THE STEERING OF A REAL TIME CLOCK TO UTC(NBS) AND TO UTC

by

J. Levine and D.W. Allan

Time and Frequency Division
National Institute of Standards and Technology
Boulder, Colorado 80303

ABSTRACT

We describe the procedures that we use to define UTC(NBS) and to steer it towards UTC(BIPM) using an averaging process with a time constant of about 1 year. In addition, we describe the hardware and software that is used to steer a physical clock so that its output realizes UTC(NBS) in real time. The method uses a micro-stepper whose frequency offset is updated 5 times per hour by the time scale computer. The corrections applied to the micro-stepper yield a physical tick whose average offset from UTC(NBS) is less than 0.5 nanoseconds. The algorithm can cope with various fault conditions and can also provide remote notification of a fault.

TEXT

UTC(NBS) has been steered to be synchronous with UTC(BIPM) over the last few years by inserting small frequency steps of the order of 1 ns/day at the beginning of each month. The adjustment procedure has a time constant of approximately one year. The steering steps are of the same order as the instabilities of the two time scales involved, hence these steps are not perturbing to the user.

In turn, a physical clock is computer servoed to UTC(NBS) to within less than half a nanosecond. UTC(NBS) is equal to NBS(AT1) plus the above mentioned steering equation. NBS(AT1) is not synchronized or syntonized to anything, but is designed for optimum frequency stability. It is the output the NBS time-scale algorithm, which is designed to be better than any of the contributing member clocks in the NBS ensemble in the short-term or in the long term. Hence, in theory, the above mentioned physical clock will have better stability in the long term than any of the contributing clocks.

Figure 1 is a block diagram of the NBS clock ensemble measurement system. The time differences of the clocks are measured with a precision of about 3 picoseconds and then processed every two hours for the computation of NBS(AT1) and UTC(NBS). Once a month, TA(NBS) is computed using a Kalman algorithm. NBS(AT1) is computed using the NBS algorithm described in these proceedings by Dr. Weiss. Once a month the equation between AT1 and UTC(NBS) is changed so that the rate of UTC(NBS) is synchronized in long-term with UTC(BIPM). The rate changes are always kept small enough so that they are basically within the noise of NBS(AT1) so that these changes have no net effect on our user community. That is to say the stability of UTC(NBS) is essentially as good as the stability of NBS(AT1). A micro-steeper output is driven by one of the clocks in the ensemble, and this output is steered by a computer feedback system to our official software calculated clock, UTC(NBS). As will be seen this is accomplished to within better than half a nanosecond. The purpose of this paper is to describe the benefit of these servo systems and how they work. TA(NBS), in contrast, is syntonized to the frequency calibrations given by the primary frequency standards of the NBS.

Figure 2 shows UTC and UTC(NBS) with respect to a pseudo scale offset from NBS(AT1) by -20 ns/day. This rate offset was chosen to make the pseudo scale approximately syntonous with UTC. The UTC values are computed from data from the BIH (and later BIPM) using Loran timing signals (before the arrow) and GPS common-view measurements (afterwards). UTC(NBS) is defined in terms of NBS(AT1) by an equation whose parameters are administratively adjusted on the first day of each month to track UTC. The administrative adjustment steers the rate of UTC(NBS) so that the time difference between UTC and UTC(NBS) will be removed after 1 year using a Kalman extrapolation of the rate of UTC.

Figure 3 shows the most recent portion of figure 2, beginning when we changed from Loran timing to GPS common view, on an expanded scale. The phase shift between UTC (dotted curve) and UTC(NBS) (solid curve) is due to the 1-year time constant in the administrative adjustment algorithm. The steering equation servo to synchronize UTC(NBS) to UTC was turned on when the GPS common-view data became available -- effectively at the beginning of the data plotted in Figure 3. As one can see from the Figure, this time was a stress test case: the time difference was a few microseconds, the frequency difference was significant off from zero and of the sign to drive the time difference error further away. One of the messages of this Figure is that the servo works even in this adverse condition.

Figure 4 is a plot of the error (in seconds) between UTC(NBS) and a clock that is steered to UTC(NBS) using a computer-driven micro-stepper. The steering algorithm applies a frequency correction to the micro-stepper every 12 minutes computed from the measured difference between the output of the micro-stepper and the calculated value of UTC(NBS) and the integral of this time offset. The loop gain (i.e., the time that would be necessary for the applied frequency correction to remove the measured error) is 1.5 measurement intervals or 18 minutes. This gain was determined empirically as a compromise between rapid response to frequency changes and minimizing overshoot following large corrections.

The servo has additional logic to cope with large errors (produced by start-up transients, for example). This additional logic uses the previous estimates of the frequency of the actual clock driving the micro-stepper in a double feed-forward arrangement to estimate the asymptotic rate that must be applied to the micro-stepper even when the error is large. The first feed-forward loop uses the maximum slew rate of the micro-stepper. The integrator is clamped during this period to minimize the subsequent overshoot that would otherwise result from the integration of the very large error. This loop is automatically disabled when the error has been reduced to less than 100 nanoseconds. At that point, a second feed-forward loop estimates the asymptotic frequency offset using the previous values of the frequency of the input clock, and this value is loaded into the integrator (which is still kept locked). When the error has been reduced to less than 5 nanoseconds, the normal algorithm is enabled and the loop locks within a few measurement cycles. Although this response is totally causal, it could not be implemented as a single stable loop since it depends upon a knowledge of the frequency of the clock that is used to drive the micro-stepper.

Figure 5 shows a plot of the auto-correlation function of the time-series shown in figure 4. The value at zero lag is a measure of the total variance in the time-series, while the values at other lags are an indication of the spectral content. Note that the number of samples is small for the largest lags, so that the uncertainties of these estimates are correspondingly large.

Figure 6 is a plot of the power-spectrum of the data shown in figure 4. The statistically significant lines at periods of 295 minutes and 48 minutes appear quite often in different data sets. We do not know their cause, although some environmental factor is quite likely.

Figure 7 is a plot of frequency stability of the physical clock with respect to the software clock and one sees again the stability of less than half a nanosecond. The micro-stepper shown in figure 1, which is manually steered, is separate from the computer controlled one explained above. The manual steering occurs about once a week, which keeps its output within approximately ten nanoseconds of the software clock, UTC(NBS).

Figure 8 illustrates the need for long-term steering of any primary timing center. In this case we have probably what is the best primary cesium clock in the world; left free running it gradually diverges from UTC(BIPM). This is, of course, the nature of physics and of natural stochastic processes. Any two clocks will not stay synchronous unless there is some feedback mechanism.

Figure 9 shows the fractional frequency stability of UTC(BIPM) versus PTB Cs 1. Note the following: 1) The drop off of the stability at approximately one year would indicate the evidence of an annual term between these two time scales; 2) The stability at one year would imply that even if one steered only once per year, the time synchronization could be kept within about a microsecond. Clearly steering more often would allow one to steer well within the microsecond limit, which is the current international goal. For comparison purposes figure 10 shows the stability of PTB Cs 1 against NBS(AT1). One will note that the stability curves are essentially identical even though TAI and NBS(AT1) are nearly independent time scales. UTC(BIPM) and TAI have the same frequency stability by construct.

