Naval Research Laboratory

Washington, DC 20375-5320

NRL/MR/5523--93-7184

A Channel Simulation Processor for the Multi-Node Tactical Network Simulator

JOSEPH P. MACKER

Communication Systems Branch
Information Technology Division

January 17, 1993

Approved for public release; distribution unlimited.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY (*Leave Blank*)	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	January 17, 1993	

4. TITLE AND SUBTITLE

A Channel Simulation Processor for the Multi-Node Tactical Network Simulator

5. FUNDING NUMBERS

PE-0603717
PR-N0003992WXER077

6. AUTHOR(S)

Joseph P. Macker

7. PERFORMING ORGANIZATION NAME(S) and ADDRESS(ES)

Naval Research Laboratory
Washington, DC 20375-5320

8. PERFORMING ORGANIZATION REPORT NUMBER

NRL/MR/5523--93-7184

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

Space and Naval Warfare Systems Command
Washington, DC 20363-5100

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION/AVAILABILITY STATEMENT

Approved for public release; distribution is unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT (*Maximum 200 words*)

The design and development of a digital signal processor (DSP) based groundwave and ionospheric communication channel simulator is described. The primary objective is to simulate communication channel perturbations for a network of communication links via an array of DSP modules operating in real-time. This array of channel simulator processors is a subsystem of the multi-mode tactical network simulator (TACNETSIM) being developed by the Information Technology Division at the Naval Research Laboratory(NRL).

The propagation model developed is based upon intra-battlegroup high frequency (HF) communications. The communication channel model consists of two modes of propagation: HF groundwave and HF skywave. The HF skywave component is simulated via a Watterson ionospheric model. The real-time DSP implementation of the Watterson ionospheric model is described in terms of subprocesses. The performance of each subprocess is shown to be within established statistical parameters.

Detailed specifications are given of both the hardware and software design. The simulator is designed for dynamic channel modeling and includes a data interface for real-time program control.

14. SUBJECT TERMS

Watterson Model | Digital Signal Processing | Propagation Modelling

15. NUMBER OF PAGES

28

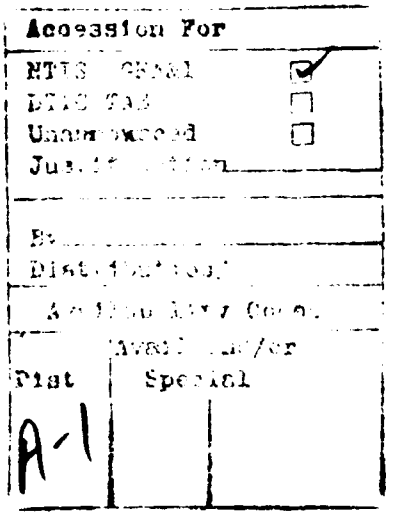
16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	UL

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std 239-18
298-102

CONTENTS



A CHANNEL SIMULATION PROCESSOR FOR THE MULTI-NODE TACTICAL NETWORK SIMULATOR

INTRODUCTION

This report describes the design and development of a digital signal processor (DSP) based groundwave and ionospheric communication channel simulator. The primary objective is to simulate communication channel perturbations for a network of communication links via an array of DSP modules operating in real-time. This array of channel simulator processors is a subsystem of the multi-node tactical network simulator (TACNETSIM) being developed by the Information Technology Division at the Naval Research Laboratory (NRL). Other TACNETSIM subsystems include; the radio frequency (RF) processing subsystem, the analog baseband subsystem, and the control subsystem. The specific operation and design of these other subsystems shall be covered in future documents. Our focus here is strictly on the design and implementation of the baseband channel simulator processing subsystem.

BACKGROUND

Since the 1970s, real-time, baseband ionospheric channel simulators have been available to perform measurement of stressed communication link performance under controlled conditions. Past simulators have tended to be large and expensive with a small number of independent channel paths available to the user during testing. They have served as useful tools in measuring the point-to-point link performance of digital communication equipment under predicted stressed channel conditions.

As tactical digital radio communication evolves away from point-to-point, single-link operation and towards multiple-node networks, communication performance becomes more difficult to predict. For instance, in a modern tactical communication scenario the connectivity or network topology changes largely because of dynamic RF channel conditions, the relative positions of network members, and the existence of intentional and/or unintentional interference sources. By employing an independent link simulation between each transmitting node and each receiving node, realistic on-line network channel simulation, including ionospheric propagation, becomes possible. The goal of TACNETSIM is to perform such real-time simulations using actual tactical communication components as part of a laboratory-based simulation testbed.

Manuscript approved November 5, 1992.

