42nd Annual Frequency Control Symposium - 1988

ISRAEL'S NEW SYNCHRONIZED TIME SCALE, UTC(INPL)

A. Shenhar and W. Litman
National Physical Laboratory of Israel
Jerusalem 91904, Israel

A. Lepek† and A. Citrinovitch
Time and Frequency Limited
Holon 58117, Israel

D.W. Allan and T.K. Peppler
National Bureau of Standards
Boulder, CO 80303 USA

Summary

The National Physical Laboratory of Israel (INPL) together with Time and Frequency Limited (TFL) and NBS is building a software clock to be used as the Israeli national time base, UTC(INPL). The software clock is based on several commercial Cs clocks (HP and TFL) whose outputs are routed sequentially through a TFL programmed switch into a time interval counter. The phase differences are processed in a personal computer using a procedure adopted from NBS to generate the software clock. The system also has an input from a common view GPS receiver for time comparisons. Steering of the software clock is possible from the PC keyboard.

The system has the capability to compare any additional clock (such as units under calibration) against the software clock, thus upgrading the accuracy of calibration.

This paper presents the principles of generating the software clock, its performance and the method to compare it to UTC and UTC(NBS). The paper also presents the optimization procedures for synchronizing and syntonizing UTC(INPL) with UTC as a coordinated time scale.

Introduction

The UTC(INPL) is generated presently from one master Cs clock, HP-5061A* option 004 which is backed-up by two similar clocks and one TFL-5440A. The clocks are temperature stabilized within ± 0.5 °C. The system is equipped as well with two common view GPS receivers and the feedback for comparisons is made through the Bureau International des Poids et Mesures(BIPM) Circular-T. The stability over 10 days is better than 1e-13. The software clock system, which was installed in April 1988, runs now in parallel and upon completion of the run-in phase will be declared as the new UTC(INPL) generating system.

The software clock system generates its time base from the above four clocks by using an algorithm which filters the weighted average outputs of each clock. The outputs of the system, which are obtained in real time, are the time and frequency differences of each of the ensemble clocks from the software clock.

The advantages of using the software clock system over the previous one are listed below (some of these are not yet fully implemented): upgrades stability by ensemble averaging; filters individual clock's anomalies; takes care automatically of a failed clock, thus increasing reliability. Using a software clock enables additional functions: steering to UTC, comparison to other clocks, lock of a hardware clock onto the system and automated comparisons using common view GPS.

Hardware

The guidelines in designing the system were high flexibility for future upgrades in a way that uses as much as possible existing instrumentation in a time and frequency lab.

Figure 1 is a block diagram of the software clock and Figure 2 is a photo of the complete UTC(INPL) system. The only additional piece of equipment that was custom designed is the switch matrix in Figure 1. The mode of operation is as follows: The phase differences of each pair of clocks in the ensemble are measured sequentially using the time interval counter. The divider is used to divide by two and generate a lower frequency at the "start" input of the counter to eliminate phase modulo ambiguity. The construction of the switches enables one to feed any clock at both inputs to the counter, thus making possible measurements such as system jitter and delays. Figure 3 is a short summary of the hardware features.

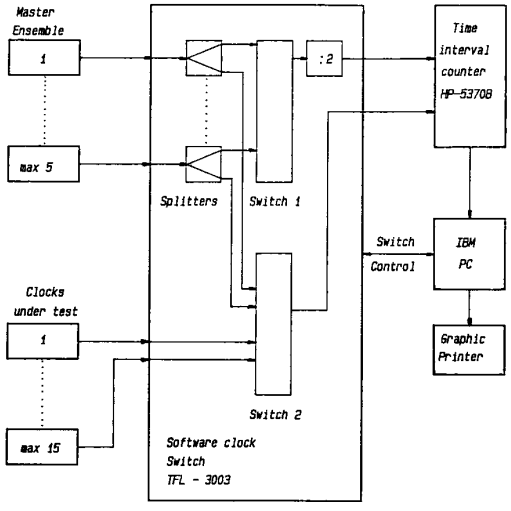

Fig. 1 A Block Diagram of the Software Block System.

†A. Lepek is a consultant to INPL.

*Identification of a commercial company does not imply recommendation or endorsement by the National Bureau of Standards, nor does it imply that any identified entity is the only or the best available for the purpose.

