

**Supplement to Technical Information**  
**about the**  
**Small Radio Telescope**

The SRT Digital Receiver  
(notes by J.D. Litster).....1-4

SRT Circuit Descriptions  
(document from the Haystack website) ..... 5-10

Help File Contents from the Original SRT Program ..... 11-15

## The SRT Digital Receiver

These are some additional notes on the digital receiver in the SRT 21-cm radio telescope. They are intended to supplement the Haystack [1] notes. Some of the discussion in the Haystack document applies only to the analog receiver, and some is obsolete; these notes are intended to say more about the digital receiver.

### Digital Receiver:

The digital and analog receivers are identical until the output of the mixer. In the digital receiver the 800 KHz i.f. passes through a Panasonic ELK-EA220FA low pass filter that begins to roll off at 100 MHz before going into an amplifier with a voltage gain of about 5.7 that rolls off above 10 MHz. Then it goes to the A/D converter whose 8-bit output goes from 0 to 255 as the input voltage goes from  $-0.5$  V to  $+0.5$  V. The input voltage is sampled at 40 MHz.

The A/D output is fed to the Graychip receiver as bits 3 to 10 of its 12-bit input, corresponding to inputs from 8 to 2040 in steps of 8. The Graychip acts as a product detector (like a mixer with an i.f. at zero frequency) and produces 16-bit outputs that are in phase and  $90^\circ$  out of phase with the Graychip's local oscillator (derived from the 40 MHz clock). The resulting complex number will be input to the 56F803 DSP and a discrete Fourier transform will be taken to get the spectrum of the signal near the frequency the SRT is tuned to.

### Digital Signal Processor:

A lot of stuff is done by this chip; absent the code that runs it, I will guess about some of the things it does. I also assume that it is designed so that the digital receiver information sent back to the computer is largely backwards compatible with the information from the analog receiver. Most of the differences are associated with the different "modes" of the digital SRT.

The DSP listens for computer output that contains the word "freq", such as might be found in a line

```
: freq 1419.0 4
```

in a command file. The computer parses this line (see `cat.java`) before passing it along to the DSP so it will know what is going on, too. The computer sets the following global variables.

Mode	freqsep	nfreq	intg
1	0.0078125	64	0.52488
2	0.00390625	64	2*0.52488
3	0.001953125	64	4*0.52488
4	0.0078125	156 <sup>†</sup>	0.52488

<sup>†</sup> depends on number of ROLLOFF channels

What do these mean? First, *nfreq* is the number of complex output numbers that are taken from the Graychip as input to the FFT. The discrete FT means the following is calculated

$$X_{\omega_k} = \sum_{n=0}^{N-1} x(n\tau) e^{-i\omega_k n\tau} \text{ or } X_{\nu_k} = \sum_{n=0}^{N-1} x(n\tau) e^{-i2\pi\nu_k n\tau}$$

if *nfreq* = *N* and the A/D output is sampled every  $\tau$  seconds. The frequency coefficients  $X_{\nu_k}$  are all squared to give something proportional to power, rather than voltage.

For mode 1, where *freqsep* = 0.0078125 MHz and *N* = 64 the total sampling time is  $64\tau$  and the total bandwidth (which is  $1/\tau$ ) is  $0.0078125 \times 64 = 0.500$  MHz. This requires  $\tau = 2.00 \mu\text{s}$ . Thus each block of 64 samples of the A/D output requires  $128 \mu\text{s}$ . When 4096 blocks are accumulated to determine the power, this represents an integration time  $128 \times 4096 \times 10^{-6} = 0.52429$  s. (Presumably the 0.52488 represents some processing time or perhaps is a result of integer digital arithmetic.) For mode 2, the bandwidth is 0.250 MHz and  $\tau = 4.00 \mu\text{s}$ . For mode 4, *nfreq* = 156 (instead of  $3 \times 64 = 192$ ); the reason is found in the function `spectra()` in `procs.java`. In mode 4 the system makes three measurements, centered at the requested center frequency and that frequency  $\pm 0.360$  MHz. Thus 18 points of the upper and lower scans overlap the central one. They are used to match the upper and lower scans to the central one, and so the upper and lower scans each contribute only 46 new channels to give a total bandwidth of 1.21875 MHz.