The number of channel simulators required for TACNETSIM operation is determined by the number of network nodes involved. To establish full, bi-directional network connectivity the number of independent channel simulators required is n*(n-1), where n is the number network nodes. Here we discuss the design of an inexpensive, real-time controlled channel simulator to be used as the core unit of the channel simulation processor subsystem.

Design Approach

For the initial design phase, our goal is to accurately simulate the communication links for an extended line-of-sight (ELOS) high frequency (HF) groundwave network. The communication link model consists of a groundwave propagation component and an HF skywave ray component, which may or may not be present. An overview of this model is shown in Figure 1. The ionospheric effect on the skywave ray component is simulated using the Watterson ionospheric channel model [1].

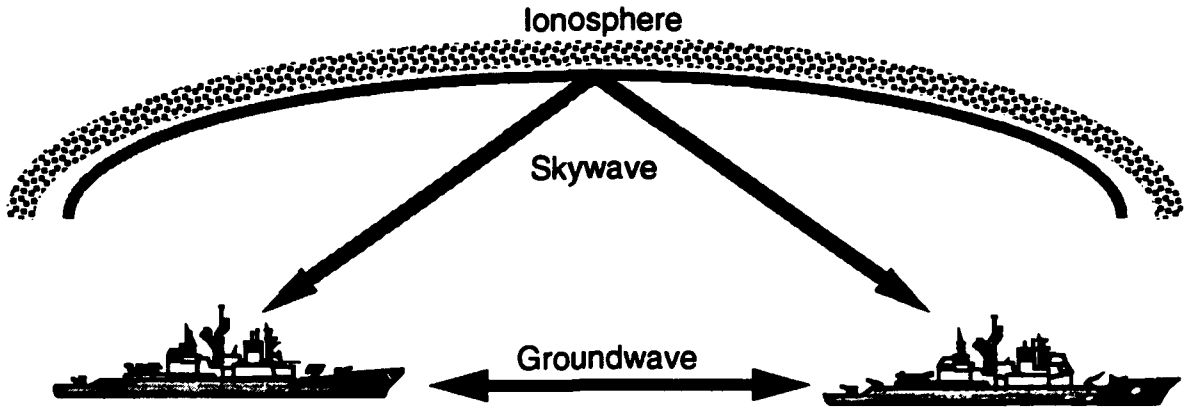


Figure 1: HF ELOS Propagation Components

The requirements for the channel simulation processor are low cost, high performance, and external control. First, low cost per processor must be maintained because an array of processors is required for network simulation, with the array size growing as a second-order polynomial of the number of nodes. Second, a processor with high performance characteristics is needed to implement the computationally intensive HF channel simulation in real-time. Third, in order to perform simulations of dynamic network

topologies and channel characteristics a control interface must allow the simulation parameters to be periodically updated.

Hardware Components

The hardware host for the channel simulator is based upon the existing NRL-developed Variable Speed Modem (VSM) processor board [2]. The processor board contains a TMS320C25 DSP chip, no-wait state external RAM and ROM, an analog CODEC interface chip, and an 8-bit parallel input latch for external simulator control. Present availability of thirty 2.5"x4" processor boards makes an inexpensive, compact six-node simulator achievable. These boards are being used as an inexpensive approach for the TACNETSIM initial system development phase. Figure 2, shows an overview of the VSM processor board and its external control interface.