Figure 11 shows the frequency stability of the steered UTC(PTB) versus PTB Cs 1 with the steering being accomplished once a week. It is evident that the residuals can be well characterized by white noise PM with a standard deviation of about 15 or 16 nanoseconds. As has been pointed by Dr. de Boer this can clearly be reduced if one shortens the steering time interval.

To try to get an independent estimate of the stability of NBS(AT1) it was analyzed against PTB Cs 1 and against a GPS satellite vehicle clock ensemble. A three cornered hat stability estimate plot is shown in figure 12. It becomes obvious that one of the problems we need to deal with in steering timing centers are environmental variations such as annual terms -- the best solution being to fix the problem that causes the annual variations in clocks. A less desirable but near term solution may be to compensate for such effects.

CONCLUSION

Real time clocks can be made to mirror software clocks with arbitrarily high accuracy. Within the digital era in which we live software clock values may be more useful in many applications than physical clock values. If one can, in fact, always appreciate in the software calculation a better time than can be physically realized (at least in the long-term) then it would be advantages to use calculated values as the best references. If a physical clock is needed for a particular application, then it could be servoed appropriately.

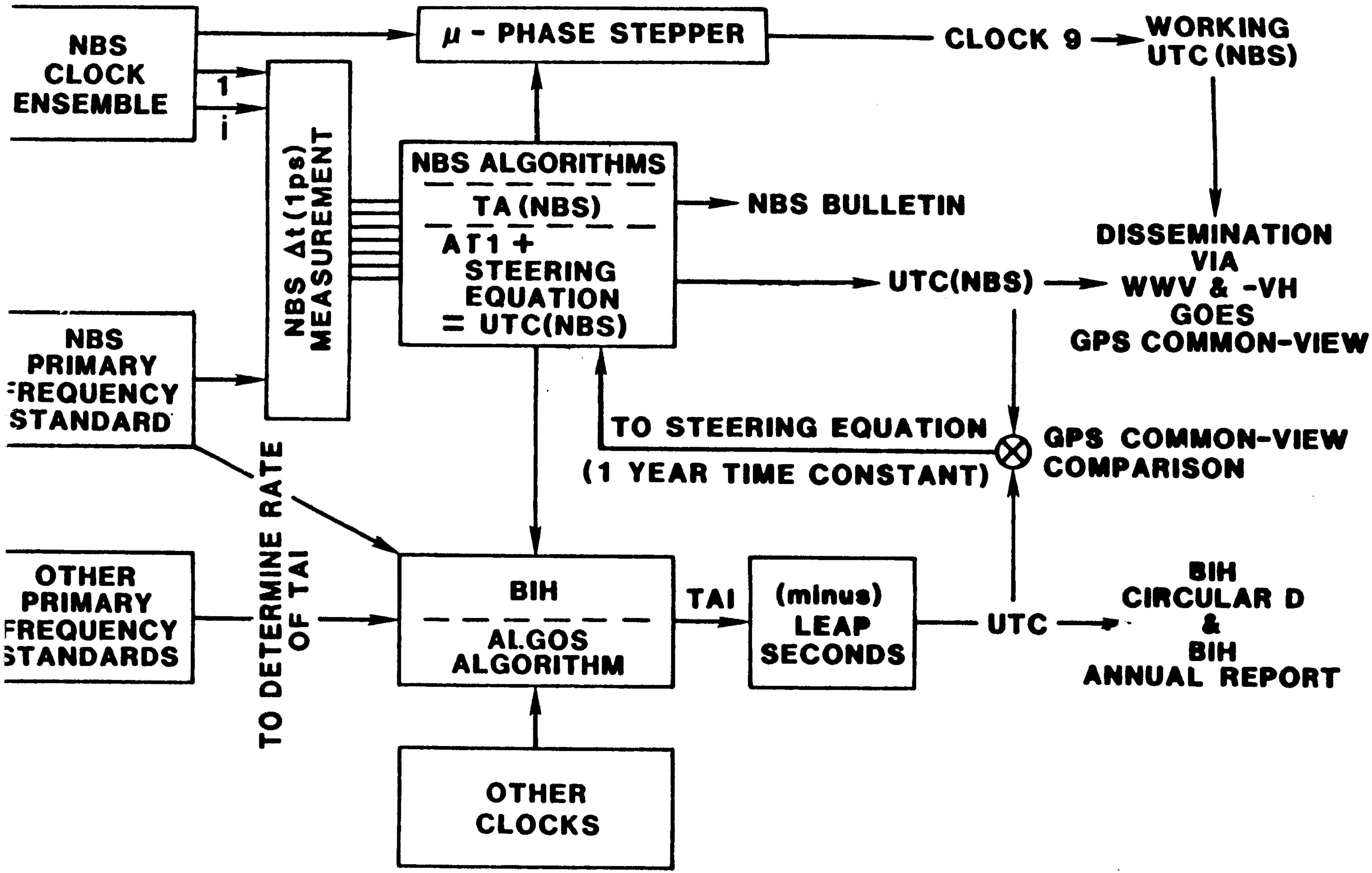
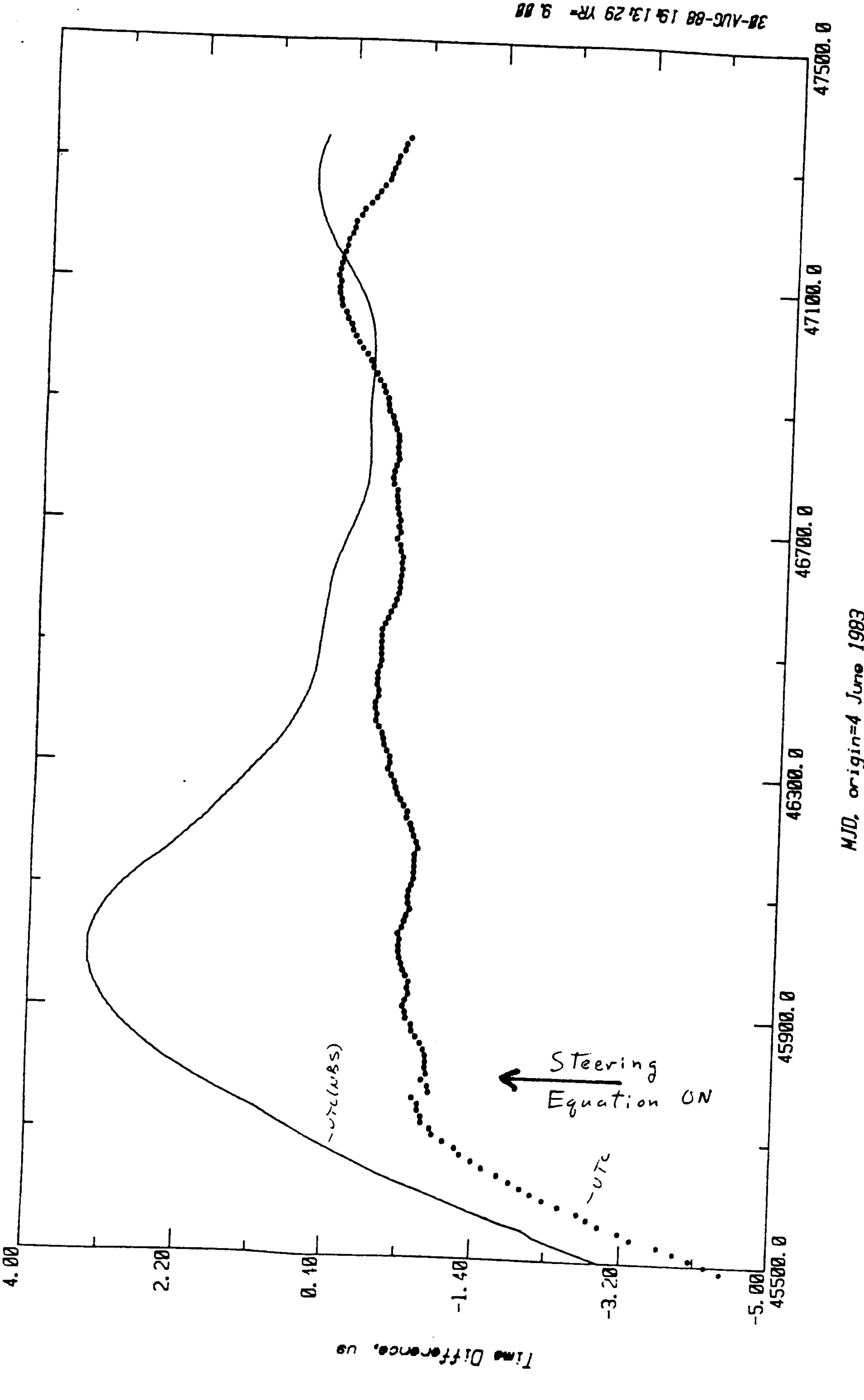

