

Phoenix software

M Kesteven & E Troup

February 1, 1995

Introduction

This note describes the software provided at Parkes and Mopra in support of the Phoenix program.

The software complies with the design document External Interface Design Document (May 18, 1994), and its revisions (June 20, July 29, October 29 and November 23).

Requirements

CSIRO will provide two serial lines - one expecting action commands, the other producing a stream of monitor data. The two lines connect to the CSIRO system at a terminal server; the line characteristics are currently set to 9600 baud.

The command line connects to a CSIRO task, SETIOBS. The monitor line connects to a CSIRO task, SETIMON.

To the outside world (SETI) the Parkes version of these tasks is the same as the MOPRA version.

The root directory is OBS\$1:[SETI] at Parkes, and OBS\$0:[SETI] at Mopra. These will be referred to generically as OBS\$x: in the text.

User Guide

Both tasks will maintain logs of all the messages received from or sent to the SETI system. These files are kept in the area user1:[observer.seti.log] at Parkes, and OBS\$0:[SETI.LOG] at Mopra. The files are called setimon.log and setiobs.log; a new file is opened each time the task is started.

The local implementation of the environment is given in the files SETIMON.DEF and SETIOBS.DEF in the OBS\$x:[SETI.WORK] area. (These are listed in appendix A).

In addition, a log is maintained in the file at `$log:sngl_command.log` of all the messages sent to the control system.

SETIOBS

To start the task : `$ run/nodebug obs$x:[seti.work]setiobs`

Once started the task will receive commands from the SETI system and pass them to the relevant CSIRO subsystem.

All incoming messages will be displayed on the task's screen; they will also be logged; they will be put into global common so that SETIOBS can be checked out remotely.

Every message from SETI will elicit a response. The responses are logged and displayed.

A 'QUIT' message from SETI will kill setiobs.

A 'MON_QUIT' message will kill setimon (see below).

Local specific information is stored in the file SETIOBS.DEF in `OBS$x:[SETI.WORK]`

Initialisation - MOPRA

When SETIOBS starts it puts the receiver and synthesiser in a standard setting:

- Select noise diode A
- Noise diode OFF
- Test Tone switch OFF
- Coarse attenuator set to 1
- Fine attenuator set to 4

- Frequency set to 2.0 GHz
- RF level set to 13 dB

Safety

An activity monitor has been placed on the Mopra version. A warning is sent back to Parkes if there is no activity in a five minute interval. The antenna will be stowed if there is no activity in a ten minute interval.

Setimon will show the warning/stow condition in the second record. (Data2 - the status word will switch to 000000000001 on warning, and 000000000999 on stow).

Debug mode

If the terminal name (in `setiobs.def`) is set to `TEST` then the SETI link is cut and the terminal keyboard becomes the input source.

Finer control of the debug modes is available in the file `obs$x:[seti.work]setiobs.def` – one can disable entirely the antenna drive (ie, one can test `SETIOBS` without requiring the antenna), or one can choose to drive just the ME (Parkes only).

SETIMON

To start the task : `$ run/nodebug obs$x:[seti.work]setimon`

This task will send two monitor records back to SETI at regular intervals. The period is currently set to 10 secs. It can be changed.

Every record is logged, as is the rare message from `setiobs`.

Communication with SETIMON

Information and commands can be sent to SETIMON through the shared common; we can use the `SETIOBS` communication channel to get information to SETIMON by prefixing 'MON_' to the message.

The only commands recognised at this stage:

<code>mon_interval secs</code>	This will change the reporting interval to secs (secs must be an integer in the range $3 < \text{sec} < 60$)
<code>mon_quit</code>	This will cause SETIMON to exit gracefully
<code>mon_timeout</code>	Setimon will issue an <code>INACTIVITY</code> warning
<code>mon_stow</code>	The inactivity-induced <code>STOW</code> message
<code>mon_clear</code>	Clears the inactivity warning message

Debug mode

If the terminal name (in `setimon.def`) is set to `TEST` then the SETI link is cut and the output is directed to the local screen.

SETIVIEW

This is a utility that allows remote access to the global common so that a check can be made on the operation of SETIOBS and SETIMON: all the traffic from the SETI system and the SETIMON reports will be displayed.

To start the task : \$ run/nodebug obs\$x:[seti.work]setiview

Code Management

The code is maintained using the Dec code management system (CMS). The library is in OBS\$x:[seti.code.cms], with a plain text copy of the current version in [seti.code.ref]

The executables are in [seti.work]

APPENDIX - the environment files

Parkes

setiobs.def

```
TEST:
PARKES
obs$1:[seti.work]marconi.def
user1:[observer.seti.log]setiobs.log
ENABLE
ME
```

line 1 : terminal line from vax (terminal server) to the seti HP
line 2 : observatory identifier
line 3 : file with marconi information
line 4 : log file
line 5 : Antenna Drive ENABLE/DISABLE
line 6 : Parkes specific : config .. ME; 64M; ALL

Mopra

setiobs.def

```
mopra_setiA_port:  
MOPRA  
obs$0:[marconi.work]marconi.def  
obs$0:[seti.log]setiobs.log  
ENABLE  
ALL
```

line 1 : terminal line from vax (terminal server) to the seti HP
If set to TEST, the SETI HP port is redirected to the
keyboard for debugging purposes.

line 2 : observatory identifier
line 3 : file with marconi information
line 4 : log file
line 5 : drive ENABLE or DISABLE
line 6 : PKS specific - ME/64M/ALL

setimon.def

```
MOPRA_setib_port:  
obs$0:[seti.log]setimon.log
```

line 1 : terminal line name; if set to TEST, setimon will take
its input from stdin.