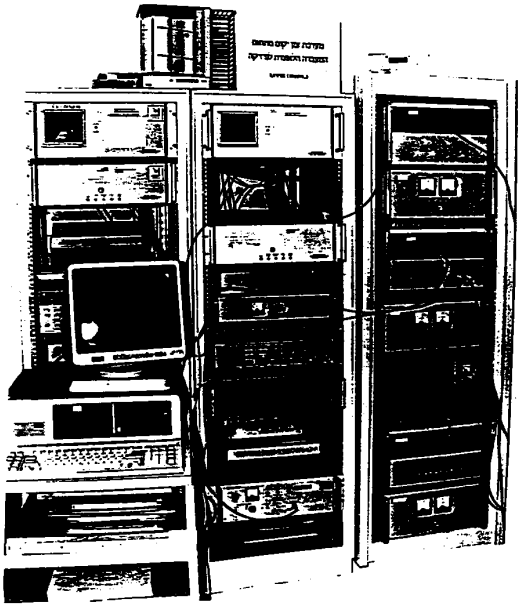

Fig. 2. The Complete Set-up of the UTC(INPL) System.

Maximum ensemble clocks	5
Maximum clocks under test	15
Isolation between inputs	120 dB
Ensemble clocks' frequency	1 pps
Frequencies of tested units	10 MHz and sub-harmonics down to 1 Hz
Switching time	1 ms
Jitter	less then 30 ps
Computer synchronization output	by the counter's
Information storage	20 Mb hard disk with diskettes backup

Figure 3. A summary of the hardware features.

The Main Algorithm

The algorithm is based on a simple procedure used successfully at NBS; it is described in [1] and is outlined here shortly. The inputs to the algorithm are the measured phases Xij of all clock pairs, with time intervals between measurement cycles of TO. TO was chosen to be 3 hours, an interval long enough to eliminate the influence of the system's jitter on the computed frequency offsets.

The procedure starts by assigning a high weight to the present master clock (77%) and equal small weights to the other clocks. The time constant of the exponential filter which allows the change of the weights (Nr in ref [1] notation) was chosen to be 15 days. The master's phase difference from the software clock at start-up is defined to be 0. Good convergence was obtained when deriving the weights from the square of the second differences of the phases; that is, the weights used in the next iteration to compute the average frequencies are obtained for each clock as follows:

$$E(t+T1) = [SSD(t) + Nr*E(t)]/(Nr+1) \quad (1)$$

$$W(t+T1) = k/E(t+T1) \quad (2)$$

where T1 is the time between iterations chosen in the present procedure to be 24 hours.

SSD is the squared second difference of each clock from the software clock,
E is the filtered squared second difference,
W is the weight to be used in the next iteration,
k is a dynamic normalization factor which keeps the sum of all W's equal to 1.

Once the offsets Yio from the software clock are computed, they are exponentially filtered with a dynamic time constant, Mi, which is related to the time interval (multiple of TO) for which Allan variance is a minimum (see ref. [1]), as follows:

$$\hat{Y}io(t+T1) = [Yio(t+T1)+Mi*\hat{Y}io(t)]/(Mi+1) \quad (3)$$

$$\hat{X}io(t+T1) = Xio(t) + \hat{Y}io(t)*T1 \quad (4)$$

where $\hat{Y}io$ and $\hat{X}io$ are the next predicted values of the frequency offset and phase respectively of the i^{th} clock. Xio (the final phase difference from the software clock) is obtained as follows

$$Xio(t) = sum(j)\ \{Wj(t-T1)*[\hat{X}jo(t) - Xji(t)]\} \quad (5)$$

where Xji is the measured phase of clock j minus the phase of clock i.

The outputs of the system are the phases Xio and frequency offsets, $\hat{Y}io$ for each ensemble clock i at intervals T1. For monitoring purposes the two-sample Allan variance is computed for time intervals TO. The reasons to choose T1 = 24 hours were:

a) It differs from TO (=3 hours) by a power of 2, which makes it convenient in Allan variance calculations.

b) The diurnal cycle influences are minimized by taking one measurement every day at the same time.

c) It coincides with the BIPM circular-T with the measurements taken at 00 hours UTC.

Figure 4 shows the filtered SSD (see above) which is the basis to compute the weights. They evolve restricted by the time constant Nr from their initially assigned values.