NBS CLOCK ENSEMBLE
1
i
μ - PHASE STEPPER
CLOCK 9
WORKING UTC (NBS)
NBS Δt (1ps) MEASUREMENT
NBS ALGORITHMS
TA (NBS)
AT1 + STEERING EQUATION = UTC(NBS)
NBS BULLETIN
UTC(NBS)
DISSEMINATION VIA WWV & -VH GOES GPS COMMON-VIEW
NBS PRIMARY FREQUENCY STANDARD
TO STEERING EQUATION
(1 YEAR TIME CONSTANT)
GPS COMMON-VIEW COMPARISON
TO DETERMINE RATE OF TAI
OTHER PRIMARY FREQUENCY STANDARDS
BIH
ALGOS ALGORITHM
TAI
(minus) LEAP SECONDS
UTC
BIH CIRCULAR D & BIH ANNUAL REPORT
OTHER CLOCKS

FIGURE 1



UTC and UTC(NBS) - AT1 + 20 ns/day

FIGURE 2

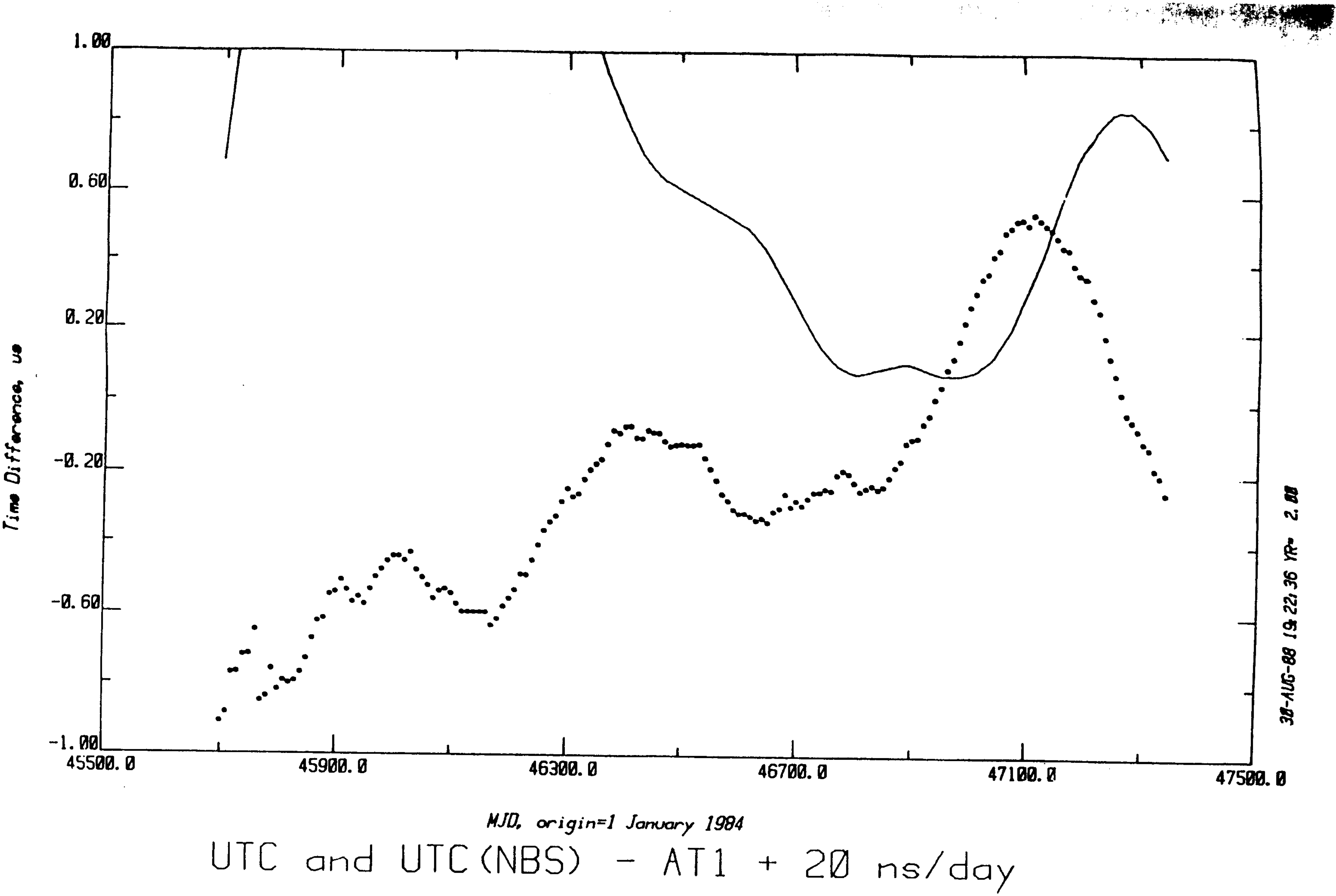

1.00
0.60
0.20
-0.20
-0.60
-1.00
Time Difference, us
45500.0
45900.0
46300.0
46700.0
47100.0
47500.0
MJD, origin=1 January 1984
30-AUG-88 19:22:36 YR= 2.00
UTC and UTC(NBS) - AT1 + 20 ns/day

