

All-Digital GPS Receiver Mechanization

PETER C. OULD and ROBERT J. VanWECHEL

ABSTRACT

THIS PAPER describes all-digital baseband correlation processing of GPS signals. The features of this highly digital mechanization are its (1) potential for improved anti-jamming (AJ) performance, (2) fast acquisition by a digital matched filter, (3) reduction of adjustment, (4) increased system reliability, and (5) provision of a basis for realization of the high degree of very large-scale integration (VLSI) potential for development of small economical GPS sets.

Described are an experimental GPS receiver/digital processing system that has been operating for two years, and an improved engineering development model that is now in test. The basic technical approach consists of a broadband, fix-tuned RF converter followed by a digitizer, digital-matched-filter acquisition section; phase-and delay-lock tracking via baseband digital correlation; software acquisition logic and loop filter implementation; and all-digital implementation of the feedback numerically controlled oscillators (NCOs) and code generator. Baseband in-phase (I) and quadrature (Q) tracking is performed by an arctangent angle detector followed by a phase-unwrapping algorithm that eliminates false locks induced by sampling and data bit transitions, and yields a wide pull-in frequency range approaching one-fourth of the loop iteration frequency.

Test data from the experimental unit includes basic resolution and computation noise of the digital processing, pseudorange and range-rate errors, and acquisition times for the digital matched filter.

I. INTRODUCTION

Continuing advances in digital VLSI technology and recent developments in digital-matched filter mechanization provide the basis for improved GPS receiver designs that yield better AJ performance, considerably faster acquisition, higher precision, increased reliability, and reduced size and weight. These designs maximize digital VLSI technology. Although RF LSI technology has also been developing rapidly, digital VLSI advances have been even more dramatic.

The prototype receiver discussed here requires a relatively simple, fixed-frequency converter, a 10,000-gate digital signal processor and a microprocessor. (Microprocessors with adequate capability are readily available.) The digital signal processor can be contained on 10 or 12 semi-custom, large-gate arrays (at very modest cost for development) using standard LSI technology, or in a single custom circuit package using 1- or 2-micron feature geometry in the technology of the Defense Department's VHSIC program. Our prototype digital processor is constructed on four circuit boards using standard LSI logic. It uses baseband rather than IF correlation for carrier track, code correlation and track, and data demodulation.

Code correlation in this type of receiver is accomplished digitally, using either digital matched filters (DMF) or digital correlators, depending upon performance requirements. By employing recent developments in threshold biasing, these techniques yield improved AJ performance compared with conventional linear IF correlation approaches. Additional benefits of DMF use include much faster acquisition and extended-range delay error measurement.

II. DIGITAL BASEBAND CORRELATION

Background

Although the concepts for digital baseband correlation in spread-spectrum navigation re-

Mr. Ould is a Project Manager and Mr. Van-Wechel is Chief Scientist at Interstate Electronics, Anaheim, CA. This paper was presented at the ION Aerospace Meeting, April 1981.

ceivers have been known for well over 20 years,¹ most currently available receivers employ analog correlation implemented in the receiver's IF amplification stages.² One reason for this almost universal use of linear IF correlation is that digital hardware has been impractical for this application for many years. Another reason is that digital-correlation receivers can be easily jammed if proper precautions are not taken in their design. These obstacles, however, have been removed with the advent of modern digital LSI technology and some fairly recent developments in DMF theory.^{3,4,5} As shown by Turin⁴ and Lim⁵, the use of biased, adaptive thresholds with DMF can improve performance in the presence of non-Gaussian jamming, whereas the linear, analog system may be preferable if only Gaussian noise must be tolerated.

DMF Structures

The DMF concept is extended here to include its variations that permit correlation of long codes and also to allow arbitrarily long integration regardless of DMF length. A two-bit version of such a device is shown in Fig. 1. The IF amplifier output's input signal is quadrature-converted to baseband, sampled, and digitized; then it is correlated with the reference code by means of a shift-register-type of one-bit correlator. Two-bit quantization is accommodated by

two parallel shift register correlators—one for each bit. The correlator outputs are scaled (by a simple shift) and summed. Further integration of the repetitive correlation function can be accomplished by the final accumulator random-access memory, in which the correlation function can be summed over as many samples as desired. The reference code register is replenished with a new code section at uniform intervals depending upon the length of the shift register correlators. This permits use of any code length.

Tracking Loops

The complex correlation coefficients from this device are the loop error data, processed numerically in the microprocessor to support acquisition, carrier track, code track, and data demodulation functions. For acquisition, the entire length of the shift register correlator allows a very rapid search for initial sync. In track, (Fig. 2) the complex on-time correlation element is used to derive an angle error estimate for carrier phase tracking; the correlation coefficients on either side of the peak are used for code tracking in a manner analogous to an early-late delay-lock loop. A sample frequency of approximately two samples per code chip is used so that enough samples are available on the correlation function for accurate code tracking. It can readily be seen

