

Subj: Usuda Time Correction File: The dt program
From: Glen Langston
Date: 99 March 31

Summary

This document describes software tests made during the development of a new version of the Time Correction File (TCF) program for the Usuda Tracking station. The program, called dt, runs after Usuda tracking passes are complete to create TCFs to be used at correlators. The dt program takes many input files created at the Usuda station and generates a single FITS format output file containing one or more TCF tables. (See OVLBI-ES memos 64, 65, 66, and 71, for background information, on the web at <http://www.gb.nrao.edu/ovlbi/memos.html>)

The new TCFs are successful in reducing the high residual rates found during correlation of Space VLBI data with data from ground antennas. The residual clock offsets are slightly reduced, but remain of order $\pm 10 \mu\text{sec}$. The dt program has a unique capability; it can make corrections to the up and down-link frequency transmissions, if new predictions are available after data recording. This feature has been successfully used at the Penticton, Socorro and Mitaka correlators. Unfortunately, it has not been possible to reduce the variable part of the clock offset in the TCF files below $10 \mu\text{sec}$. Further work is required to achieve this goal.

Appendix A describes the Usuda Closed Loop mode and appendix B describes the dt program.

Background

Unknown problems existed in the Usuda Time Correction Files (TCFs), which resulted in large residual offsets ($\sim \pm 10 \mu\text{sec}$) and rates ($\sim \pm 500 \times 10^{-12} \text{sec/sec}$) at correlation. Tests of the Usuda station hardware had found no problems and also found that all Usuda time constants used in the process of making a TCF were consistent with hardware measurements. (See OVLBI-ES memo 74, "Usuda Time Correction File Tests: Phase detection delays")

Tests had shown that there appeared to be inconsistencies in the Usuda up and down-link frequency predictions that could cause high delay rate residuals seen during data correlation. (See OVLBI-ES memo 75, "Usuda - Halca Range Predictions: The orbitUsuda program".) These inconsistencies could be corrected after the tracking pass, but this required extensive modification of the TCF generating program.

The original TCF program, written by T.Nishibori (Boris), had a number of good features, including functions to read in the Usuda station output files and combine the time offset measurements to create the FITS format output. The program contained internal on-line help and produced a large amount of diagnostic print-outs. One particularly valuable feature was that the program read in both the Usuda up-link prediction file *and* the files containing the actual up and downlink IF frequencies used during the tracking passes. This feature was used in the new version to correct the link predictions after the pass.

The original program, written in C, was contained in a single file (module) containing over 4000 lines. It was difficult to work with this program and to understand the relationship of all of the functions in the module. The new dt program is still large, more than 900 lines, but many of the original functions have been "modularized".

Nori Kawaguchi and his student Kazumasa Suzuki were making good progress towards reducing the high frequency residuals via a different method, changing the up-link frequency so as to zero the downlink frequency residuals. (Called the closed loop method.)

dt Inputs

The dt inputs consist of command line arguments and a set of output files from the Usuda tracking station. Each of these files has a two part name. The first part is a date and pass number string. The second part is a three character extension type. There are ?? possible input files to the dt program. The extensions are include at the end of the descriptions. The inputs are:

Prediction : Satellite up and down-link predictions, tabulated once a second for each tracking pass. (See OVLBI-ES memo 75.) (.fpr)

G1S1 : File containing the counter values of time since last Satellite 1 Hz (S1). The counter values are read every Ground 1 Hz (G1). The counter is based on a high frequency (5 MHz) ground signal and the counter is reset to zero every S1 pulse. (.tic)

Up-link Frequency : The IF frequency actually used to generate the up-link frequency sent to the satellite. In Usuda open-loop mode, the up-link IF frequency is the same as the predicted IF frequency. (.ful)

Up-link Control : The IF frequency control log includes the intended up-link control frequency extracted from the .fpr file. In Usuda open-loop mode, the up-link IF frequency is the same as the predicted IF frequency. (.fuc)