### What does the “power” mean?

In the `.rad` file the values of *freqsep* and *nfreq* are reported. Then for each frequency, there is a number given. What is this number?

It is a number that comes back from the radiometer, either from the DSP with the digital receiver or Stamp controller in the case of the analog receiver. According to the Haystack notes, the Stamp controller in the analog receiver returns the number of pulses from the V to F converter that is attached to the square law detector in the receiver. The DSP must return something that can be interpreted the same way. If you look in `sport.java` (which handles the serial I/O) you can see how this works. There are two routines that set up the frequency the radiometer will look at: `radio()` is used for the analog receiver and `radiodg()` is used for the digital receiver; the routine `cal()` is used with the calibration vane and also turns on the noise calibration diode.

To make each measurement, the command “`freq 1419.0 1`” is sent by the computer to the radiometer (with appropriate frequency and mode numbers). The radiometer returns a 128 byte array, or three of them when mode = 4, with the “power” for each frequency channel is encoded as a 16-bit integer in two elements of the array. The numbers returned are modified in two ways before being used or sent to an output file.

First, there is a table of 33 numbers in `procs.java` that are apparently used to compensate for filtering done by the Graychip detector. Here is the routine that installs the table:

```

void grayfilt(global g)
{
  int i;
  double cf[] =
  {
    1.000000, 1.006274, 1.022177, 1.040125, 1.051102, 1.048860, 1.033074, 1.009606,
    0.987706, 0.975767, 0.977749, 0.991560, 1.009823, 1.022974, 1.023796, 1.011319,
    0.991736, 0.975578, 0.972605, 0.986673, 1.012158, 1.032996, 1.025913, 0.968784,
    0.851774, 0.684969, 0.496453, 0.320612, 0.183547, 0.094424, 0.046729, 0.026470,
    0.021300
  };
  for (i = 0; i <= 32; i++)
  {
    if (i < 32)
      g.set_graycorr(cf[i], i + 32);
    if (i < 33)
      g.set_graycorr(cf[i], 32 - i);
  }
}

```

Here  $i$  is the number of the frequency channel (the middle is  $i = 32$ ) and the power returned by the digital radiometer for each channel is divided by  $graycorr(i)$  so long as  $graycorr(i) > 0.8$ . There is an additional “curvature” correction that may occur. The power for channel  $i$  will next be divided by  $1.0 + a^2 \times curvcorr$ , where  $a = 0.4 \times freqsep \times |i - 32|$ . The value of  $curvcorr$  can be set in the `srt.cat` file and the default is usually zero. The resulting number is presumably closely similar to what the radiometer with an analog receiver would return.

For both radiometers, the result will be multiplied by a constant called *calcons*; this is normally 1.0, but can be set to something else in the `srt.cat` file and is adjusted during calibration (next page).

The routine `radiodg()` also calculates an average power, which is an average over the central 44 of the 64 frequency channels.

Nowhere in the java code did I find anything like the discussion using *scalef* that is in the Haystack notes, so it was still not clear what the “power” numbers meant, other than they were proportional to the power at the receiver’s detector. A closer examination of the `noisecal()` routine in `procs.java` showed that the calibration procedure has changed. The help file tells something about this, but it’s not easy to read the way the SRT program’s “GUI” works. Probably better to look at the java source, `hdisp.java` (see appendix).

### Noise Calibration:

Calibrations are now done in terms of noise temperatures using the noise diode. The vane is not needed and no longer used. The noise temperature of the diode calibrator is  $T_{noise} \simeq 115\text{ K}$  [2] and this number should be provided as “NOISECAL 115” in the `srt.cat` file. Another number, that is of less use, is the temperature of antenna “spillover”, which is compiled into `srt.java` as  $T_{spill} = 20\text{ K}$ .

First, the telescope should be pointed at a part of the sky where there are no radio sources, and the frequency and mode (=1) should be set with the `freq` command. Then give the `noisecal` command in a command file or at the console (`Cal`). It may be a good idea to do all of these things in a command file with `record` turned on. Then you will have a good record of everything you did and can get the relatively “raw” data from each frequency scan.

