expects the following arguments to be supplied on the command line in the order shown:

| | | |
|---|---|---|
| 1. RECEIVER | char | One of the JCMT receivers: 'RxA3', 'HARP', 'RxB', 'RxW' |
| 2. SKYFR | double | Sky Frequency in GHz |
| 2. IFCFR | double | IF center frequency in GHz |
| 3. SIDEBD | char | 'UPPER' or 'LOWER' |

**Returned Arguments:** SET_RECEIVER returns a space-delimited string containing the following quantities in the order shown. The string is terminated with a <CR>.

| | | |
|---|---|---|
| 1. STATUS | Int | The Status value (1 if OK, 0 if not OK) |

If the RxA3 is selected, it gets these parameters through the VAX. It will automatically tune to the correct LO value. This includes the micrometer settings of the Gunn. Since the SMA controls both the Sky frequency and the IF center frequency, the LO frequency is determined by it. This is important, since the control of the LO will be done by the SMA (the PLL and the LO reference signals are connected to RxA3).

The lock status must be fed to the RxA3 by a hardware line. The reason for this is that the control software in the receiver is in control of the micrometer settings. Having a lock signal of the (external) PLL available will help locking the LO.

This function should only be called at the beginning of an observation. Calling this function in between observations, will adjust (if necessary) the mixer current, what will result in possible phase shifts.

A set of status information is needed from the receiver towards the SMA, for monitor issues. The following parameters are needed:

a) Mixer bias voltage
b) Mixer current
c) Lock status

This function is not available yet and should be created. The following command description will be used:

GET_RECEIVER_STATUS – **Get status information of the selected receiver**

**Function:** Get status information of the selected receiver

**Description:** Set the parameters for the receiver for observation

**Arguments:** GET_RECEIVER_STATUS expects the following arguments to be supplied on the command line in the order shown:

| | | | |
|---|---|---|---|
| 1. | RECEIVER | char | One of the JCMT receivers: 'RxA3', 'HARP', 'RxB', 'RxW' |

**Returned Arguments:** GET_RECEIVER_STATUS returns a space-delimited string containing the following quantities in the order shown. The string is terminated with a <CR>.

| | | | |
|---|---|---|---|
| 1. | STATUS | Int | The Status value |
| 2. | MXVOLT | Double | The Mixer Bias Voltage (in mV) |
| 3. | MXCURR | Double | The Mixer current (in uA) |
| 4. | LOCK | Char | Lock Status ('LOCKED' if locked) |

### 1.3. Single dish pointing of the telescope

At the JCMT a pointing routine is available. This is a single dish (JCMT) pointing routine. A quick 5 point pointing should be activated by the SMA at the beginning of each observation (and perhaps more if needed). This routine will take about 5 minutes to complete. A new routine must be written, which starts a pointing routine at the JCMT side. The following command description is suggested:

SD_POINTING – **Starts a quick 5-point pointing routine**

**Function:** Starts a quick 5-point pointing routine at the JCMT

**Description:** Starts a quick 5-point pointing routine at the JCMT, so the JCMT is pointing correctly. This should be invoked at the beginning of each observation.

**Returned Arguments:** SD_POINTING returns a space-delimited string containing the following quantities in the order shown. The string is terminated with a <CR>.

| | | | |
|---|---|---|---|
| 1. | STATUS | Int | The Status value |

As soon as the pointing is finished a flag is set in the JCMT software (The 'SD_POINTING_READY' flag). This flag can be checked by the following routine:

CHECK_SDPOINT – **Checks if pointing is ready**

**Function:** Checks the 'SD_POINTING_READY' flag at the JCMT to see if SD_POINTING is completed.

**Description:** Looks at the 'SD_POINTING_READY' flag at the JCMT to see if SD_POINTING is completed.

**Returned Arguments:** CHECK_SDPOINT returns a space-delimited string containing the following quantities. The string is terminated with a <CR>.

| | | | |
|---|---|---|---|
| 1. | STATUS | Int | The Status value |
| 2. | POINT | Char | 'READY' of pointing routine is ready and instrument is ready to observe. |

The pointing routine must be worked out if it is really required. Some workarounds must be made in the current routines at the JCMT, since manual interaction is required at this moment. Interferometric pointing might be implemented as well, which will be driven by the SMA software, using telescope control routines earlier described.