FIGURE 3

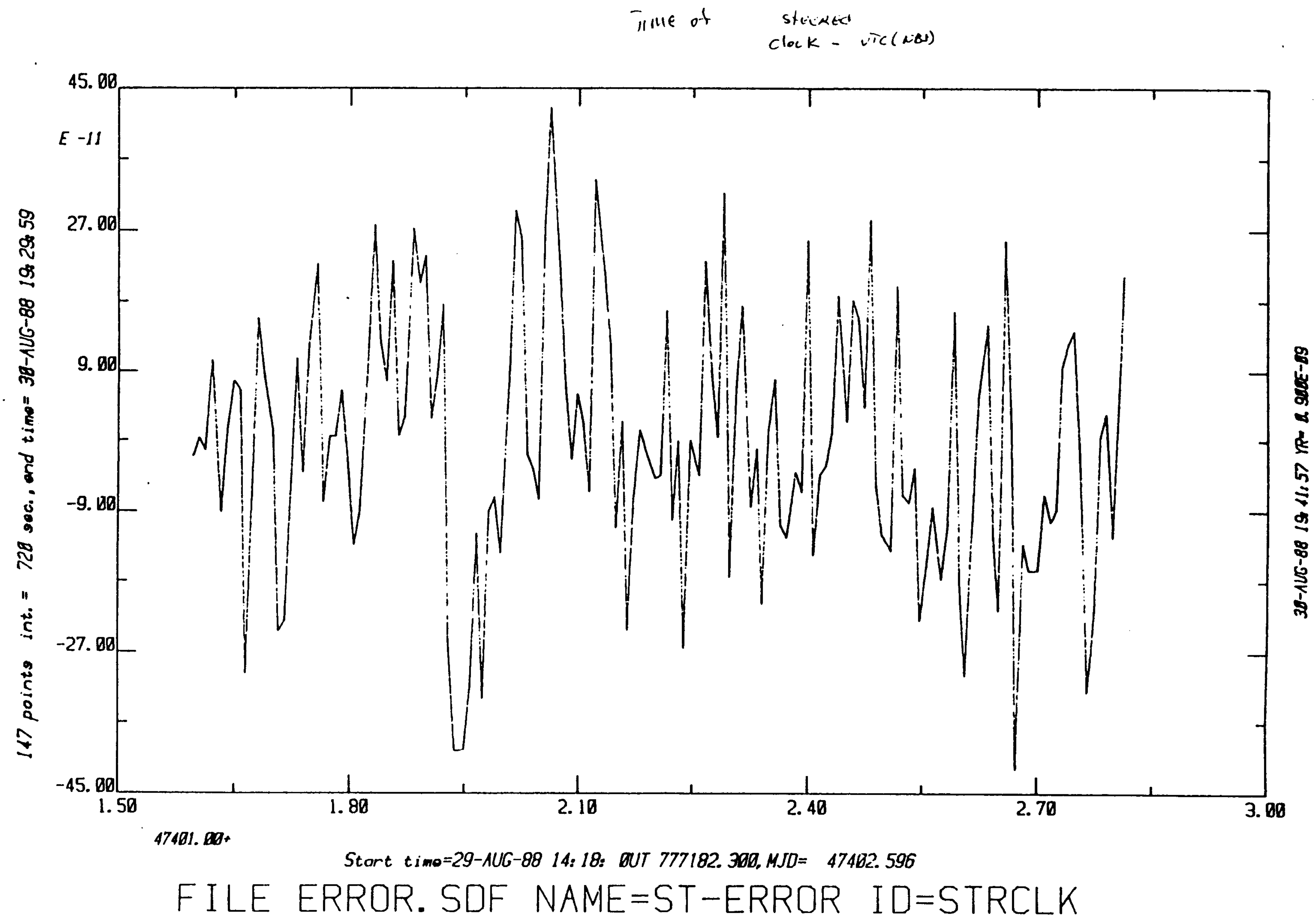
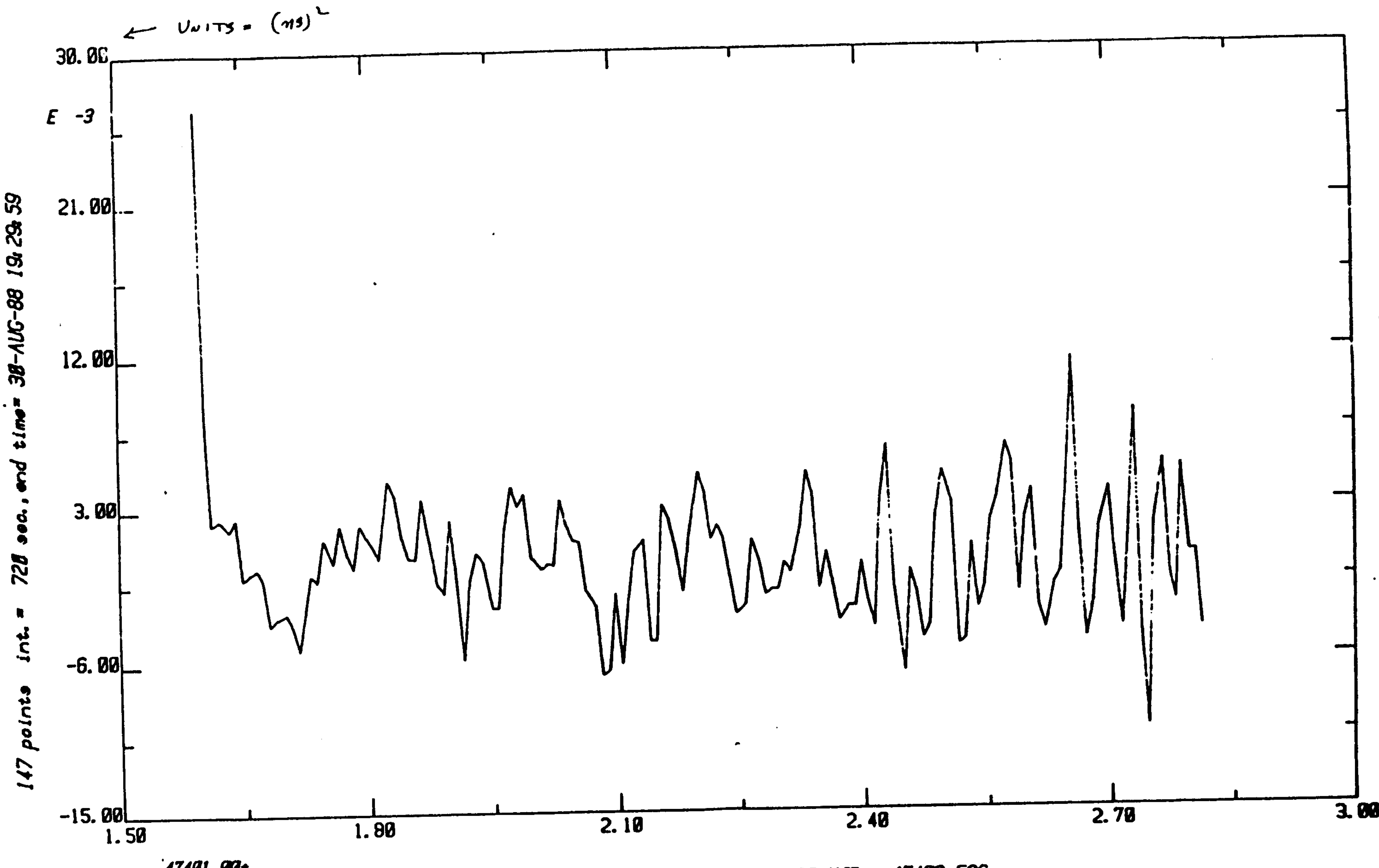
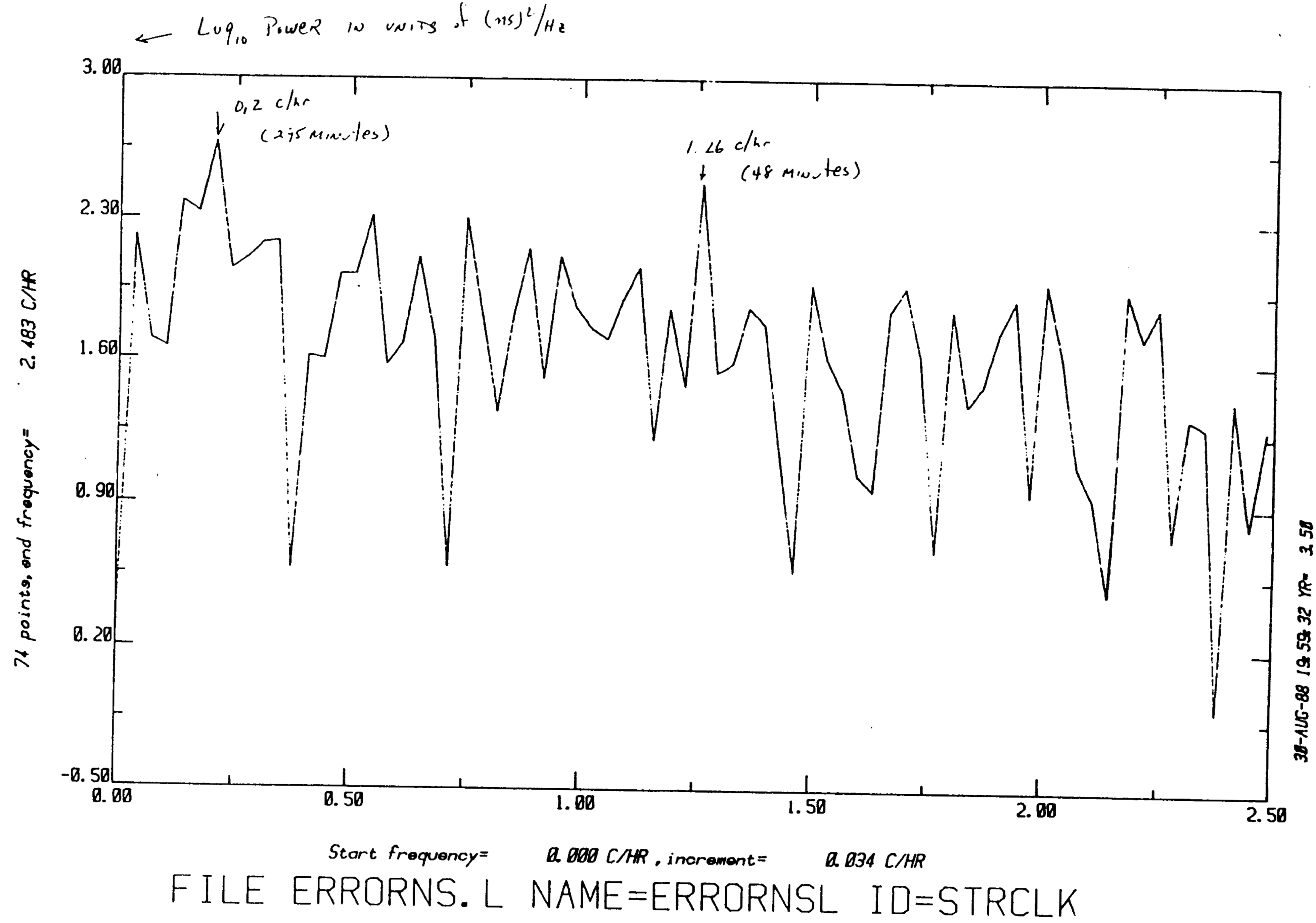