Figure 5 shows the change of the weights with the time constant. The weight of clock 0 (the previous master) was initially assigned a high value to

enable a smooth start-up (actually the SSD were assigned initial values).

Figure 6 shows the Yio for the first month after the start-up. The Yio were given some initial arbitrary values which evolve with the time constant Nr towards the right values.

Figure 7 shows the evolution of the phases Xio. In this figure initial values are subtracted to allow insertion of all graphs into one scale.

Figure 8 shows the square root of the Allan variance of one clock. The value at time interval of 12 hours is sometimes higher than expected from the adjacent values and supports the probability of diurnal influences. As noted above,the choice of T1 considered this problem.

T1 being so long may impose some inconvenience from the point of view of accessibility, such as when frequent comparisons are needed or a lock of a less stable hardware clock to the software clock is desired. To overcome this problem it is possible to run the algorithm in parallel with two values of T1 i.e. T1a = 24 hours T1b = 3 hours, all the time constants generated with the run with T1a are used for the run with T1b. Since. at any time, the difference between Xioa and Xiob obtained in this way is known, the software clock can be hardware accessed at the shorter time intervals, T1b.

Steering

The present system enables steering of its output to compensate for deviations from UTC. The method follows the guidelines described elsewhere [2] for steering to GPS.

The steering control loop is described in Fig. 10. The task is to keep the systems steered phase and frequency as close as possible to UTC. In the steered system, Xiutc, the phase of clock i with respect to UTC is

$$Xsutc = Xiutc - Xio - Xos \quad (6)$$

$$Xis = Xio + Xos \quad (7)$$

where

Xsutc is the steered output with respect to UTC,
Xos is the steering phase added to the software phase.
The phase of clock i from the steered system is now Xis.

At the start-up, for the previous master clock we define Xlo and Xos = 0, and therefore Xiutc = Xsutc.

To obtain convergence of Xsutc to zero we change Xos at the rate Ys defined for the next iteration as follows

$$Ys(t+T) = \{m*Ys(t) - [Xsutc(t) - Xsutc(t-T)]/T\}(m+1)-\ell*Xsutc/T \quad (8)$$

where

m is the exponential time constant for filtering Ys, T is 10 days, the BIPM Circular-T time interval, ℓ is a parameter which defines the phase steering.

The next Xos is defined simply by

$$Xos(t+T) = Ys(t+T)*T+Xos(t). \quad (9)$$

The values for the parameters m and ℓ were obtained using the simulation of ref [2] with BIPM Circular-T data files. The simulation assumed very conservative flicker FM levels for the INPL ensemble and for UTC (1e-13 and 2e-14 respectively). The former is about a factor of 2 larger than it will be and the latter is degraded because we have to use a predicted value to estimate UTC for the current time for servo control.

The parameter choices involve trade-offs between the best short-term or long-term stability. Figure 10 shows the simulated UTC performance used in the calculations and Figure 11 that of the clock ensemble.

Figure 12 is a typical output from which the parameters m and ℓ were chosen. The values ℓ = 6.0 and m = 3.2 were a reasonable trade-off choice. These values give rms time residuals from the simulation of 250 ns, which should also be conservative.

Unfortunately, because of the long time constants involved in the steering, we do not have yet meaningful results of steering the software clock to be shown in the present paper.

Calibration

As shown in Figure 1, up to 15 clocks may be connected to the system. These may include GPS receivers for direct comparisons with the system. The better performance of the new time scale together with the automated measurements are expected to upgrade the precision of the services given by the INPL.

Future prospects

The INPL is considering some additions to the present system. One such addition is a hardware clock with good short term stability which will be locked to the system and provide improved hardware time much the same as a microstepper but with much lower jitter.

References

[1] F. B. Varnum, D. R. Brown, D. W. Allan and T. K. Peppler, "Comparison of time scales generated with the NBS ensemble algorithm," 19th Precise time and time interval symposium, Dec. 1987.

[2] W. J. Klepczynski, H. F. Fligel and D. W. Allan, "GPS time steering," 18th Precise time and time interval symposium, Dec. 1986.