As a result of the `noisecal` command the SRT will do two frequency scans. The first will be with the noise diode turned on, and the second with it turned off. The power data for both scans will be written to the `.rad` file. With the digital receiver, the central 44 frequencies of the 64 for each scan will be averaged to get  $pwr_1$  (noise on) and  $pwr_0$  (noise off). The program calculates the following temperatures and prints the results in a comment line:

$$T_{\text{rcvr}} = T_{\text{noise}} \times \frac{pwr_0}{pwr_1 - pwr_0} - T_{\text{spill}}$$

$$T_{\text{sys}} = T_{\text{rcvr}} + T_{\text{spill}}$$

and sets the constant (see previous page)

$$calcons = \frac{T_{\text{sys}} \times previous\_calcons}{pwr_0},$$

where *previous\_calcons* is the value of the calibration constant that was in effect when the numbers  $pwr_0$  and  $pwr_1$  were measured; thus repeated calibrations should give closely similar results. After this, the power figures returned by subsequent calls to the radiometer should all be in Kelvins, based on  $T_{\text{noise}} = 115$  K. They will be the sum of  $T_{\text{sys}}$  and  $T_{\text{ant}}$ , where  $T_{\text{ant}}$  is the signal at the antenna feed that results from the source in the sky that you are looking at. Remember to subtract  $T_{\text{sys}}$  in order to obtain  $T_{\text{ant}}$ .

If you get a message “\* ERROR communicating with radio” it usually means that the radiometer did not send back the expected 128 bytes of power data to the computer.

A message “\* ERROR lost count” comes from the routine that moves the azimuth or elevation motors. Each motor usually sends 12 pulses per revolution and these are counted to determine the position of the antenna.

A message “\* ERROR gnd 1 radio 0” means that the computer has lost contact with the “ground” controller box (on the table beside the computer) or the radiometer. The numbers following “gnd” and “radio” tell you how many errors of each type have occurred. Following this error message, the software will stow the antenna.

### Antenna:

The antenna for our SRT is different from the one described in the Haystack receiver circuit notes. It has a diameter of  $D = 90$  in (2.3 m) and so its beamwidth given by the conventional  $1.22\lambda/D$  formula is about  $6.5^\circ$ . This formula, incidentally, gives the full angular beamwidth to the first zero of the diffraction function  $[2 J_1(x)/x]^2$ , where  $x = (2\pi D/\lambda) \sin \theta$  and  $\theta$  is the angular offset from the axis of the antenna. The full width at half power for the antenna is about two times smaller; it requires  $x \simeq 1.6$  which means a beamwidth of  $0.51\lambda/D$  radians or  $30\lambda/D$  degrees ( $\simeq 3^\circ$ ).

As these notes are intended to accompany the Haystack document, a typeset copy of it ([http://haystack.mit.edu/edu/undergrad/srt/receiver/receiver\\_circuit.html](http://haystack.mit.edu/edu/undergrad/srt/receiver/receiver_circuit.html)) follows on the next six pages.

## SRT Circuit Descriptions

### A) Low Noise Amplifier (LNA)

The LNA provides a nominal 24 dB gain. The input is matched to the 50 ohm feed for lowest noise using a capacitor and inductor to provide a good noise match as described (<http://www.haystack.mit.edu/edu/undergrad/srt/receiver/5988-4210EN.pdf>) in the ATF-34143 data sheet. The output of the preamp is filtered with a 1420 MHz ceramic filter with 40 MHz 3 dB bandwidth to prevent out of band signals from producing intermodulation products in the mixers. To minimize connections the preamp is powered by +5 v carried through the R.F. connector. At some sites where there are strong interfering signals it may be necessary to add the low loss filter on the input to the preamp.

### B) Image Rejection Mixer

A pair of mixers with 90° quadrature phased L.O. drive followed by 90° I.F. phasing networks make a “phasing type” single sideband receiver. The lower sideband is only rejected by 20 dB but since the unwanted “image” sideband is adjacent to the wanted sideband we don’t expect strong signals in the radio astronomy band to be present in the image. Both the preamp and the MAR-6SM amplifier provide a large amount of reverse isolation so that the level of L.O. radiation, which might interfere with others, is minimized. The unconventional use of the single conversion to a low “baseband” frequency I.F. has the advantage of avoiding a difficult image problem, allows simple active I.F. filtering, as well as allowing for easy future digital signal processing of the baseband. The disadvantage is that we need to rely on amplifier reverse isolation to prevent significant L.O. radiation. The AD8011 operational amplifiers are unit gain all-pass filters with 90 degree phase difference determined by  $C_1$  and  $C_2$ . Since most of the radiometer’s gain is in operational amplifiers it is very stable and requires only infrequent calibration.