Time of stecked clock - UTC(NBS)
45.00
E -11
27.00
9.00
-9.00
-27.00
-45.00
1.50
1.80
2.10
2.40
2.70
3.00
147 points int. = 720 sec., end time= 30-AUG-88 19:29:59
30-AUG-88 19:41:57 YR= 0.900E-09
47401.00+
Start time=29-AUG-88 14:18: 0UT 777182.300, MJD= 47402.596
FILE ERROR. SDF NAME=ST-ERROR ID=STRCLK

FIGURE 4

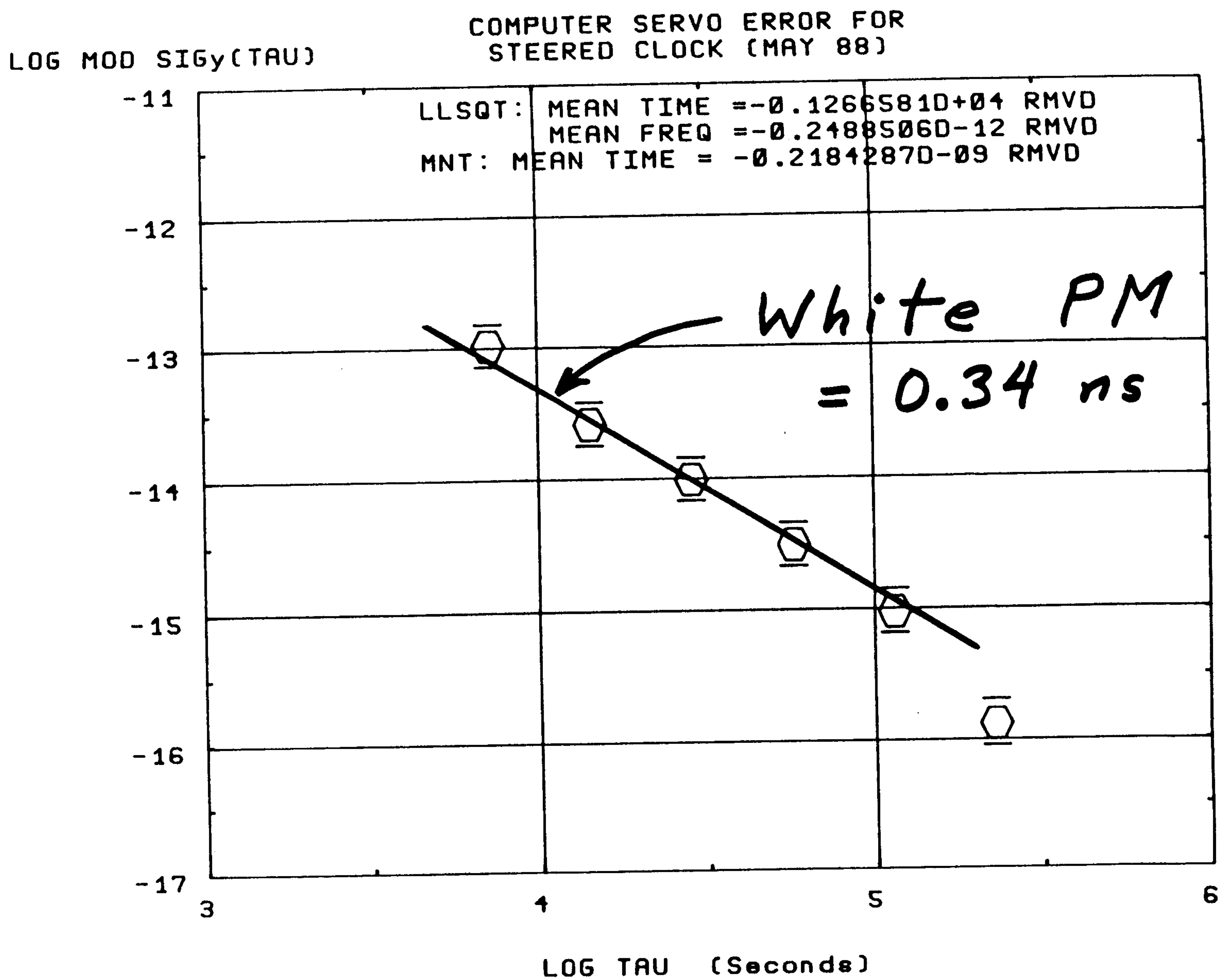


FILE AUTOCOR.S NAME=X CORREL ID=ERRORN



← Log10 Power in units of (ms)²/Hz
3.00
2.30
1.60
0.90
0.20
-0.50
0.2 c/hr
(275 minutes)
1.26 c/hr
(48 minutes)
0.00
0.50
1.00
1.50
2.00
2.50
74 points, and frequency= 2.483 C/HR
30-AUG-88 19:59:32 YR= 3.50
Start frequency= 0.000 C/HR , increment= 0.034 C/HR
FILE ERRORNS.L NAME=ERRORNSL ID=STRCLK