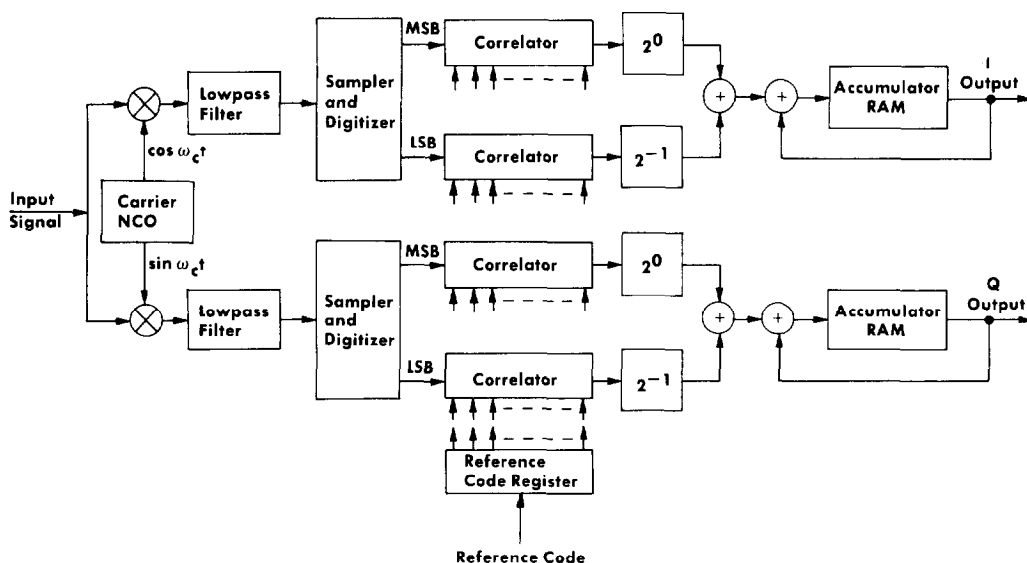


Fig. 1—DMF/correlator.

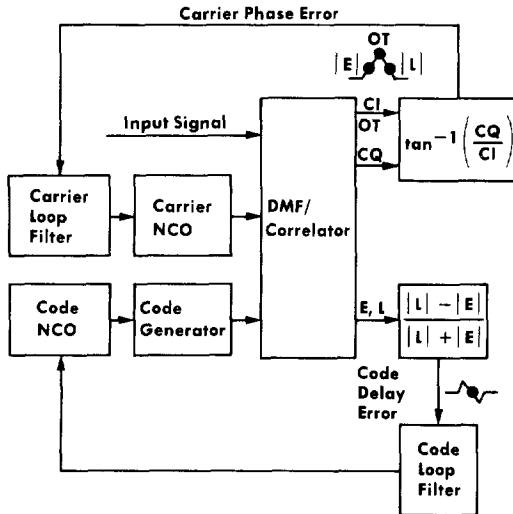


Fig. 2—Carrier/code tracking loops.

that such a DMF makes possible code tracking with greater than one code chip of error. Methods of extended-range-chip tracking can be applied.⁶

The carrier-phase tracking loop (Fig. 2) is a suppressed carrier-phase tracking loop somewhat similar to a Costas loop but with code correlation within the loop. Also, an arctangent replaces the usual Costas angle detector. We have found that the arctangent detector closely approximates the Costas angle detector in signal-to-noise performance. It also offers the advantage of being automatically scaled so that it does not need any external AGC to maintain constant loop gain. This type of carrier-phase estimation has been investigated,⁷ and it has been found that even with one-bit digitization in the correlator, the arctangent is an unbiased estimator of carrier phase for the negative signal-to-noise ratios (SNRs) of interest in GPS applications.

Use of the arctangent function for estimating carrier-phase error permits phase unwrap algorithms⁸ to be used in the loop to increase pull-in range and eliminate any false lock on either sampling or data sidebands. These algorithms, which have been used in digital signal processing to remove the 2π discontinuities in reconstructed phase curves, eliminate all false-lock modes in the digital carrier-tracking loop except the one at half the sample frequency. The algorithm we

used was identical with that of Robinson⁸, except that we modified it to remove phase discontinuities every π radians rather than every 2π radians for compatibility with a biphasemodulated signal. We tested successive samples of the phase, requiring that each successive phase sample be within $\pi/2$ of the previous phase sample. When the phase difference between successive samples exceeded $\pi/2$, increments of π or $-\pi$ were added to the phase to reduce the difference to less than $\pi/2$.

Using this algorithm in acquisition, the narrow pull-in frequency range (normal in a Costas loop) is increased to approach almost one-fourth the sample frequency. In such cases, phase error transients of a complete cycle or more can occur. Above one-fourth the sample frequency, the loop pulls toward the half-sample frequency. Other false-lock conditions, however, are eliminated, and there are a great number of candidates at various fractions of the sample frequency.