### C) Gain Control

A 10 dB attenuator can be switched in to prevent overloading the power detector when observing strong signals like those from the sun when is in an active state.[Analog receiver only; the digital receiver saturates when the radiometer output is 50 000.]

### D) Active I.F. Filter/Amplifiers [analog receiver]

A pair of AD8011 operational amplifiers provide 53 dB of gain and high/low pass filtering. (<http://www.haystack.mit.edu/edu/undergrad/srt/receiver/ad8011.pdf>) The high pass cuts off frequencies below 10 kHz while the low pass cuts off frequencies above 70 kHz. The high pass is a 4-pole Butterworth filter with 56 kHz 3 dB passband.

### E) Square Law Detector [analog receiver]

The conversion of the I.F. frequency voltage waveform to a voltage which is proportional to the I.F. signal power uses a back-diode or Schottky diode in the “squaring” region. Because the square law region is at a very low level the diode output is amplified by a factor of 50 using a OP-27 amplifier.

### F) Analog to Digital Conversion [analog receiver]

The I.F. power is an analog signal which is converted to a pulse whose duration is inversely proportional to the average power present during the pulse period. The AD654 and the stamp *pulsin* routine performs this “integrating” analog to digital conversion (ADC). The AD654 output is a 12 ms positive pulse for a nominal 0.5 volt output from the OP-27 followed by a fixed 3 ms negative pulse.

This negative pulse is shortened by the current through a diode when the output goes low. See the AD654 data sheet for the details of its operation. The stamp uses the *pulsin* routine to measure the duration of the positive pulses.

### G) Local Oscillator Synthesizer

A voltage turned oscillator (JTOS-2000) is “phase locked” to the 8 MHz crystal oscillator using a LMX2325 synthesizer chip.

The details ([www.haystack.mit.edu/edu/undergrad/srt/receiver/LMX2315.pdf](http://www.haystack.mit.edu/edu/undergrad/srt/receiver/LMX2315.pdf)) of the LMX2325 and the principles of the phase lock are described in the data sheet. A MAR-4SM amplifier is used to isolate the oscillator from the mixers and an OP-27 (<http://www.analog.com/en/prod/0,2877,0P27,00.html>) is used to boost the LMX2325 phase detector voltage so that the oscillator can reach the OH frequencies at 1665 MHz which require a tuning voltage above +5 v.

### H) Serial Communication

The dc power along with bi-directional serial communication are multiplexed onto a single conductor coax cable which connects the “ground” electronics to the receiver electronics on the antenna.

The voltage ([www.haystack.mit.edu/edu/undergrad/srt/receiver/LM117.pdf](http://www.haystack.mit.edu/edu/undergrad/srt/receiver/LM117.pdf)) from the LM317T conveys the dc power and the “uplink” to the receiver as follows:

RS-232 input	LM317 out	Voltage on cable
+12v	22v	19v
-12v	17v	14v

A CMP402 ([www.haystack.mit.edu/edu/undergrad/srt/receiver/cmp401.2.pdf](http://www.haystack.mit.edu/edu/undergrad/srt/receiver/cmp401.2.pdf)) comparator in the receiver is provided with a 16 v threshold which converts the 19 v (RS-232 high) to 0 volts and the 14 v (RS-232 low) to 5 volts. The cable voltages in the above table are for the “downlink” not transmitting state for which the receiver draws a 150 ma. When the downlink is active and a transmitted “one” is conveyed by drawing an additional 100 ma which drops the cable voltage by 2 volts so that a CMP402 on the “ground” with 13 v threshold can be used to drive the RS-232 output. (A 0 to 5 volt output is used to drive the serial COM port. This is not a true RS-232 signal but should work with all PCs)

### I) Stamp Controllers

The radiometer and antenna drives are controlled via Stamp 1 controllers. The RS-232 serial port is used to talk to both stamps using a different “keyword” for each stamp and a wired logical “OR” for the response from each stamp. This method allows communication with one stamp at a time. Under most circumstances this is acceptable since we don’t need the radiometry data while the antenna is in motion. [The digital receiver uses a DSP instead of a stamp controller, but the RS-232 port sharing is still used.]