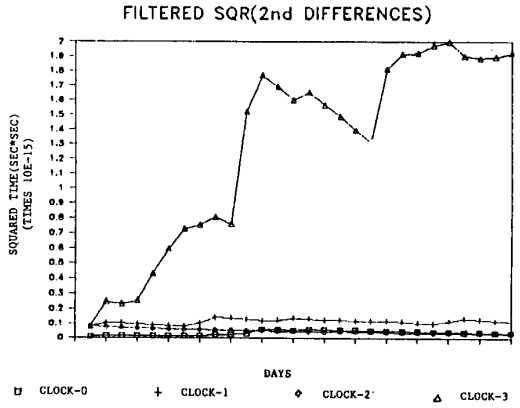

Fig. 4 Filtered Squared Second Differences From Which Clock Weights are Derived.

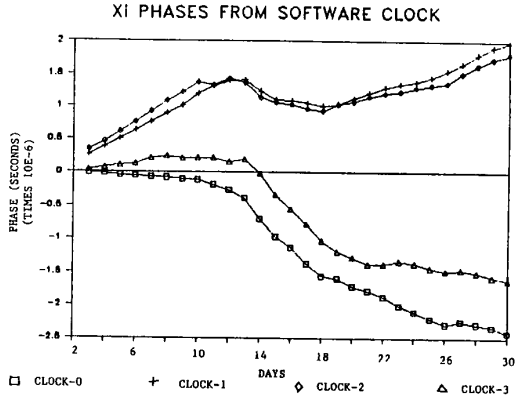

Fig. 7 Phase Evolution of Individual Clocks Versus Time.

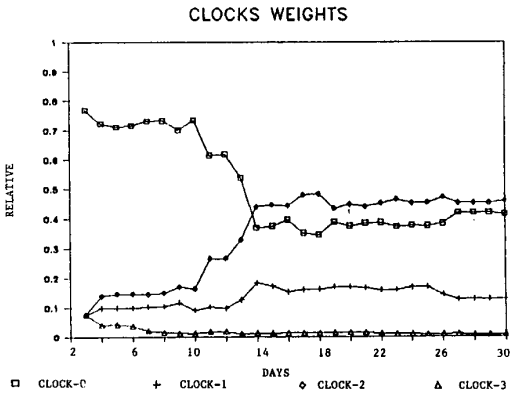

Fig. 5 Individual Clock Weights Versus Time.

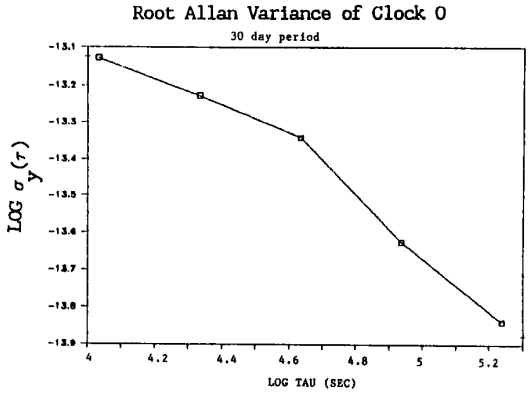

Fig. 8 Square Root of Allan Variance of Clock 0.

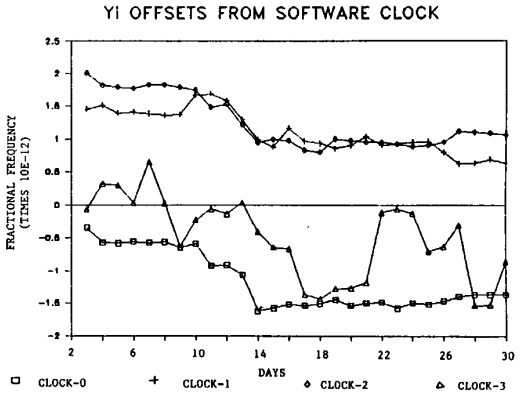

Fig. 6 Fractional Frequency Offsets of Individual Clocks Versus Time.

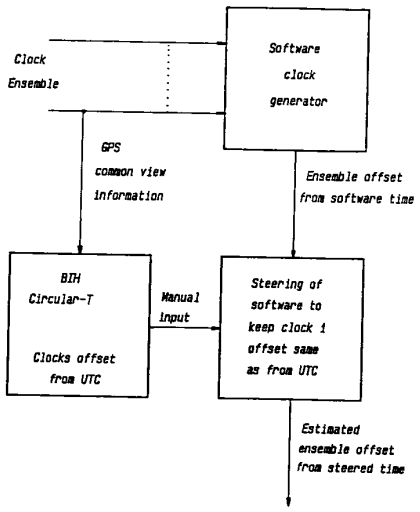

Fig. 9 Block Diagram of the Steering Control Loop.