Biased Digitizer Performance

As shown by Turin⁴ and Lim⁵, using biased digitizing thresholds in the DMF of Fig. 1 can significantly improve AJ characteristics. For a properly biased digitizer, the output SNR for incoherent, constant amplitude jamming on a two-bit digitized DMF can be expressed as:

$$\text{SNR}_o \approx \frac{8L}{\pi^3} \sqrt{\text{SNR}_i}$$

Figure 3 compares the potential resulting AJ

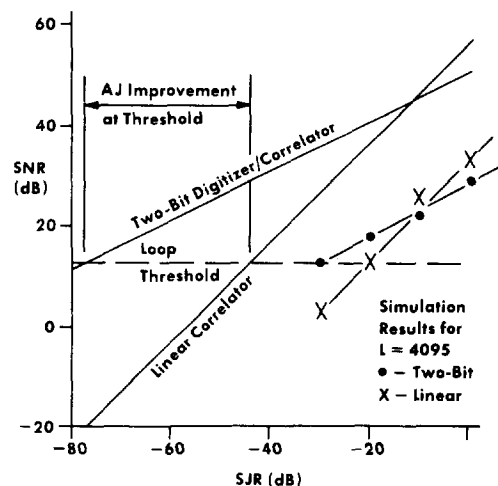


Fig. 3—Correlator snr vs sjr with c. w. jamming.

improvement offered by this square-root relationship with linear correlation. This improvement can be seen to be very significant under heavy jamming, which should not be surprising because the linear matched filter is known to maximize SNR₀ only for the Gaussian-noise case. A nonlinear DMF with biased thresholds can do better on other types of interference. Even in white Gaussian noise, however, the DMF implemented with two-bit quantization and biased thresholds only shows a loss of 1 dB or less as compared with the linear filter.

Acquisition Time

A DMF for C/A-code acquisition provides considerably faster acquisition than possible

Table 1—C/A-Code Acquisition time
(2500-Hz Offset)

Parallel Search Chips (no.)	Acquisition Time (sec)
6	95.0
16	35.5
32	17.7
64	8.9
128	4.5
256	2.2
512	1.1
1023	0.55

with conventional IF correlators, which are normally used in a single-chip-at-a-time search mode. Table 1 lists our acquisition-time projections for matched filters of various sizes and with an initial Doppler offset of 2500 Hz. It is assumed that the Doppler is searched in 200-Hz steps, which is the step size we used in obtaining test data with our experimental system.

III. FUNCTIONAL DESCRIPTION

Receiver

The receiver (Fig. 4) consists of an RF converter, digital signal processor, and microprocessor. The microprocessor accomplishes overall receiver control.

This prototype GPS receiver can be used as a multiplex, single-channel navigation receiver or as a single-channel timing receiver. The digital processor's correlator, code generator, and two NCOs are treated as the microprocessor's peripherals, accessible on the microprocessor bus.

The RF converter's frequency synthesizer synthesizes all local oscillator (LO) signals for the converter, as well as the processor's timing signals. The converter's frequency plan was subjected to thorough analysis to eliminate possible inband spurious responses and simplify the syn-

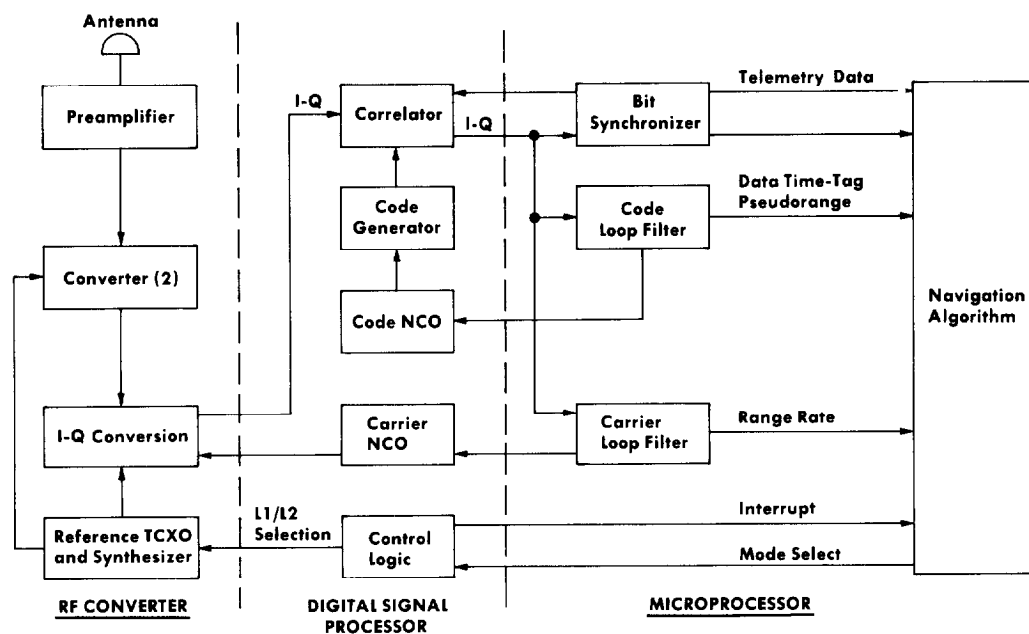


Fig. 4—GPS receiver.

thesis as much as possible. In this approach, all of the converter LOs are fixed-frequency sinusoids except for the final down-conversion to baseband, which uses a signal from the carrier NCO with the carrier Doppler. The output of the final baseband down-conversion contains I and Q signals that are lowpass-filtered to recover the coded signal plus noise at baseband. The I and Q signals are then applied to a highpass filter before being sampled and digitized. (These simple, single-pole passive filters are an integral part of the final down-converter.) Since the high-pass filter need only pass the longest expected string of 1's or 0's in the codes, an effective notch filter is placed at the center frequency of the spread-spectrum signal. This prevents any external or self-jamming components at the carrier frequency from entering the digitizer. The baseband signals are then sampled and digitized by a self-adjusting, biased-threshold-type of two-bit digitizer before being correlated with the reference code. The sampling frequency used for the signal and the reference is approximately two samples per code chip but is not synchronous with the chip frequency.