### J) Biasing the Operational Amplifiers

Only a positive voltage is supplied to the radiometer. Since the operational amplifiers require both polarities the inputs are biased to a voltage above ground potential with resistor network. In addition there are many decoupling capacitors to filter the power and bias voltages.

### **K) Motor Control**

LMD-18200 ([www.haystack.mit.edu/edu/undergrad/srt/receiver/LMD18200.pdf](http://www.haystack.mit.edu/edu/undergrad/srt/receiver/LMD18200.pdf)) power H-bridges to control and reverse the +24 v dc power to the motors.

### **L) Antenna Feed**

The radiometer is connected to a feed which is illuminated by a 10 foot diameter dish with  $f/D$  ratio of 0.4. The feed is made by adding a  $\lambda/4$  probe to the C-band (4.2 GHz) feed. The outer diameter of the C-band feed is about 8 inches, which supports a  $TE_{11}$  circular waveguide mode with  $a/\lambda \simeq 0.5$  which is close to optimum for a  $f/D \simeq 0.4$  [see Methods of Experiment Physics, Astrophysics edited by Marton, vol. 12B page 32]. The aperture efficiency should be close to 60% (or 50% when the feed and feed support blockage is taken into account) so that the antenna temperature is about

$$T_{\text{ant}} = (FA\eta/2k)$$

where

$A$  = area of the reflector in  $m^2$

$F$  = radio source flux density in  $w m^{-2} Hz^{-1}$

$\eta$  = aperture efficiency  $\simeq 0.5$

$k$  = Boltzmann's constant =  $1.38 \times 10^{-23} w Hz^{-1} k^{-1}$

The antenna beamwidth should be

$$qB = 1.22 \lambda/D \simeq 5^\circ \text{ (10 ft dish diameter at 1.4 GHz)}$$

### **M) Digital Receiver**

The digital receiver replaces the single 40 kHz filter and analog square law detector with a 8-bit analog to digital down converter (AD9283) digital downconverter GC1011A (see [www.ti.com](http://www.ti.com) - search on Graychip) and Motorola 56F803 digital signal processor (DSP).

#### **a) R.F. Section**

The output of the LNA is further amplified by a MAR6SM and down converted using an image reject mixer. Current firmware selects an I.F. frequency of 800 kHz and a high side L.O. has been chosen. i.e., the frequency synthesizer is set to a frequency 800 kHz higher than the desired R.F. center frequency. A total I.F. gain of 50 dB before A/D conversion.

#### **b) A/D Conversion**

The 8-bit A/D converter provides an offset binary code which is connected to bits 3 thru 10 of the digital down converter. The MSB (bit 11) is set to zero as are bits 0 thru 2. This scheme avoids the need to invert the MSB of the A/D to generate the two's complement data expected by the Graychip.

#### **c) Digital Down Conversion**

The GC1011A uses quadrature digital mixers with numerical local oscillator. The "sine" and "cosine" outputs are low pass filtered in stages to avoid aliasing. The low pass filtered 16 bit outputs are accessed by the 56F803 DSP.



#### **d] DSP**

The 56F803 DSP is a general purpose processor that performs the following functions (with current firmware code):

1. Uses the built-in UART to listen for the keyword “freq” which will be followed by the mode.
2. Sets up the LMX2325 L.O. frequency synthesizer.
3. Sets up the GC1011A Graychip digital L.O. frequency, decimation ratio, gain, and selects the default 80% filter.
4. Reads output from the Graychip 64 complex samples in a block.
5. Each 64 sample block is Fourier transformed using 64 point complex FFT.
6. The FFT outputs are squared to obtain the spectral power.
7. The spectral power from each block is accumulated for 4096 blocks.
8. The accumulated outputs are sent back at 2400 baud.

The code is placed in flash memory of the 56F803 using the Metrowerks embedded development system via the JTAG port. The source code (which is mostly in C, is available) requires the development system and JTAG converter to be altered and re-downloaded.