Down-link Frequency : The IF frequency actually used to generate the down-link frequency used for phase comparison. In Usuda open-loop mode, the down-link IF frequency is the same as the predicted IF frequency. (ASCII values, .fd1)

Down-link Control : The IF frequency intended to be used to generate the down-link frequency used for phase comparison. In Usuda open-loop mode, the down-link IF frequency is the same as the predicted IF frequency. (ASCII values, .fdc)

Phase data : The Usuda station data are recorded at rate of 10 times a second and written into a file in a compressed manner, where the digits of phase data are written in an ASCII form. The time tags are written in DDDhhmmss format (where DDD is day of year) once a second, followed by 10 residual phase measurements. (.kph)

Station Log : The Usuda station log contains times of clock resets, as well as describing when the link to the satellite is stable. (Japanese, 16 bit char, text file, .ulg)

VLBA Tape Log : Short log file describing how the VLBA data are written to the wide band data tape. Usually only one of the three tape logs is present. (ASCII, .mk1)

S2 Tape Log : Short log file describing how the S2 data are written to the wide band data tape. (ASCII, .s2l)

VSOP Tape Log : Short log file describing how the VSOP data are written to the wide band data tape. (ASCII, .vs1)

GPS Log : The GPS log records contain the GPS time and the offset between station G1 Hz and GPS 1 Hz recorded once a second, on the GPS 1 Hz. (G1 Hz - GPS 1Hz is recorded in the file). This file is different from other files; it is generated daily, not per pass. The file name is the 6 character date in YYYYMMDD format. (ASCII, .tml)

Additional files are generated at the Usuda station after each pass, but are not used in the TCF process. These files are:

Range Rate Residuals Range rate residuals are generated by a device very similar to that which generates the phase residual data. (.krr)

Phase Cal : Phase calibration tone data extracted from the wide band data. (.pcl)

Phase Cal Log : Describes the setup used for extraction of Phase calibration tone data. (.plg)

Ku Telemetry : Contains the telemetry data extracted from the Ku band data stream. (Binary, .ktu)

Before describing tests of the new dt program, several aspects of the Usuda station time keeping are discussed. One particularly valuable features of the Usuda station is the regular recording of the offset between Satellite 1 Hz (S1) and Ground 1 Hz (G1). The G1 signal is kept as close as possible to UTC time, within $1 \mu sec$. The G1-S1 value is read by a hardware device which counts pulses of a high frequency signal (5 MHz) locked to the maser. The counter is read on the G1 signal and the counter is zero-ed on the S1 signal. When G1 and S1 are perfectly aligned, G1 - S1 is zero.

A significant difference between the Usuda and Green Bank implementations is the manner in which the S1 signal is defined. The Usuda data decoder outputs only a high frequency clock signal. A separate device counts the pulses of the high frequency clock and generates the S1 signal. The counting device can be signaled to delay the next S1 signal until the time of the next G1 signal. This is done at every satellite acquisition, at the clock setting event. This technique allows use of the VSOP data recorder timing capabilities for keeping track of the wide band data time tag. (The down link and station delays are not included in the clock setting process). The original dt program deduced clock setting events based on the long interval between G1 and S1 at the clock setting event (from the .tic file).

Tests

A program, readDeltaT, was written to print values of the TCFs for use in diagnosis of problems and for plotting. The program is most commonly used within a "C-shell" script called plotDt to make two plots of a TCF file. Figure 1 shows the second of the two plots for the original TCF file. This TCF file has been used in Penticton, and large residual delays and rates occurred during correlation.

Figure 2 shows a the same information for a new TCF for the same tracking pass. Application of this TCF file was known to result is much smaller residual delays and rates (see OVLBI-ES Memo 74).