Digital Signal Processor

The digital signal processor consists of a two-bit DMF correlator, two NCOs (code and carrier), code generator, and miscellaneous control logic functions.

DMF/Correlator

The baseband DMF/correlator (Fig. 1) generates a normalized delay error function, as shown in Fig. 2.

The early and late components arise from sampling the signal and reference codes at approximately twice the chip rate. The combined effect of the code NCO and asynchronous sampling process results in a system with the delay resolution characteristics shown in Fig. 5 (continuous curve).

Our experimental system included an acquisition-only type of matched filter using the TRW 1010J integrated circuit (IC). This device provided a 64-element filter in a single chip, but the output level drift associated with the current summing circuits made it impractical for anything but experimental use. We used a tracking device that was five samples in length, con-

structed with standard STTL logic to support P-code tracking at a 20-MHz sample rate.

The two most recent correlator designs provide two-bit processing and are now being evaluated in our developmental system: one is a 64-sample device based on the TRW TDC-1023J digital correlator chip that is capable of acquisition and C/A-code tracking; and the other is a three-sample, two-bit correlator (Fig. 6) used on both C/A and P codes and which is the prototype for a gate-array development. In this design, the quantized samples are scaled to produce all positive values so that simple up-counters can be used to integrate the most-significant bit (MSB) and least-significant bit (LSB) independently. The integrated MSB and LSB values are presently combined in the tracking microprocessor to produce the tracking error function.

NCO

The GPS receiver mechanization shown in Fig. 4 employs a code clock generator and a

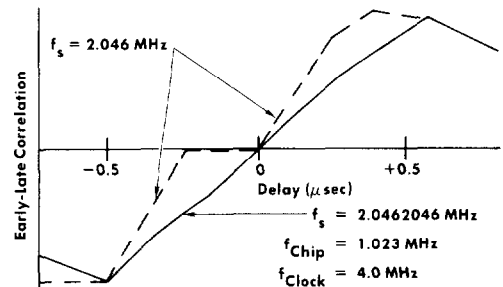


Fig. 5—Early-late correlation (from simulation program).

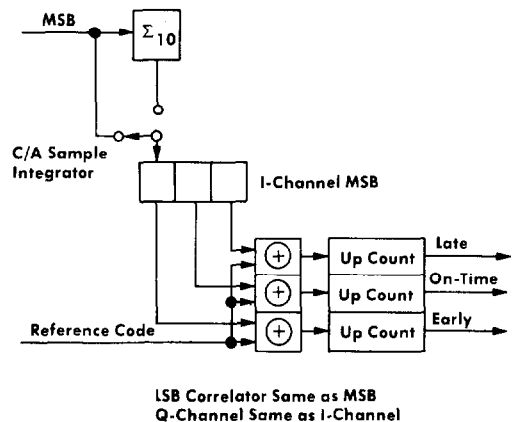


Fig. 6—Three-sample, two-bit correlator.

carrier frequency generator (LO) in the form of an NCO. The code NCO, which is typical of both NCO used in the system, consists of a 32-bit accumulator which is caused to overflow periodically at the desired output frequency.^{9,10} By defining the accumulator's output number range as 2π , and using a sine-table ROM and digital-to-analog (D/A) converter (Fig. 7), a sine wave can be produced at the overflow frequency; however, we have found that a satisfactory alternative to the sine-table ROM and D/A converter is simply to use the accumulator's MSB as the NCO output wave, which eliminates ROM and D/A converter complexity. This approach appears to be adequate if the NCO clock frequency is selected carefully to be asynchronous (or at least synchronous to a high order) with NCO output frequency over the Doppler range of interest. Such a condition is easily achieved for the GPS-satellite Doppler range and for users with subsonic velocities.

We selected this device over available alternatives for the following reasons:

- (1) *Ease of mechanization in digital integrated circuits.* The required clock rates are compatible with CMOS logic and require a minimum number of linear circuits, whereas rate-multiplier designs require very high clock rates for fine delay quantization, and multistage linear circuits are required to up-translate the delayed clock.
- (2) *Direct setting of NCO phase (code delay in fractional chips) by the loop filter in*

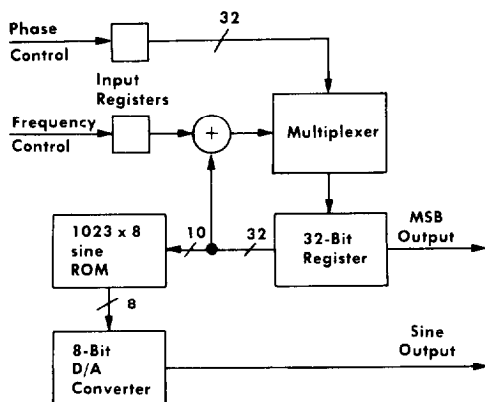


Fig. 7—NCO.