#### **e] Power, Communication and Digital Clocks**

As in the analog receiver the ground controller provides the dc power and 2-way serial communication. The communication from the controller to the 56F803 is performed by the supply voltage which changes from a nominal 14 volts to 18.8 volts for the “uplink”. (this is based on a current of 210 ma - which is typical of the receiver and a 13.3 ohm series resistor in the controller.) The down link is obtained by increasing the current from about 210 ma to about 400 ma. The 3.3 volts for the A/D, Graychip and DSP is provided by an efficient DC to DC converter (LT1616). The fundamental clock for the receiver is the 8 MHz crystal of the R.F. synthesizer. The 8 MHz is sent to the 56F803 whose internal clock circuitry multiplies the 8 MHz to 40 MHz for the A/D sample clock and the Graychip clock and to 80 MHz for the internal core DSP logic.

#### **f] Ground Controller**

A new ground controller circuit provides control for the analog receiver (series resistor of 20 ohms - jumper removed) or the digital receiver (series resistor of 13.3 ohms - jumper in place). The new circuit also includes a true RS-232 level converter to ensure reliable communication for longer cables between the PC and the controller.

## N) Radiometer Stamp Control Firmware

The radiometer stamp communicates with the PC serial port, controls the L.O. frequency synthesizer, the 10 dB attenuator and measures the radiometer power.

### a] L.O. frequency synthesizer

The reference divides (R register) is set to 200 to provide a 8 MHz/200 = 40 kHz reference. [This could be changed by small amounts for added frequency flexibility but large changes may require changing the loop filter capacitor values.] The prescaler is set to 64/65 by making S15 = 0. The A register is set by first calculating its value in the host PC  $j = (freq/0.04) + 0.5$ , where  $freq$  is in MHz and  $j$  is a 32 bit unsigned integer. Then separating  $j$  into bytes

$$b11 = (j >> 14) \& 0xFF$$

$$b10 = (j >> 6) \& 0xFF$$

$$b9 = j \& 0x3F$$

Because the synthesizer is quantized the actual frequency will be

$$freq = ((256 * b11 + b10) * 64 + b9) * 0.04$$

in the case that the frequency selected is not a multiple of the 40 KHz reference frequency. The bytes b11, b10, b9 along with a byte b8 to control the attenuator are sent to the Stamp following the keyword “freq”.

### b] Power measurement [analog receiver]

The radiometer power is measured by using the *pulsin* routine (see Stamp 1 manual) to count the Stamp clock cycles in a pulse from the AD654 ([www.haystack.mit.edu/edu/undergrad/srt/receiver/ad654.pdf](http://www.haystack.mit.edu/edu/undergrad/srt/receiver/ad654.pdf)) voltage to frequency converter. To enhance the accuracy we count up to  $10^4$  10 microsecond cycles. During this time we may have several AD654 positive pulses so we return the total count  $w1$  and the number of pulses counted  $w2$ . The host PC then calculates the power as  $Power = scalef \times (w2/w1)$  so that, for example, a 0.5 volt out of the OP-27 which produces a pulse length of 12 ms so that  $w2 = 8 = b4$ ,  $w1 = 9600 = b3 \times 256 + b2$  and  $Power = 833.33$  when  $scalef = 106$ . The scale factor of 106 is approximately the value which makes the output in units of K but each individual radiometer needs to be calibrated if you want the power to be in units of Kelvins.

## O) Antenna Drive Control Stamp

The antenna azimuth and elevation drives are controlled one at a time with a Stamp 1 in the power supply unit which is located near the control PC. The motors are activated when the stamp detects the keyword “move”. The keyword is followed by a byte *mm* which gives the axis and direction to the move and the number of “counts” of the reed microswitch on the drive gear to move. The motors drive a magnetic disk with 12 poles giving 12 contact closures per rotation which in turn drive a sprocket gear with 8 teeth which chain drives a large 54 tooth sprocket gear which in turn drive a sector gear of 52 teeth via a worm gear.

Thus the number of counts per degree is  $12 \times 54 \times 52 / (8 \times 360) = 11.70$ . The Stamp counts the positive pulses from the reed microswitch using the *pulsin* routine. The routine is repeated until the desired number of pulses have occurred or the *pulsin* routine times out and returns to zero. Only one axis can be moved at a time because the *pulsin* routine can only monitor one microswitch at a time.