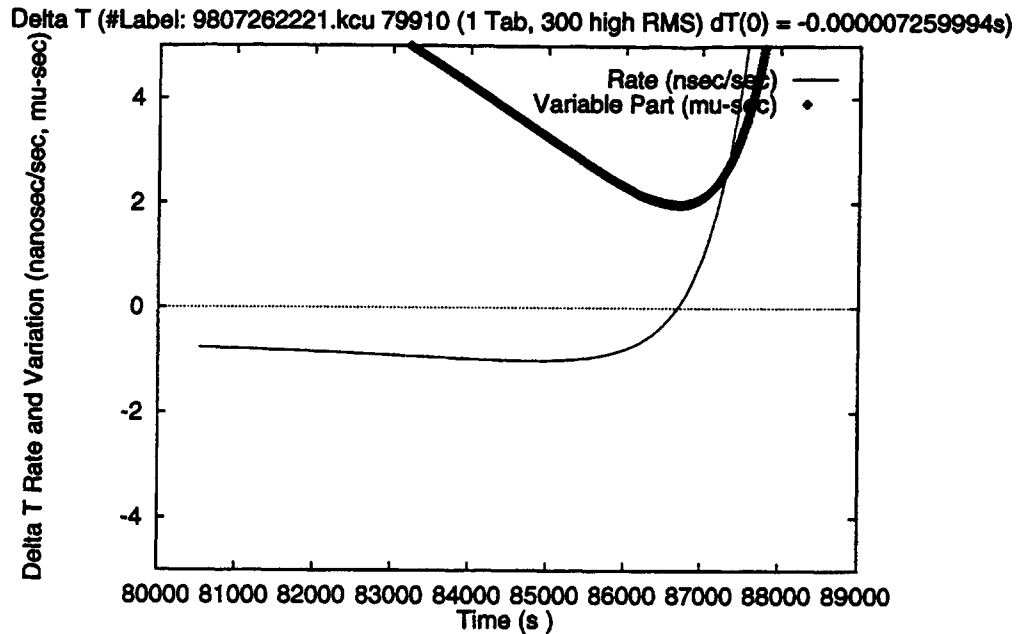


Figure 1: Plot of original Usuda TCF 9807262221.kcu. The X-axis is the time of day (seconds). The Y-axis units are either variable part of TCF value (μ sec) solid line, or rate of change of downlink nano-sec/sec (symbols == thick line).