FIGURE 6


COMPUTER SERVO ERROR FOR
STEERED CLOCK (MAY 88)
LOG MOD SIGy(TAU)
LLSQT: MEAN TIME =-0.1266581D+04 RMVD
MEAN FREQ =-0.2488506D-12 RMVD
MNT: MEAN TIME = -0.2184287D-09 RMVD
White PM
= 0.34 ns
-11
-12
-13
-14
-15
-16
-17
3
4
5
6
LOG TAU (Seconds)

FIUGRE 7

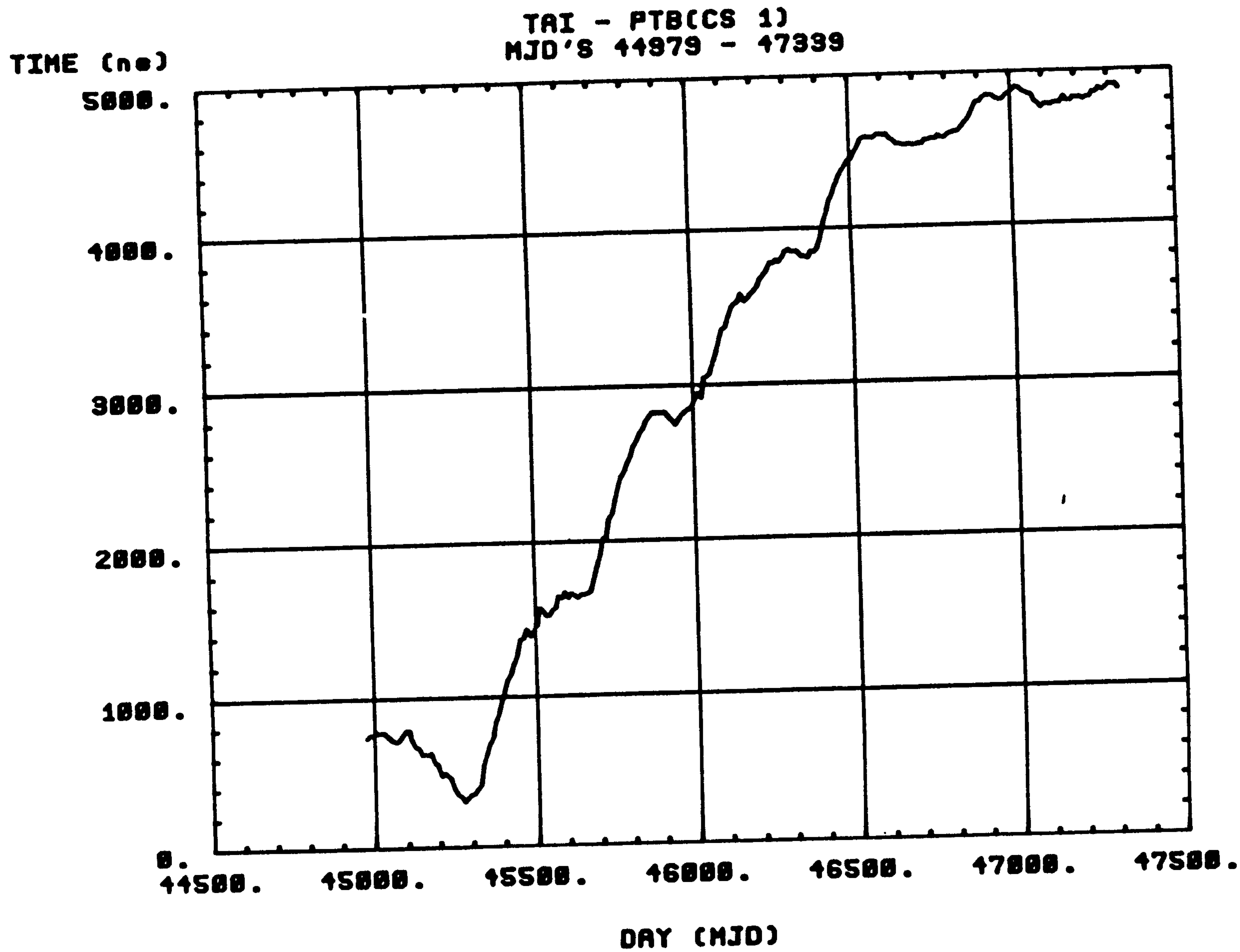

TAI - PTB(CS 1)
MJD'S 44979 - 47339
TIME (ns)
5000.
4000.
3000.
2000.
1000.
0.
44500.
45000.
45500.
46000.
46500.
47000.
47500.
DAY (MJD)