**Note added by JDL:** The “move” command is used internally by the *java* program and is not accessible from the console or a command file. The synopsis is: `move mm counts` where the meaning of the byte *mm* is

<b>mm</b>	<b>Result</b>
0	decrease azimuth (CCW)
1	increase azimuth (CW)
2	decrease elevation
3	increase elevation
6	turn off noise diode
7	turn on noise diode

## Appendix: SRT Program Help

This is a printed list of the help information in the original SRT *java* program. It is from the original file `hdisp.java`; some of it still applies to the new user interface program and some does not.

### SRT.CAT

```
srt.cat file contains keywords which are case sensitive
keywords:
STATION lat(deg) lon(west deg) name
AZLIMITS anti-clockwise_viewed_from_above(deg) clockwise(deg)
ELLIMITS lower(deg) upper(deg) - upper limit will be greated
than 90 degrees as dish goes over or it's back to look north
[COMM serial_port(default = 1) normally 0 for Linux]
[CALCONS initial_ratio_of_degK_per_count - gets changed by
calibration]
[TLOAD load_temperature(K) - default 300K]
[MANCAL 0 - motor driven vane MANCAL 1 - vane calibration is
to be done manualy]
[TSPILL antenna_spill_over(K) - default 20K]
[BEAMWIDTH antenna_beamwidth(deg) - default 5 deg]
[AXISTILT az_axis_tilt(deg) tilt_az(deg) el_axis_tilt(deg) - defaults 0 0 0]
SSAT satellite_ID sync_orbit_satellite(deg_west)
AZEL az(deg) el(deg) name - catalogs a fixed azel position
GALACTIC long(deg) lat(deg) name
SOU ra(hh mm ss) dec(dd mm ss) name [epoc]
negative declination must have - in front of dd
Sun - add the Sun to the catalog
Moon - add the Moon to the catalog
DIGITAL - if present assumes receiver is digital
NOISECAL temp(K) - calibration temperature of noise diode
- needs to be present to support electronic noise cal
[TOLERANCE counts - counts of error which can accumulate
before command to drive [default = 1]]
[COUNTPERSTEP counts - counts per step for stepped
antenna motion [default = no stepped motion]]
[ELBACKLASH - optional correction for
elevation backlash to improve pointing in flipped mode [default = 0]]
[CURVATURE - optional correction for
curvature in spectrum [default = 0]]
[RECORDFORM TAB - adds tabs to separate columns in output
file - default is space delimited VLSR adds vlsr and
DAY forces a file change at each new day]
SIMULATE RECEIVER - alternate to using command line
SIMULATE ANTENNA value - alternate to using command line
MAINTENANCE - alternate to using command line
* in first column makes line a comment and items within a
line following "/" are comments
```

## SRT.CMD

command file rules: (filename srt.cmd)  
1] reads one line at a time skipping blank lines,  
and lines which start with \* and lines with past times  
2] stops at and executes any line with current or future time  
time format: line starts with yyyy:ddd:hh:mm:ss and is followed by cmd  
alternate format: LST:hh:mm:ss cmd  
current time format: : cmd  
current time plus n seconds format: :n cmd  
each line MUST start with time or : or \* for comment  
command keywords: items in [ ] are optional  
sourcename,mode,radec,azel,galactic,offset,stow,calibrate,  
noisecal,record,freq,roff  
sourcename (any name in catalog) [mode]  
mode n(for 25\_point scan) b(for beamswitch)  
radec hh:mm:ss [sign]dd:mm:ss [epoch] [mode]  
azel az\_deg el\_deg  
galactic glat\_deg glon\_deg  
offset azoff\_deg eloff\_deg  
stow  
calibrate (for vane) or noisecal (for noise diode)  
record [filename] [recordmode]  
modes: 0=normal, 1=short, 2=add vlsr, 3=special, 4=summary only  
where short - suppresses recording cmds  
roff (turns off record)  
freq frequency num [spacing] - for analog receiver  
freq frequency digitalmode - for digital receiver  
Note that 25-point scan runs only once while beamswitch until next cmd  
For example:  
: Sun  
:60  
:600 Sun n  
: stow  
goes to the sun,waits 60 seconds,does 25 point  
and after 600 seconds goes to stow  
Note that the seconds to wait is right next to the ":"  
and otherwise there is a space before the command

A command file can be checked by running java in simulate mode  
-see help on cmdline options  
another name for a command file can be entered by clicking  
on the text area.