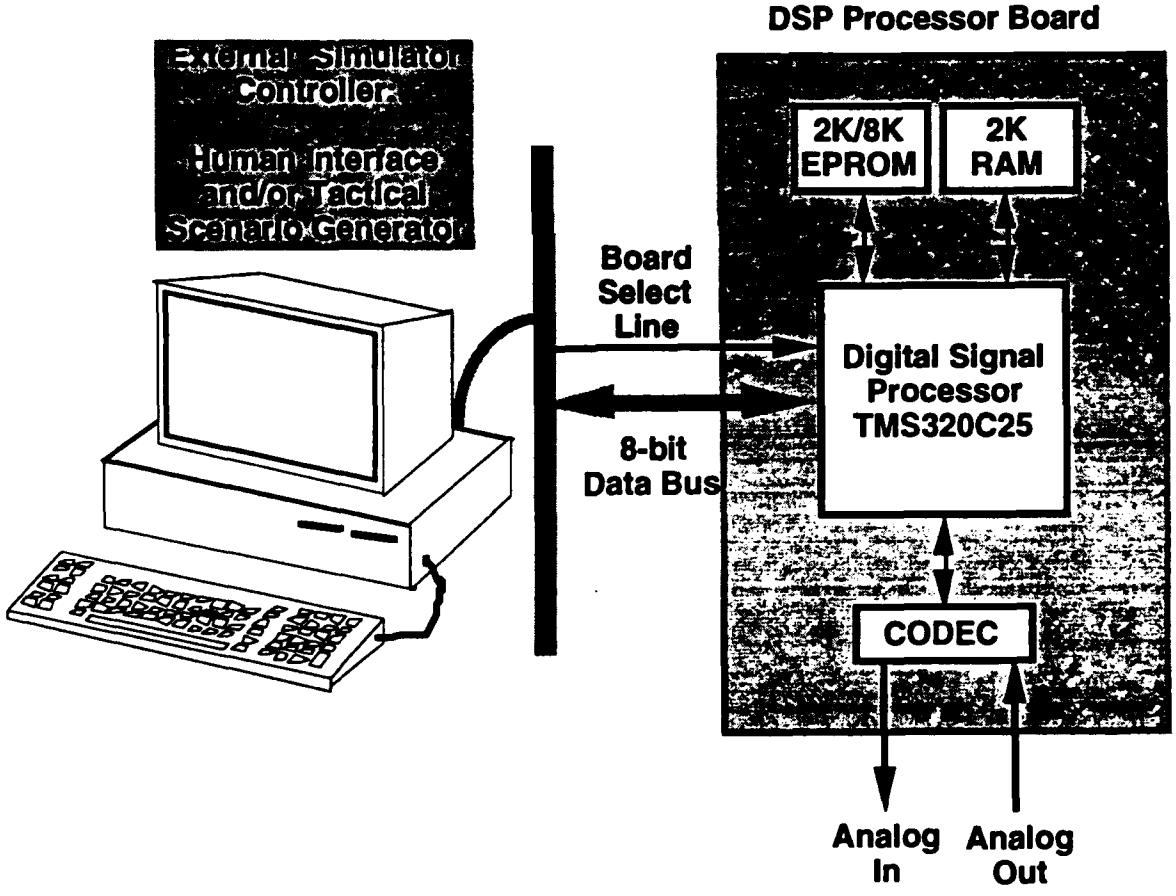


Figure 2: Simulator Processor Board with External Control

We envision the future TACNETSIM upgrades to consist of high performance DSP processor cards, each capable of performing multiple channel simulations. Multiple channel (e.g. 12-bit, 32 channel) analog interface processor boards would also be used, allowing a large number of communication links to be simulated with a small number of processor boards.

Intrabattlegroup (IBG) HF Propagation Model

The initial objective of the TACNETSIM system is to provide a real-time channel simulation for an HF groundwave IBG network. As shown previously in figure 1, the propagation model consists of two potential propagation paths between each platform; a groundwave component and an ionosphere-reflected skywave component. The relative signal strengths of these components at a receiving node depends upon several factors including transmitter distance, ionospheric condition, sea condition, and antennae characteristics.