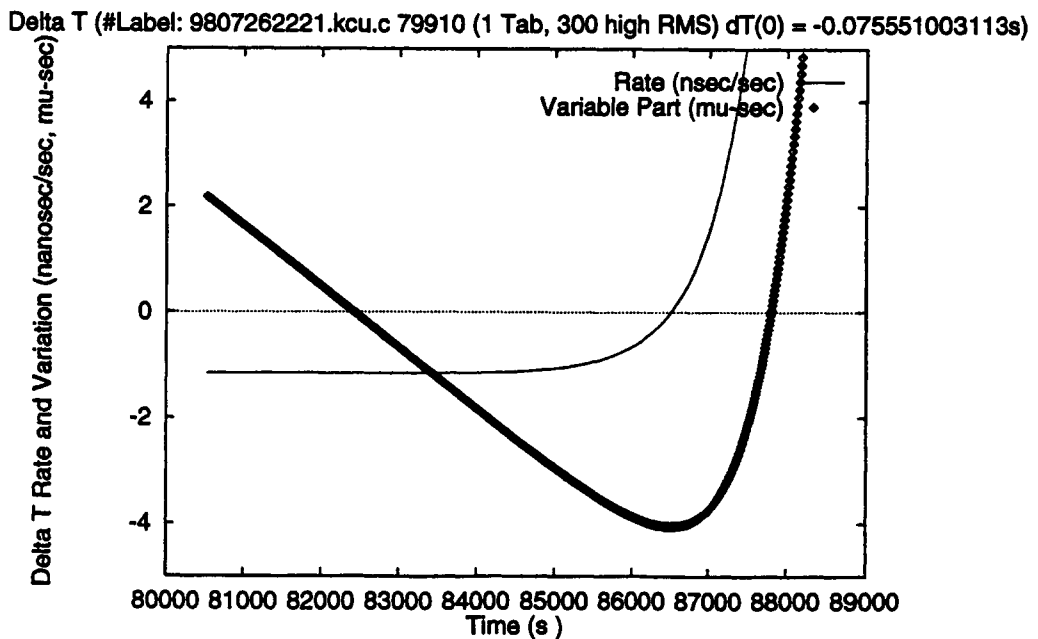


Figure 2: Plot of updated Usuda TCF 9807262221.kcu, created by the original TCF program. The X-axis is the time of day (seconds). The Y-axis units are either variable part of TCF value (μ sec) solid line, or rate of change of downlink nano-sec/sec (symbols == thick line).

Closed Loop

The closed loop mode of the Usuda station appeared to function well and yield small residuals at correlation. This is expected since the major difficulties with the station control were with consistency between up-link and down-link predictions. This is avoided in closed loop mode, since only the down-link prediction is used. the up-link is controlled by zeroing the phase residuals.

Note however that the TCF for the closed loop mode is not filled with zeros. The TCF must include the downlink delay, delay between the 10m antenna and the tape recorder, any tape time errors and the difference between station time and UTC time.

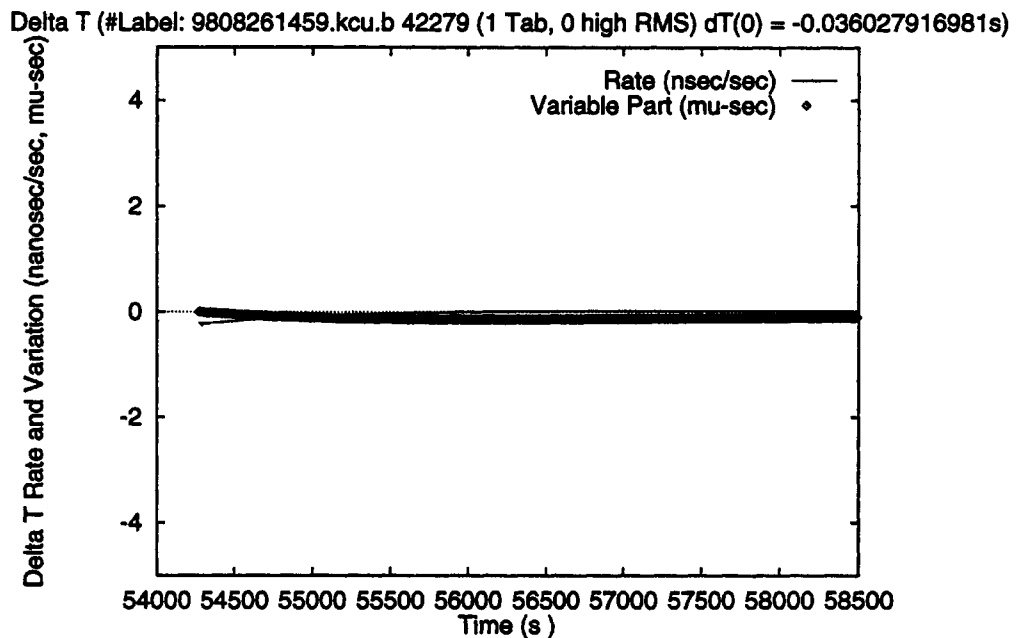


Figure 3: Plot of updated Usuda TCF 9808261459.kcu.b The X-axis is the time of day (seconds). The Y-axis units are either variable part of TCF value (μ sec) solid line, or rate of change of downlink nano-sec/sec (symbols == thick line). This Closed Loop tracking passed used a satellite range-rate prediction from ISAS, but the orbitUsuda program prediction was used to create the TCF. The residual rate is so small that it lies under the variable (integrated) part on this plot.