FIGURE 8

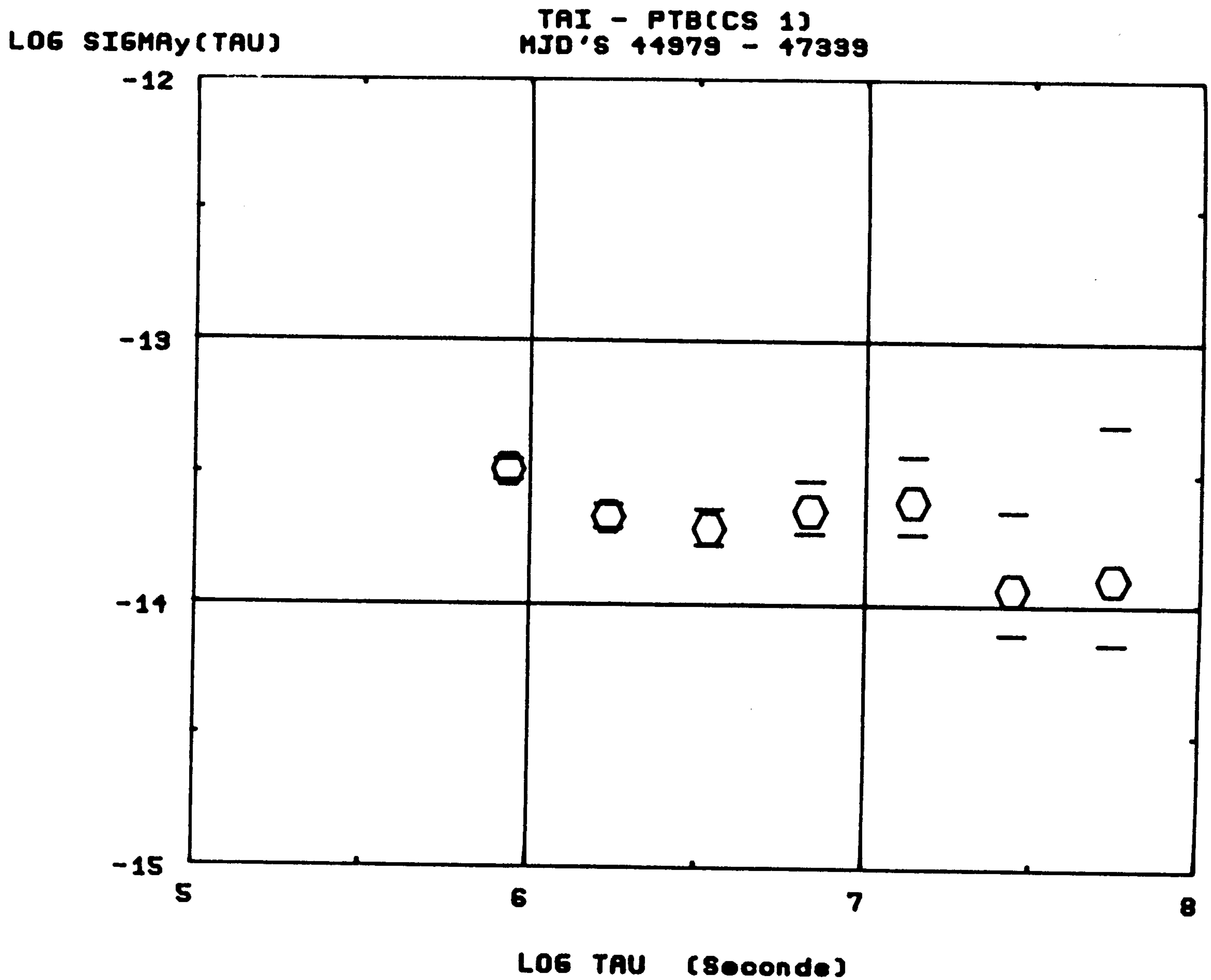

LOG SIGMAy(TAU)
TAI - PTB(CS 1)
MJD'S 44979 - 47339
-12
-13
-14
-15
5
6
7
8
LOG TAU (Seconde)

FIGURE 9

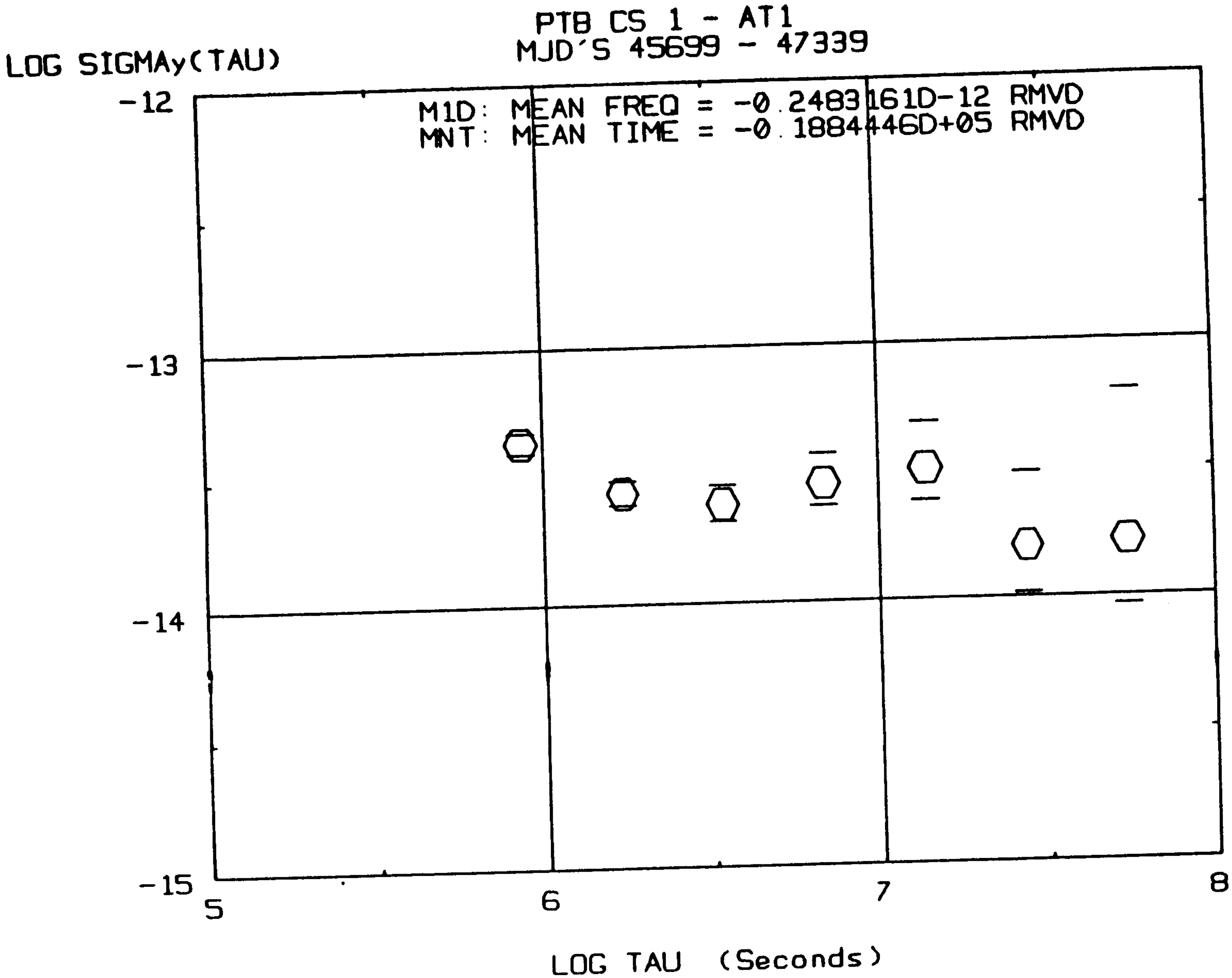

PTB CS 1 - AT1
MJD'S 45699 - 47339
LOG SIGMAy(TAU)
M1D: MEAN FREQ = -0.2483161D-12 RMVD
MNT: MEAN TIME = -0.1884446D+05 RMVD
-12
-13
-14
-15
5
6
7
8
LOG TAU (Seconds)