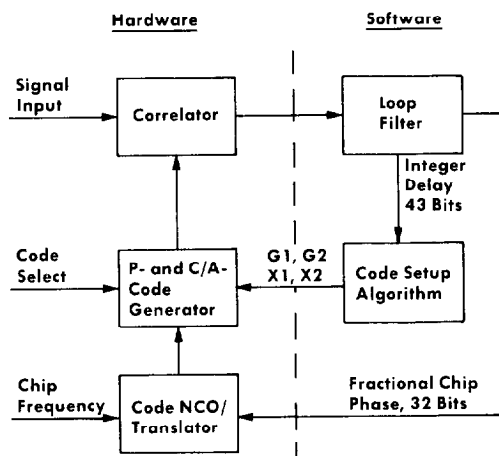


Fig. 8—NCO/code generator control in delay-lock loop.

addition to the normal code frequency input.

The ability to directly set initial phase is valuable when it is necessary to multiplex between satellites, in order to start the oscillator at the required phase at switchover. In the code NCO, the ability to directly set initial phase facilitates direct phase control, rather than frequency control, of the combined code NCO/code generator, as shown in Fig. 8. In directly controlling phase (or delay) of the code NCO/code generator, the command word from the loop filter is separated into its integer and fractional part in terms of code chip units. The fractional part sets the code NCO phase and the integer part sets the code generator's state (or starting chip). Together, the code NCO and code generator can then be started at very fine fractional chip increments on any chip desired.

This method of code NCO/code generator phase control (as opposed to frequency control) effectively moves the sample-to-sample loop integration back into the software, which uses a conventional sampled-data, digital-loop-filter integration. Integration between loop filter iterations is still performed in the NCO. The advantage is that the commanded delay or phase value for each iteration is available in the microprocessor in its precise form, much more accurately than it could be obtained by a measurement in the digital hardware. It eliminates the need for a count and measurement in the digital hard-

ware. In satellite signal simulation, it is also very useful because it permits direct, precise delay control and prevents a buildup of the numerical integration error that could occur if frequency control were used over a long period. This is very important in simulating signals with precise orbital-data characteristics.

The average NCO output frequency F_o can be expressed in terms of the frequency control word W and clock frequency F_c as:

$$F_o = \frac{W}{2^N} \cdot F_c$$

The accumulator has been implemented with a word length $N = 32$.

The frequency quantization step size can be seen to be:

$$q_f = \frac{F_c}{2^N}$$

For a 8.42-MHz clock and 32-bit accumulator, this yields a frequency quantization step size of about 0.002 Hz.

In phase setting, the quantization step size is 2^{-N} of a cycle of the output wave, on the average, when correlated and integrated over the correlation time, and provided the asynchronous relationship with the clock is maintained. Note that the fine resolution is not achieved on a cycle-by-cycle basis because the clock frequency is too low; however, when taken as a long-term average over the integration time, the high resolution is achieved in the correlation function, which is used for tracking. For a 32-bit accumulator, then the resolution in phase when viewed at the correlation function is so high (2^{-N} of a code chip) that it may be regarded as continuous for all practical purposes.

NCO performance can be demonstrated most effectively at the system level because of the close interrelationship with the DMF/correlator, code generator, and associated numerical processing. Our simulation (Fig. 5) consisted of an accumulator-type NCO from which only the sign or MSB was used as the output. This output then was used to clock a code generator correlated against a second fixed code. Both codes were sampled at the same frequency. The NCO clock was simulated at 4 MHz, and the NCO output average chip frequency was 1.023 MHz. Two sample frequencies, 2.046 MHz and 2.0462046 MHz, were tested. Since the former is

synchronous with the chip frequency, an undesirable delay quantization occurs as the NCO delay control is varied; however, use of the second sample frequency, which is asynchronous with the chip frequency, results in a smooth, high-resolution, delay-lock-loop error characteristic.

Two NCOs (carrier and code) were mechanized with 32-bit accumulators in an experimental C/A-code receiver built to test these concepts. Only the MSB of the output was used, without any sine-table ROM or D/A converter. The one-bit correlator operated with a sample frequency of 2.0462046 MHz, which is 1 part in 10^4 removed from synchronism with the C/A code in order to maintain asynchronous operation over all possible Doppler shifts. Fig. 9 shows the carrier-loop phase jitter and random Doppler error measurements from this system plotted against a varied loop bandwidth. A rather high SNR of -10 dB in the C/A-code bandwidth was used to demonstrate basic resolution of the system. (A negative SNR is essential in a one-bit correlation system for proper operation.) It can be seen that the loop phase jitter flattened out at about 2 degrees rms, which was the level caused by the system's internal numerical truncation noise. No RF sources with large phase jitter, as caused by flicker noise, contributed in this region. The Doppler random error reduced with loop bandwidth to a level well below 0.001 Hz rms.

Fig. 10 is a computer-derived plot of the carrier NCO input word from this same experimental receiver when tracking the NAVSTAR 4 C/A code. The slowly changing Doppler shows as

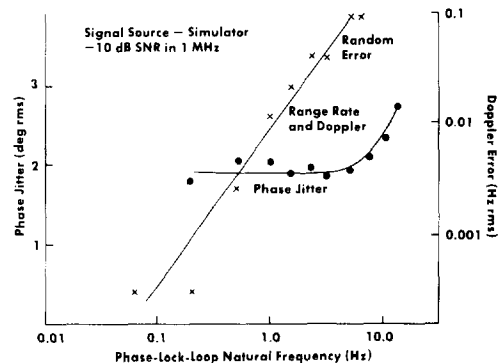


Fig. 9—Measured phase jitter and random doppler error vs loop bandwidth.