Recent Tests

Two passes on 99 February 5 and 10 yielded good residual rate results based on correlation done in Socorro. For these tracking passes, the same prediction file (from orbitUsuda) was used to control the satellite up-link and for production of the TCF files. Figures 4 and 5 show the variable part of the TCF correction and the rate of changes of the correction.

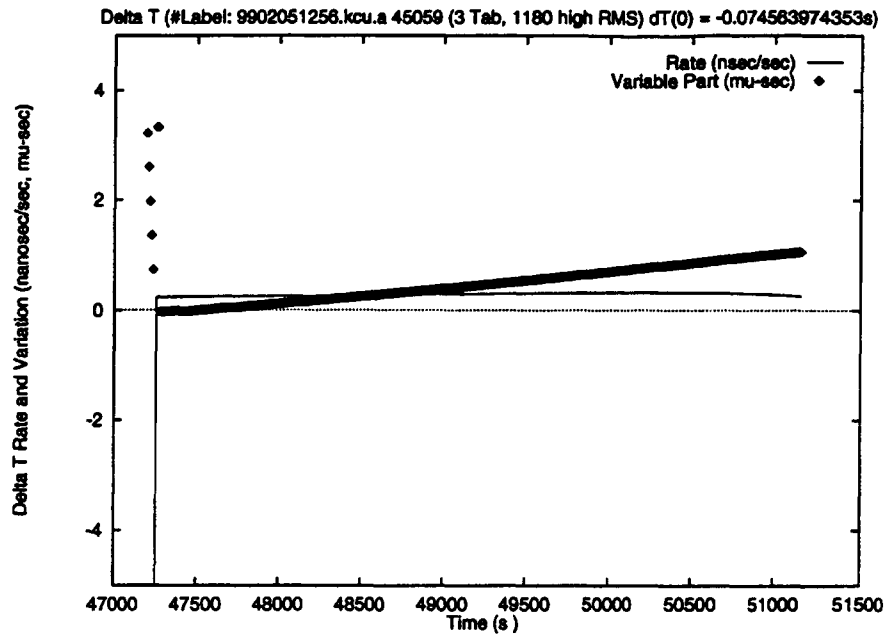


Figure 4: Plot of Usuda TCF 9902051256.kcu.a This tracking passed used a satellite range-rate prediction from the orbitUsuda program. In the beginning of the pass, there is some noise in the clock setting event which is removed shortly after the pass starts. The symbols are plotted every 2 seconds.