## PLOTS

instantaneous spectrum:

black plot on right is the spectrum from the last reply from the receiver

accumulated spectrum:

red plot is the accumulated spectrum with a baseline constant and slope removed  
in beamwith mode the difference in total power from the sum of all frequencies is given with error estimate

accumuated spectra plot:

a separate plot window can capture the accumulated spectrum by clicking on the spectrum  
when the size of the window is changed a new spectrum is captured  
the window can be minimized and later restored without forcing a new capture  
if a new capture is desired - close and reopen the window

contour map:

the last npoint map can be replotted by clicking on the plot area

power vs time:

a "strip chart" of the total power is plotted just above the sky display. The plot offsets modulo the plot scale. colors change with the offset starting with black for zero offset  
the plot scale default is 400K but can be toggled to 40K then 4K and back to 400K by clicking on the text

## OUTPUT FILE

the output file is an ASCII file with spaces or optionally tabs as delimiters  
the fields are as follows:  
field 1 - time (yyyy:ddd:hh:mm:ss)  
field 2 - azimuth(deg)  
field 3 - elevation(deg)  
field 4 - azimuth offset(deg)  
field 5 - elevation offset(deg)  
field 6 - first frequency(MHz)  
field 7 - digital frequency separation(MHz) or first freq  
for analog receiver  
field 8 - digital mode  
field 9 - number of frequencies in digital mode  
field 10 - first frequency channel for digital receiver  
field 11 - second frequency  
following fields - continuation of frequency sequence  
units are in deg K  
last field optionally selected by keyword RECORFORM VLSR  
is the velocity of the local standard of rest VLSR (km/s)  
other information is added with \* in first column  
if the output file doesn't exist it is created in the  
current directory  
If the file already exists the data is appended to the  
end of the file

## CMDLINE

when srt is run from the command prompt  
command line options:  
srt without arguments to get options and version  
srt 0 to run with receiver + antenna  
srt 0 1 to simulate antenna  
srt 1 to simulate receiver  
srt 1 1 to simulate both receiver and antenna  
srt 1 10 to simulate and speed-up time x 10  
srt 1 -1 to simulate with 1 hour advance  
srt 1 0 1 for antenna maintenance  
takes incremental commands and doesn't drive to stow

Simulation:  
in simulation signals are generated for the Sun, Moon  
and Cass with strengths of 100K,1K and 2.6K respectively  
in addition Gaussian noise is added with the theoretical  
amount for each receiver mode

## HOWTO

pointing adjustment:

run npoint on the sun and subtract the azimuth offset  
from the start of the azimuth limit  
subtract the elevation offset from the starting elevation  
limit if needed tilt and backlash corrections can be added

checking receiver communications problems:

if antenna communication doesn't work check this first  
make sure receiver is connected - try powering the controller  
down and back up to reboot the receiver

checking antenna communications:

disconnect RS232 cable from controller or turn controller off  
srt software should say "waiting on azimuth"  
disconnect antenna coax and antenna cable  
srt software should indicate an immediate timeout in azimuth  
and elevation

screen saver:

the screen saver may cause the srt program to stall  
possible solutions are:  
a) minimize srt program when not using  
b) turn off screen saver

error calculations:

the noise in each spectral bin for a single response from  
the receiver (a single line in the output file) is given  
by  $t_{\text{sys}} / \sqrt{\text{resolution} * \text{intg. period}}$   
the resolution for the digital receiver is equal to the spacing.  
However in many cases the errors are determined by systematic  
effects. For example the errors in a beamswitch continuum observation  
are estimated from the variance in the data from many on/off

## REFERENCES

- [1] [http://haystack.mit.edu/edu/undergrad/srt/receiver/receiver\\_circuit.html](http://haystack.mit.edu/edu/undergrad/srt/receiver/receiver_circuit.html)
- [2] [http://haystack.mit.edu/edu/undergrad/srt/receiver/calibrator\\_report.pdf](http://haystack.mit.edu/edu/undergrad/srt/receiver/calibrator_report.pdf)



MIT  
OpenCourseWare  
<https://ocw.mit.edu>

8.13-14 Experimental Physics I & II "Junior Lab"  
Fall 2016 - Spring 2017

For information about citing these materials or our Terms of Use, visit: <https://ocw.mit.edu/terms>.