### 1.4. Single dish focusing of the telescope

At the JCMT a focus routine for all the three coordinates (X, Y and Z) is available. This is a single dish (JCMT) focusing routine, possible with 5 and 7 points. The following command description is suggested:

SD_FOCUS – **Starts a single dish focus routine**

**Function:** Starts a single dish focus routine at the JCMT, in X, Y or Z direction

**Description:** Starts a single dish focus routine at the JCMT.

**Arguments:** SD_FOCUS expects the following arguments to be supplied on the command line in the order shown:

| | | |
|---|---|---|
| 1. COORD | char | Coordinate for the focus routine: 'X', 'Y', or 'Z' |
| 2. NoP | Int | Number of points for the focus routine, 5 or 7 |

**Returned Arguments:** SD_FOCUS returns a space-delimited string containing the following quantities in the order shown. The string is terminated with a <CR>.

| | | |
|---|---|---|
| 1. STATUS | Int | The Status value |

As soon as the focus routine is finished a flag is set in the JCMT software (The 'SD_FOCUS_READY' flag). This flag can be checked by the following routine:

CHECK_SDFOCUS – **Checks if focus routine is ready**

**Function:** Checks the 'SD_FOCUS_READY' flag at the JCMT to see if SD_FOCUS is completed.

**Description:** Looks at the 'SD_FOCUS_READY' flag at the JCMT to see if SD_FOCUS is completed.

**Returned Arguments:** CHECK_SDFOCUS returns a space-delimited string containing the following quantities. The string is terminated with a <CR>.

| | | |
|---|---|---|
| 1. STATUS | Int | The Status value |
| 2. FOCUS | Char | 'READY' of focus routine is ready and instrument is ready to observe. |

The focus routine must be worked out if it is really required. Some workarounds must be made in the current routines at the JCMT, since manual interaction is required at this moment.

### 1.5. Polarizer

Since the SMA and the JCMT have different rotations at the sky, we have to correct for polarization. This is done by adding a half wave polarizing plate at the JCMT in case of receiver RxA3. This plate is motorized and can be controlled at the JCMT side. Since the SMA is in control of the eSMA, it should also control the position of the polarizing plate. It is therefore required that a routine is written, which calculates the position of the polarizer as function of the azimuth and elevation of the JCMT.

In the future a new heterodyne polarizer will replace the current UKT14 polarizer plate, ROVER. For HARP eSMA operations, the K-mirror of HARP will be used for correction of the polarization.

The following routine is proposed to control the plate at the SMA side:

SET_POLARIZER – **Set the polarizer at a specific angle**

**Function:** Set the Polarizer at the JCMT at a specific angle

**Description:** Set the polarizer at the JCMT at a specific angle. This command is internally translated to the 'move φ' command.

**Arguments:** SET_POLARIZER expects the following arguments to be supplied on the command line in the order shown:

| | | |
|---|---|---|
| 1. POLARIZER | Char | Polarizer: "UKT14", "ROVER", "HARP_K_MIRROR" |
| 1. POSITION | Int | Position of the plate in degrees. |

**Returned Arguments:** SET_POLARIZER returns a space-delimited string containing the following quantities in the order shown. The string is terminated with a <CR>.

| | | |
|---|---|---|
| 1. STATUS | Int | The Status value |

### 1.6. Thermally-controlled calibration

Calibration vanes (loads) must be moved in and out of the receiver beam at regular intervals to provide system temperature measurements. At the JCMT two loads are available, the hot and cold loads. The control should be done by the SMA. The following routine is suggested:

SET_LOAD – **Set the load of a receiver of the JCMT**

**Function:** Set the load of a receiver of the JCMT

**Description:** Set the load of the chosen receiver. The load could be the hot load, the cold load or the sky.

**Arguments:** SET_LOAD expects the following arguments to be supplied on the command line in the order shown:

| | | |
|---|---|---|
| 1. RECEIVER | char | One of the JCMT receivers: 'RxA3', 'HARP', 'RxB', 'RxW' |
| 2. LOAD | Char | Choice of the load of RxA3, could be "SKY', 'HOT' or 'COLD' |

**Returned Arguments:** SET_LOAD returns a space-delimited string containing the following quantities in the order shown. The string is terminated with a <CR>.

| | | |
|---|---|---|
| 1. STATUS | Int | The Status value |

The temperatures of the loads, needed for good calibration, can be obtained using the following routine:

GET_LOAD – **Get status information of the loads of a JCMT receiver**

**Function:** Get status information of the loads of a JCMT receiver

**Description:** Gives back the temperatures of the loads used at RXA3

**Arguments:** GET_LOAD expects the following arguments to be supplied on the command line in the order shown:

| | | |
|---|---|---|
| 1. RECEIVER | char | One of the JCMT receivers: 'RxA3', 'HARP', 'RxB', 'RxW' |

**Returned Arguments:** GET_LOAD returns a space-delimited string containing the following quantities in the order shown. The string is terminated with a <CR>.

| | | |
|---|---|---|
| 1. STATUS | Int | The Status value |
| 2. HOT | Double | Temperature of the HOT load (in K) |
| 3. COLD | Double | Temperature of the COLD load (in K) |

## 2. Action items

To be operational the following things must be implemented:

- Software connection of the eSMA VME crate and the jcmt04 terminal server
- Hardware connection of the 'lock' signal from the eSMA crate to RxA3
- Software development of:
    - SET_RECEIVER
    - GET_RECEIVER_STATUS
    - SD_POINTING
    - CHECK_SDPOINT
    - SD_FOCUS
    - CHECK_SDFOCUS
    - SET_POLARIZER
    - SET_LOAD
    - GET_LOAD
- Software development at the SMA side to handle the JCMT