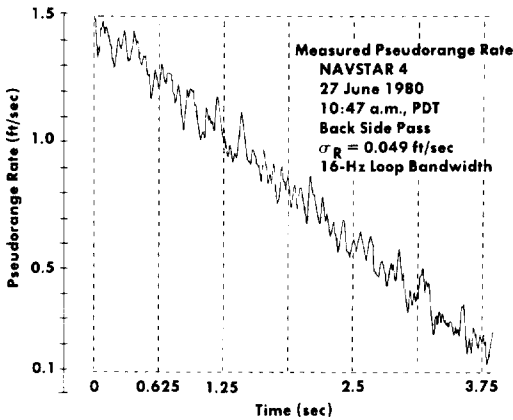


Fig. 10—Typical pseudorange-rate measurement test results.

the overall trend, and is superimposed with the random error. This error, determined by deviations from the least-squares-fit line, was 0.049 ft/sec (1σ) or about 0.08 Hz, approximately 40 times the basic NCO resolution. The data in this case was not smoothed at all and has the full noise bandwidth of the 16-Hz loop (single-sided equivalent noise bandwidth).

Fig. 11 shows data from the code NCO/code generator phase (or delay) control input for a C/A-code track NAVSTAR 4. Here again, the data was not smoothed. The random error from the best-fit line was 6.1 ft (1σ). The NCO delay resolution in this case was 32 bits on each code chip, so it is much finer than can be seen in the figure.

Code Generator

The code generator (Fig. 12) produces both the C/A and P codes. Code delay in P-code chips to any desired time of week in less than 2 msec. The time-of-week (TOW) word is supplied to the device in units of P-code chips (a 43-bit number). A dedicated-code-setup Motorola MC68000 microprocessor converts the TOW word to the initialization index words that start the various code components at their required values in the digital hardware sequence generators.

For C/A-code generation, two 1023-bit sequence generators (G1 and G2) and some indexing control generate all C/A codes¹¹ required. Two 4092-bit sequence generators (X1A and

X2A) and two 4093-bit sequence generators (X1B and X2B) are used, along with the indexing control, for P-code¹¹ generation. The sequence generators are mechanized for direct control of the starting code state.

The first step in computing the code-setup initialization indexes is to compute the Z count¹¹ for the starting chip. With the input TOW expressed as t_w in units of P-code chips, the Z count is expressed as:

$$Z_1 = \text{Int}\left(\frac{t_w}{15,345,000}\right)$$

Note that t_w has a range, for the entire week, of $0 \leq t_w < 6,187,104,000,000$. The Z_1 range is $0 \leq Z_1 < 403,200$.

The number of P-code chips since the last Z-count epoch is r_1 , which is computed as:

$$r_1 = t_w \text{ Modulo } 15,345,000 \quad (0 \leq r_1 < 15,345,000).$$

The parameter j_{1a} —the number of X1A sequences occurring since the last Z-count epoch—is then computed as:

$$j_{1a} = \text{Int}\left(\frac{r_1}{4092}\right) \quad (0 \leq j_{1a} < 3750)$$

The starting index i_{1a} for the X1A sequence generator is computed as:

$$i_{1a} = r_1 \text{ Modulo } 4092 \quad (0 \leq i_{1a} < 4092)$$

The starting index i_{1b} for the X1B sequence generator is likewise computed as:

$$i_{1b} = r_1 \text{ Modulo } 4093 \quad (0 \leq i_{1b} < 4093)$$

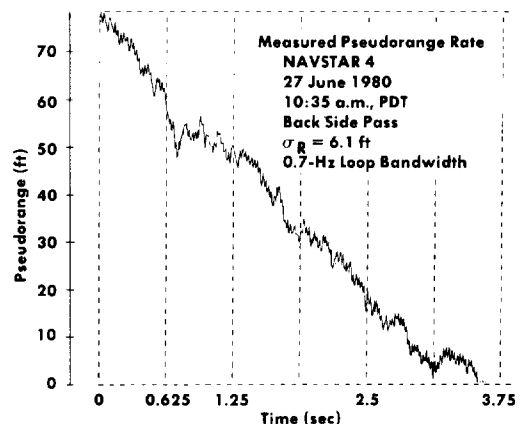


Fig. 11—Typical pseudorange-measurement test results.