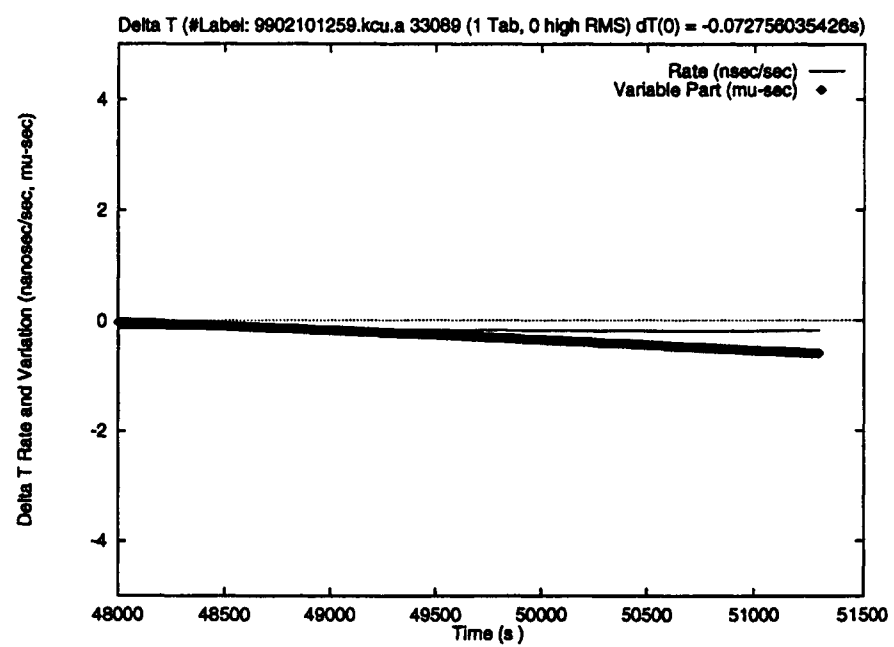


Figure 5: Plot of Usuda TCF 9902101259.kcu.a This tracking passed used a satellite range-rate prediction from the orbitUsuda program.

Conclusions

A new program, `dt` was written to create Usuda Time Correction Files. These files have been successfully used to correlate wide band Space VLBI data. The program functions successfully in 3 modes;

- a) correcting the up-link and down-link predictions to generate TCFs based on improved satellite range rate predictions,
- b) generating the variable parts of TCFs for Usuda closed loop passes, and
- c) simple creation of TCFs based on the same input and output satellite range-rate prediction.

The residual delay rates for the Usuda passes are small enough to allow straight forward correlation of wide-band HALCA data. The residual delays are smaller (generally less than $4 \mu sec$) than for earlier TCFs, but too large to allow easy correlation at all correlators. The cause of these large and variable residual clock offsets is still under investigation.

Appendix A: Closed Loop

The Usuda closed loop mode implemented by Nori Kawaguchi and his student Kazumasa Suzuki is described in this section. The closed loop mode uses the measurements of the downlink phase residual to control the up link frequency in a manner to zero the downlink phase residuals. The closed loop mode is very successful in the case where the a linear model for the change in residual phase applies. In this case, when there are is little acceleration of the range to the satellite, measurements of the residual phase are used to zero correct the up link frequency. One very important aspect should be noted; after initial satellite acquisition, based on the up link prediction, the predicted up link frequency values are not used at all.

The up link frequency values are based on a linear model for the change in up link frequency with time. Every second, the linear model is updated, based on a fit to the down link residuals. Kazumasa Suzuki is writing a memo describing in detail.

The closed loop system worked very well during 1998 July and August, after the link to the satellite is well established. However the initial version of the closed loop software took a relatively long time to stabilize after establishing a two way link to the satellite, more than 5 minutes. During this time, the VLBI data could not be used. Also after any loss of lock to the satellite, the initialization process must be restarted.

During winter 1998, Kazumasa Suzuki made some improvements to the closed loop software, but these improvements were not completed until 1999 February. Unfortunately the closed loop mode was used during 1999 January and the VLBI data were corrupted by unstable link conditions. The latest tests by Kazumasa Suzuki show that the initial acquisition time has been greatly reduced and also a method has been tested for correcting for satellite range acceleration.

Appendix B: dt

The dt program is very similar to a program used for the Green Bank Tracking station. The program is a set of "C" functions written at ISAS in Sagamihara Japan, but uses many functions written in Green Bank and in Socorro, New Mexico.

Sample output of the dt program is below:

```
% dt
```

```
# dt -help
```

```
* Usage : dt [-help] [-option] Usuda_Sub_Pass_Name  
  [-stic]DetectedTicNumber : Specify TIC number of start generation.  
  [-mtic]TicResetTime : Specify TIC reset time (manual setting).  
  [-ulg] : Specify operation log file to be used for TIC detection.  
  [-vsl] : Specify VSOP tape log file to be used for TIC detection.  
  [-mkl] : Specify VLBA tape log file to be used for TIC detection.  
  [-tic] : Specify time interval file to be used for TIC detection.
```


`[-gps]` : Specify GPS time difference file used for calculation.
`[-test]`: run test execution with files in current directory
`[-show]`: show results every `xx` minutes
`[-help]`: print this information

Code structure

All programs developed for OVLBI in Green Bank used the SCCS version control system for keeping track of program versions. This facility has many good features, including the capability of re-creating any old version of a program by supplying a date to the "Check Out" command. I highly recommend this system for software development. The SCCS system is available as a standard part of the Sun Work stations environment.