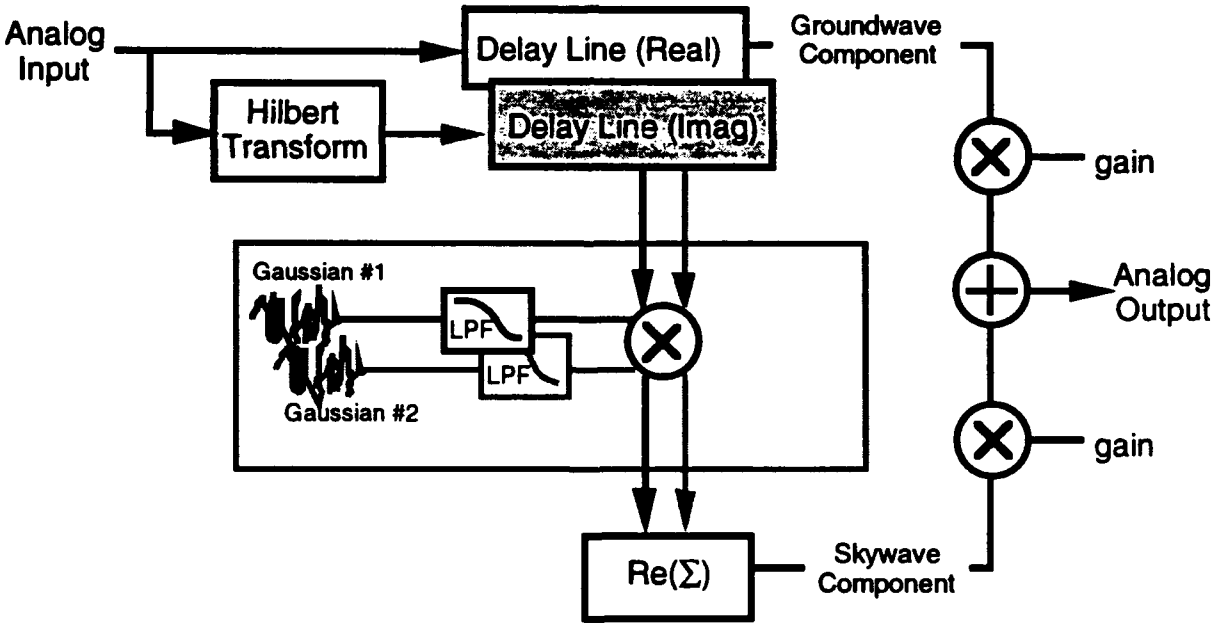


Figure 3: Processing Path for the ELOS HF Simulator

The Watterson model of HF ionospheric propagation is used to simulate all received skywave components in the IBG network. This is achieved by passing an incoming analog signal through the processing chain shown in Figure 3. Several processor-intensive

software routines are required to implement the Watterson simulation including a Hilbert transform, an independent Gaussian sample generator, and a doppler-spread bandwidth filter. These three major processing elements were analyzed prior to development to determine their real-time processing constraints. The real-time processing requirements were found to be well within the processing power of the VSM hardware [3].

HARDWARE DESIGN

As mentioned previously, the channel simulator hardware is based upon the existing NRL VSM processor board. A schematic of the processor board and its recent modifications is given in Appendix A. Here we discuss the hardware design issues relating to the processor, analog interface, and the control interface.

Processor and Memory

The VSM processor board contains the Texas Instruments (TI) TMS320C25 DSP processor. This processor is capable of operating at clock rates up to 40 MHz. At present the TMS320C25 is running at 16.128 MHz, which is a common multiple of two processor-derived system clocks required by the analog interface. The processor is based around a 16-bit data bus and provides double precision (32-bit) operations with the accumulator and multiplier product registers.

The memory used on the VSM processor board includes programmable read-only memories (PROMs) for nonvolatile program and data storage, and read/write random access memories (RAMs) for data variable storage. The processor board can be strapped to use a pair of high speed (2K x 8 bit) or (8k x 8 bit) PROMs to obtain either 2K or 8K words of no-wait-state program memory. For TACNETSIM, the boards will be strapped to use 8K PROMs, due to availability and for future program expansion. Since, all software development is done in TMS320C25 ASSEMBLY language, 8K words of memory is sufficient to perform a large set of processing algorithms.

Analog Interface

The analog interface on the processor board is based around the TI TCM2916 μ-law coder/decoder (CODEC) [4]. This device performs the analog-to-digital (A/D) conversion of the input signal, and the digital-to-analog (D/A) conversion of the simulator

output signal. The TCM2916 contains an on-chip anti-aliasing filter and serial interface which minimizes the external components required for implementation. The analog interface master clock and the sample framing pulse are derived from the TMS320C25 processor. An overview of the analog interface is shown below in figure 4.