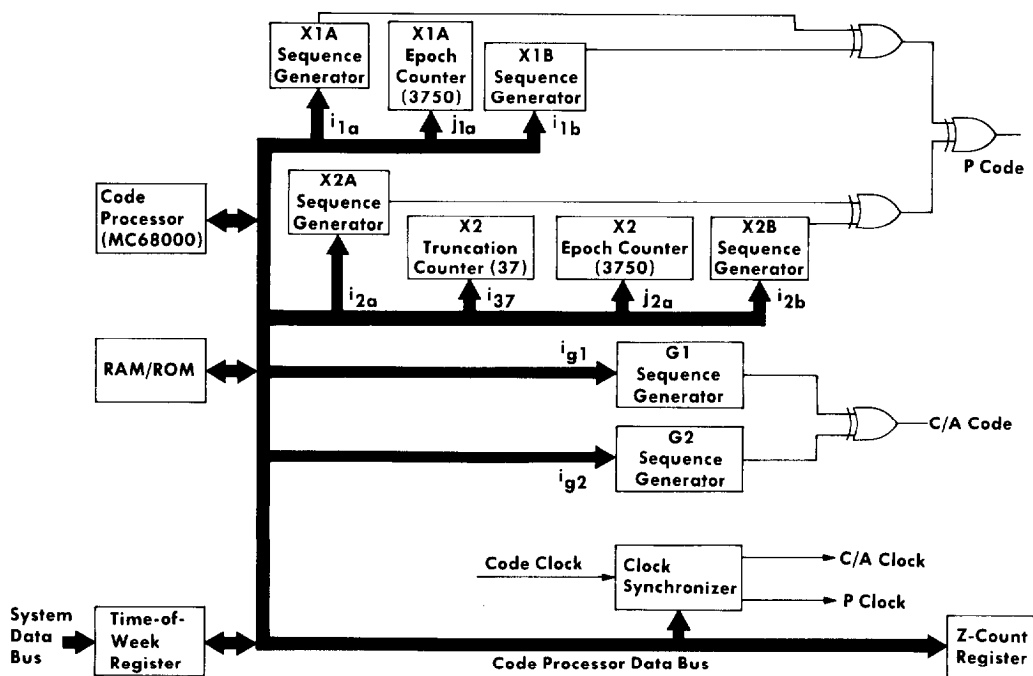


Fig. 12—Code generator.

A parameter Z_2 is then computed for the X2 sequence. This parameter is a counterpart to the Z count:

$$Z_2 = \text{Int}\left(\frac{t_w}{15,345,037}\right) \quad (0 \leq Z_2 < 403,200)$$

In a manner analogous to the X1 sequence, the remaining X2 sequence parameters are computed as:

$$r_2 = t_w \text{ Modulo } 15,345,037$$

$$j_{2a} = \text{Int}\left(\frac{r_2}{4092}\right)$$

$$i_{2a} = r_2 \text{ Modulo } 4092$$

$$i_{2b} = r_2 \text{ Modulo } 4093$$

An additional parameter i_{37} is computed for the X2 sequences as:

$$i_{37} = \begin{cases} 0 & \text{if } r_2 \leq 15,345,000 \\ r_2 \text{ Modulo } 15,345,000 & \text{if } r_2 > 15,345,000 \end{cases}$$

For C/A-code initialization, only two parameters are required: i_{g1} and i_{g2} .

For the G1-sequence starting index, i_{g1} is com-

puted as:

$$i_{g1} = \frac{1}{10} (r_1 \text{ Modulo } 10230)$$

The G2 sequence starting index is computed from i_{g1} , and from the G2-sequence delay k (for satellite selection) as:

$$i_{g2} = (1023 + i_{g1} - k) \text{ Modulo } 1023$$

Both i_{g1} and i_{g2} range from 0 to 1022.

These index values are output from the code processor to the code generator, where they are used as setup parameters, as shown in Fig. 12.

The precomputed parameters specifying a given code and code delay are entered via the code processor bus (Fig. 12) at a 0.004-sec interval, which is a function of system sample rate. At the beginning of each integration period, the delay data is strobed to precisely set the codes. When the integration period ends, the fractional chip error function derived from the correlator updates the entire delay control word (integer and fractional parts), which in turn is used to correct both the code-NCO and code-generator setting for the next integration cycle.

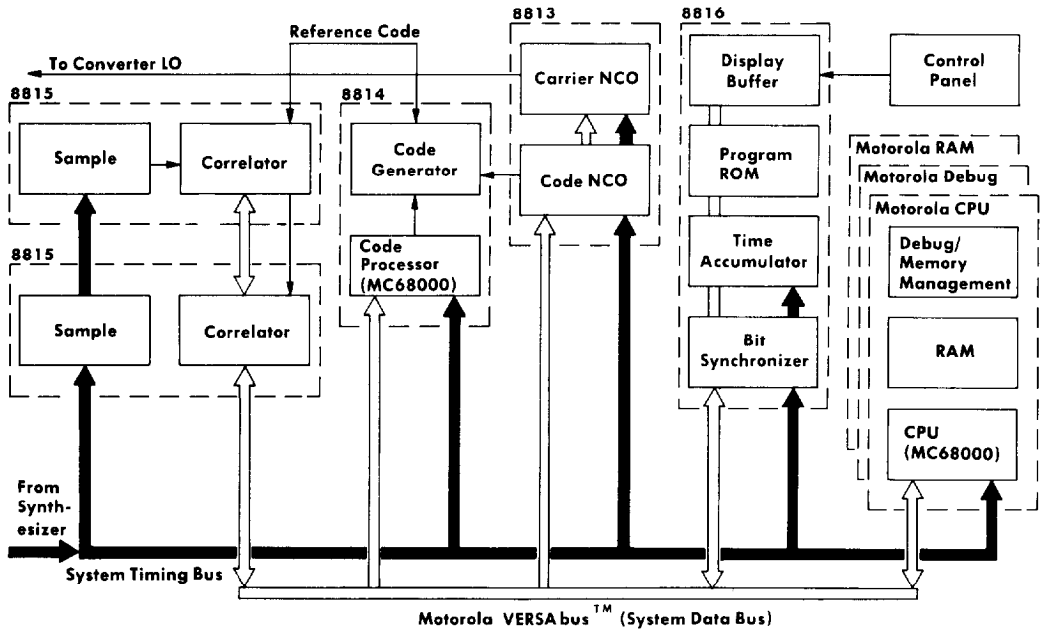


Fig. 13—GPS set functional/physical layout.