Many of the `dt` C functions were taken from the VLBA antenna and correlator control project.

The original ISAS TCF program runs under the HP Unix environment. The new `dt` program is run under the Sun Solaris Unix environment. Currently the program is run manually for each Usuda tracking pass. The steps done are first to log into the computer `xo4` and get all the input files for a Usuda pass. There is a script, `musesGet` that is in the sub-directory `glangsto` of the `vsopadm` account that I used to gather all the files and `tar` them into a single file before transferring them to a Sun workstation. On the Sun workstation, the files are `un-tar-ed` and then the new JPL prediction is given the same name as the original ISAS prediction, except that it is given a revision letter. (Ie for Usuda pass 9903221a the original frequency prediction had name `9903221a.fpr`. The new prediction is given name `9903221a.fpr.a`.

For pass 9903221a, `dt` was run with the following arguments:

```
/home/glangsto/bin/dt -test -tic -gps 9903221a.fpr
```

Many diagnostic messages are printed out, including a summary of the components of the corrections included in the TCF. The components include the residual frequency, the clock model and the downlink delay.

Module Name	Function
dt.c	Main routine; creates file names, does numerical integration
cvrtuc.c	Convert string to upper case (VLBA)
doyhhmss2Sec.c	Convert time string to seconds of a year value
findCubic.c	Perform cubic interpolation of input arrays to yield a value
findPhaseJump.c	Return times of high noise in phase residual data
findTroposphere.c	Calculates tropospheric delay using elevation and model
FITSTableW.c	Functions to write a FITS binary table file
hhmss2dbl.c	Convert a time string to double precision seconds
lastRevision.c	Find and open the latest revision of an input file
linearFit.c	Perform a linear fit to two arrays of data
liner.c	Utility functions used in dt; Only liner fit function is used
matinvert.c	Invert a 3 dimensional matrix (used by liner)
mjd2str.c	Converts a modified Julian day number to an ASCII string (VLBA)
openOutRevision.c	Open new version of file (with ".a", etc. extension)
readFreqIF.c	Read up-link and/or down-link IF frequency file, return arrays
readGps.c	Read in file of Station - Gps values, return arrays
readKsu.c	Read the Usuda KSU log and returns array of clock set events
readPhase.c	Read in the phase residual data and return arrays
readPrediction.c	Read the Usuda satellite range prediction return arrays
readTapeLog.c	Read one of three types of tape parameter files
readTic.c	Read the G1-S1 values from the log file, return arrays
readUsudaLog.c	Collect reset time information from Usuda operator log
sortResetTimes.c	Sort clock reset times collected from different files
srclist.c	Look for sub-string in a longer string (VLBA)
str2mjd.c	Convert date string (YYMMDD) to Modified Julian Day number (VLBA)
str2rad.c	Convert time/angle string to radians (Ie "12h00m0s" = 3.14159) (VLBA)
stripWhite.c	Remove white characters from both ends of a string (VLBA)
time2str.c	Convert a radian value to time of day string (VLBA)
today2mjd.c	Reads today's date from Sun operating system, return MJD
troposphericDelay.c	Calculate tropospheric delay, a function of elevation
usudaTable.c	Create TCF table KEYWORD=VALUE entries for FITS tables
yymmdd2mjd.c	Convert a date string (yymmdd format) to modified Julian Day
usuda_sconf.h	Measured Usuda hardware delays and parameters
vlb.h	Include file for some VLBA utility programs; (modified VLBA)
MATHCNST.H	Standard mathematical and astrophysical quantities (VLBA)
STDDEFS.H	Standard data types used by VLBA (VLBA)
MakeDt	Make file which retrieves source from SCCS, compiles and links dt