FIGURE 10

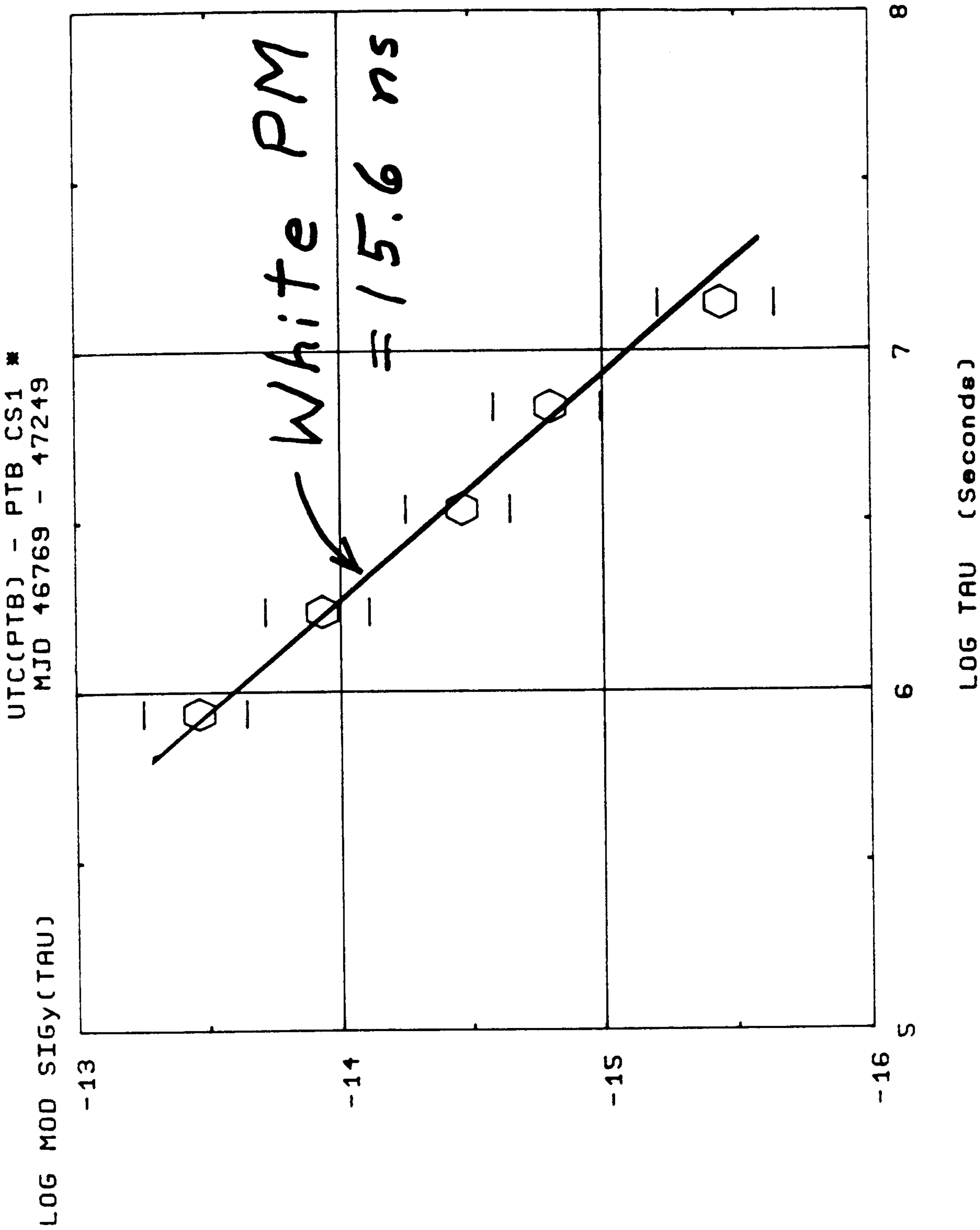
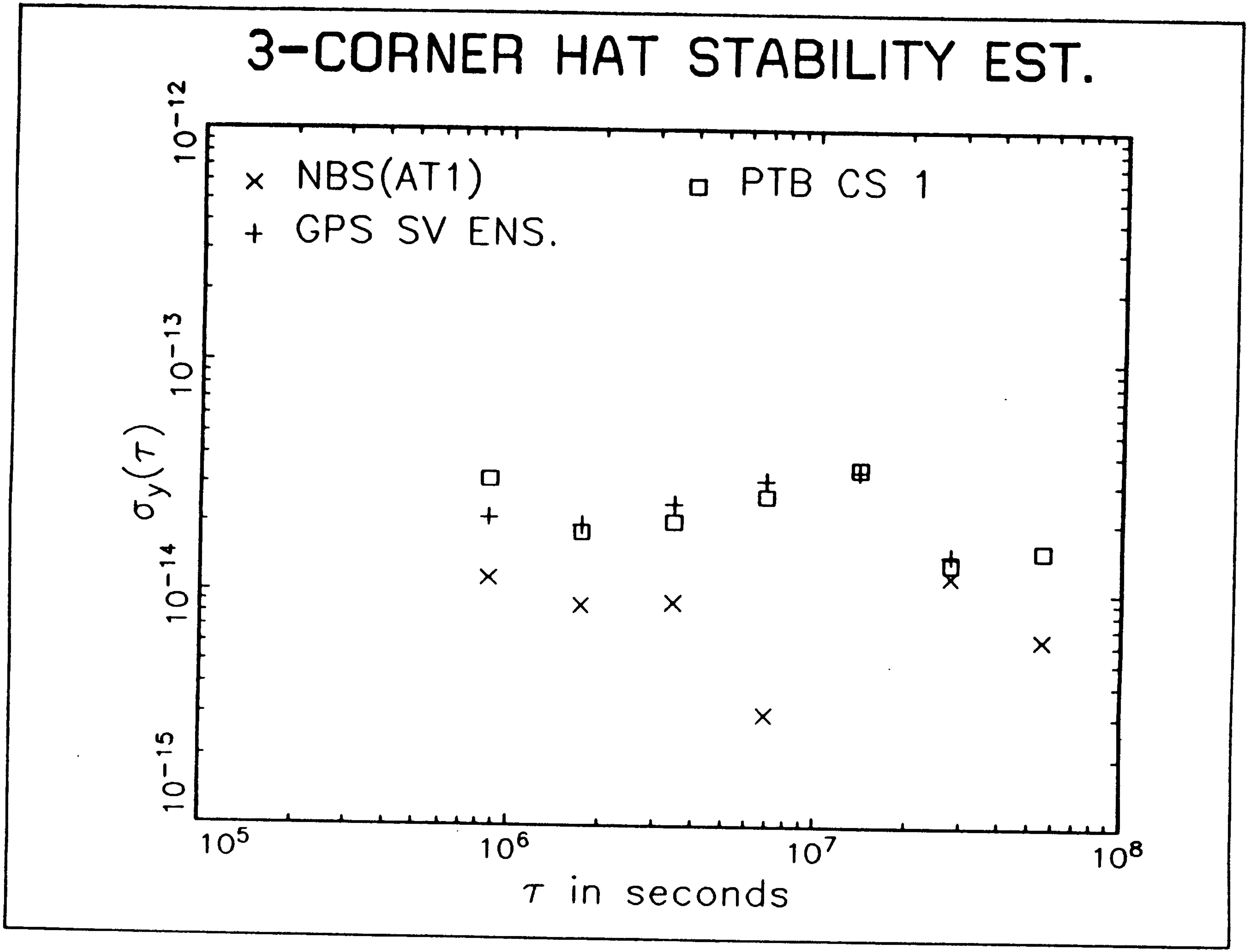

UTC(PTB) - PTB CS1 *
MJD 46769 - 47249
LOG MOD SIGy(TAU)
LOG TAU (Seconds)
-13
-14
-15
-16
5
6
7
8
White PM
= 15.6 ns

FIGURE 11



3-CORNER HAT STABILITY EST.
× NBS(AT1)
+ GPS SV ENS.
□ PTB CS 1
10^{-12}
10^{-13}
10^{-14}
10^{-15}
$\sigma_y(\tau)$
10^5
10^6
10^7
10^8
τ in seconds

FIGURE 12