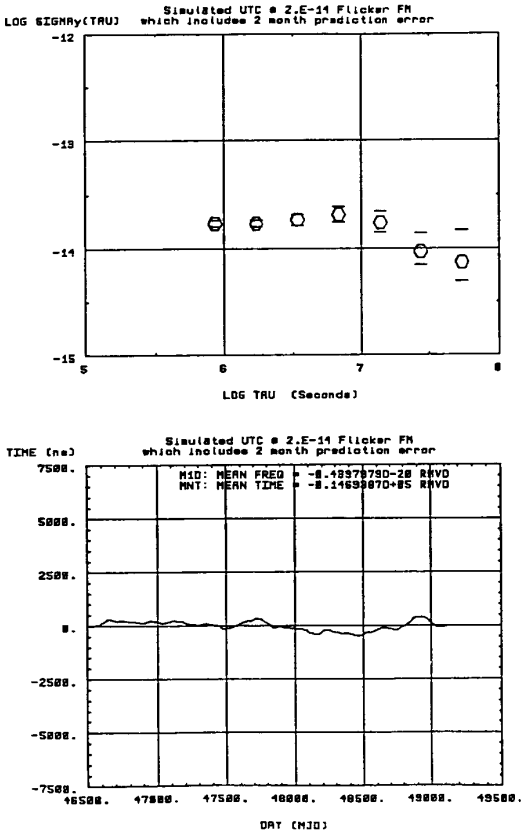

Fig. 10 Simulated UTC Performance.

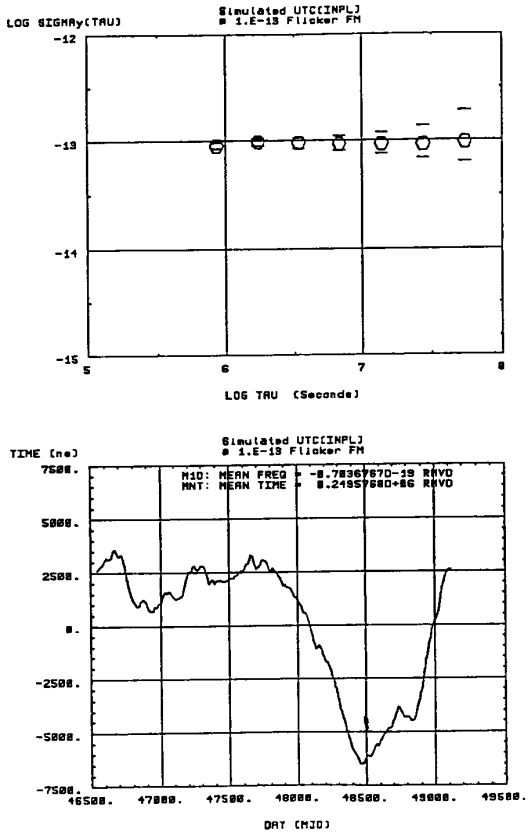

Fig. 11 Simulated UTC(INPL) Clock Ensemble Performance

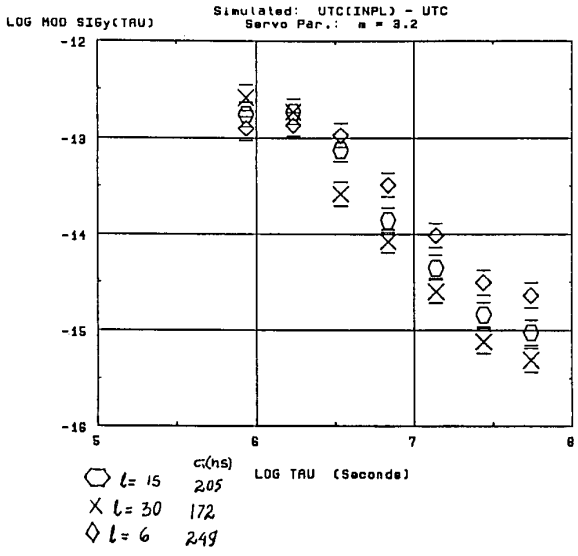

Fig. 12 Simulated Performance of UTC(INPL) - UTC, from which values for m and ℓ were chosen.