IV. HARDWARE-SOFTWARE IMPLEMENTATION

Processor/Bus/Interface

The integrated functional layout and processor/bus interface are shown in Fig. 13. The DMF/correlator, code generator, two NCO, and microprocessor are interconnected with common data and timing buses. The time accumulator, derived from system frequency standards, generates a 43-bit time reference word used to tag all system data related to timekeeping. When the system is up and locked on a GPS satellite, this time reference is synchronized to GPS time, which then becomes the source for time clock references, navigational computations, and the synchronization of multiple-satellite reception in sequential, coherent fashion.

Software

The processing program, Executive, and all functional software are coded in Motorola Assembly Language. The execution time for each loop iteration in both acquisition and track, including data demodulation, is less than 0.004 sec.

Physical Characteristics

The dashed lines in Fig. 13 indicate the digital circuit boards for the system. These eight boards plug directly into a Motorola EXORmacs* (MC68000 development system) chassis. The Motorola VERSAbus* provides a standard input/output environment for all boards and an uncommitted bus section that has been custom tailored to distribute timing signals peculiar to the GPS processor implementation.

The special-purpose boards were patterned after the standard 9-inch-by-14-inch boards (delivered with the Motorola EXORmacs*), which are ideally suited to the functional partitioning requirements of the GPS processor circuits. These boards are commercial-grade, printed-circuit types with two external signal layers and separate internal power and ground planes. The approximate IC count, regardless of the number of pins on a package, is as follows for each custom board:

* Registered trademarks of Motorola, Inc.

Board No.	Unit	No. of ICs
8813	NCO	68
8814	Code Generator	149
8815	Correlator	76
8816	Controller	80

In their present form, these boards represent the basic prototype for a VLSI developmental effort that will significantly reduce cost, size, and power consumption in the interim. The unnecessary parts of the Motorola logic and experimental aspects of the controller board will be removed in the next generation of our design, reducing the total board count from eight to six.

ACKNOWLEDGMENTS

The authors acknowledge the support of the following for their assistance throughout this development: Dr. A. J. Mallinckrodt, who contributed many ideas and overall guidance—especially in the area of sampling requirements for the digital signal processing; Dr. Robert Gold, for assistance in NCO analysis and analysis of the arctangent angle estimator; Hal Morris and Dennis Beech of Interstate Electronics Corporation for managerial support; Jim Soapes of Interstate Electronics for innumerable hours in support of the hardware development; Al Rovner and Vode Norwood, both of Interstate Electronics, for digital logic and firmware development; Jack Dressler for digital logic design; and Phil Pohl and Larry Wells, both of Interstate Electronics, for support in RF receiver design.

REFERENCES

1. Harrington, J. V., "An Analysis of the Detection of Repeated Signals in Noise by Binary Integration," *IEEE Trans. on Information Theory*, Vol. IT-1, March 1955, pp. 1-9.
2. Dixon, R. C., *Spread Spectrum Systems*, John Wiley & Sons, New York, 1976.
3. Gooding, D. J., "Increasing the Utility of the Digital Matched Filter," in *Proc. NELC Spread-Spectrum Symposium*, 1973.
4. Turin, G. L., "An Introduction to Digital Matched Filters," *Proc. IEEE*, Vol. 64, July 1976, pp. 1092-1112.
5. Lim, T. L., "Non-Coherent Digital Matched Filters: Multibit Quantization," *IEEE Trans. on Comm.*, Vol. COM-26, April 1978, pp. 409-419.
6. Upadhyay, T., Roth, R. S., Bowles, W. M., and Hlavty, C. W., "Recent Results in Global Positioning System (GPS) Receiver Code Tracking Technology," paper presented at NAECON, 1980.
7. Gold, R., "Carrier Phase Estimates Using I and Q Channel Digital Correlators," Interstate Electronics Corp. IOM 3300-281, 14 February 1978.
8. Robinson, E. A., *Multichannel Time Series Analysis with Digital Computer Programs*, Holden-Day Publishing Company, 1967, p. 65.
9. Tierney, J., Rader, C. M., and Gold, B., "A Digital Frequency Synthesizer," *IEEE Trans. on Audio and Electroacoustics*, Vol. AU-19, March 1971, pp. 48-57.
10. Cooper, H. W., "Why Complicate Frequency Synthesis?," *Electronic Design*, Vol. 22, No. 15, 19 July 1974, pp. 80-84.
11. Space Vehicle Navigation Subsystem and NTS PRN Navigation Assembly/User System Segment and Monitor Station, Rockwell Interface Control Document MH08-00002-400, Revision G, 6 August 1979.