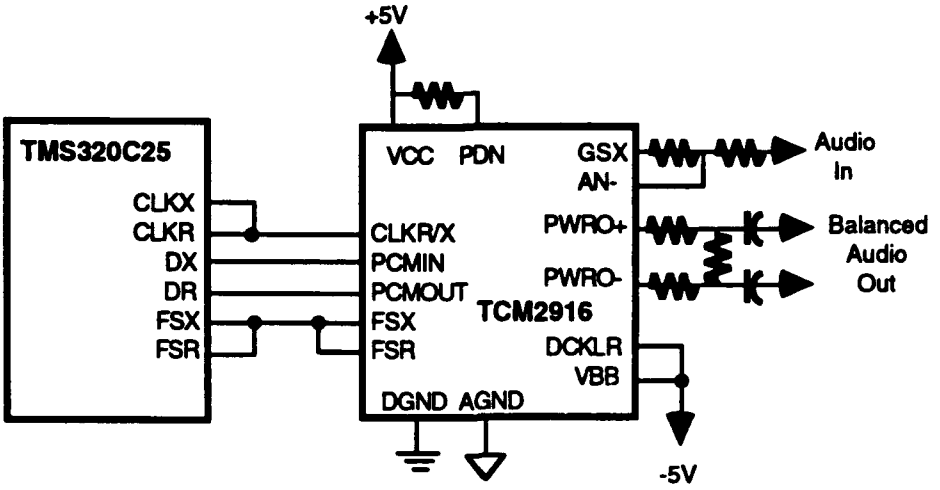


Figure 4: Processor Board Analog Interface

Control Interface

The external control interface for the TACNETSIM processor board is provided via an 8-bit parallel data input port on the TMS320C25. An octal line driver, the 74HC540, is configured as PORT 0 and is read by the TMS320C25 to input an 8-bit control word from an external controller. Figure 5 shows an overview of the control interface hardware. The BIO status pin of the processor is used as an external board select line to activate an external control input word. This is a software-polled status pin and therefore external control updates do not interrupt execution of the critical analog interface routines based upon an internal timer-interrupt cycle. To address a particular processor board, the external controller must hold the BIO pin low when transmitting new control words. In addition, the BIO pin must never be asserted unless valid, static data exists on the 8-bit input port. We discuss further details of the external control protocol in the software design section.

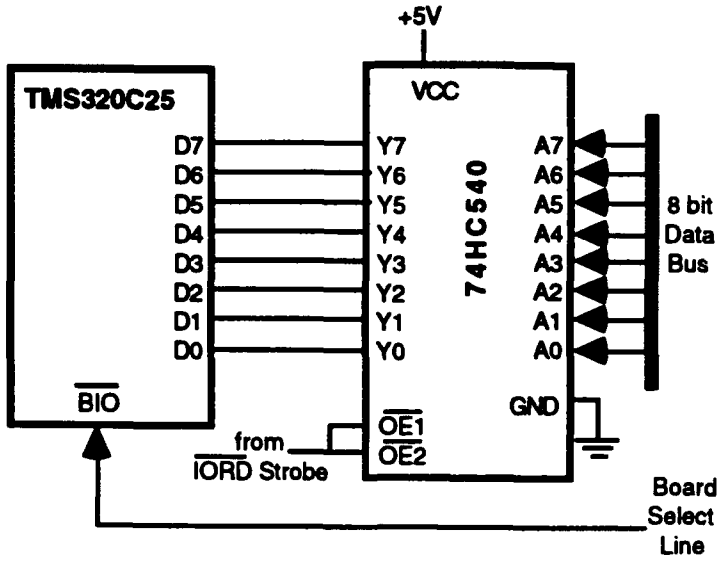


Figure 5: External Control Data Interface

SOFTWARE DESIGN

In this section, we describe the software design for the initial TACNETSIM processor board based upon the NRL VSM hardware. The processor board for TACNETSIM uses the TMS320C25 DSP to perform five main software functions, as follows:

- Processor and Program Initialization
- Analog Input/Output Control
- Propagation Model Simulation
- Simulation Parameter Control
- System Error Control

We give a brief description of each of these five processing components. Other than the initialization routines, all software routines were designed to meet real-time operating requirements.

Initialization

Upon receiving power or a reset signal the processor performs an initialization sequence. Since the TMS320C25 DSP has several programmable configurations, the first function performed is to initialize the desired processor mode and memory map. Next, all simulation variables are initialized to their default condition. In addition, all buffers used within the simulation software are cleared.

Prior to starting the real-time processing, the doppler spread low pass digital filters are pre-excited with a set of independent Gaussian random variables to eliminate the undesirable effect of the impulse response delay time. After the filters are initialized, an internal timer interrupt is activated to provide strict timing control of the analog interface handling routine.

Analog I/O Control

As stated above, a timer interrupt routine provides strict timing control for the CODEC analog interface and for the real-time processing routines. The timer interrupt register contains a number which decrements every master clock cycle. When zero, this register is reinitialized and a timer interrupt is generated. We set this timer register to generate interrupts at a rate of 7200 times a second. Within the timer interrupt service routine, a transmit/receive frame pulse is generated to set the CODEC analog sampling rate at 7200 samples per second. The formula for determining the timer register value for a specific sample rate is the following.

$$Treg = (Processor\ Clock/4)*(1/Sample\ Rate)-1$$

The TCM2916 CODEC analog interface chip used in this design performs signal companding by representing 8-bit analog samples in μ-law format. Companding results in a larger dynamic range for the CODEC operation. Non-linear μ-law operation results in a quantization noise floor effectively equivalent to a 12-bit linear quantizer. Since a linear number representation is required for the processing algorithms, a software conversion between μ-law and linear number representation is performed by the TMS320C25 for each transmit and receive sample.

Propagation Model Simulation

As stated previously, the initial objective of the TACNETSIM system is to provide a real-time channel simulation for an HF groundwave IBG network. Figure 1 shows the propagation model with its two basic components: a groundwave ray and a skywave ray. During channel simulation, the following signal parameters are controllable within the TACNETSIM simulation processor subsystem.

- System delay (0.1-10 ms)
- Groundwave propagation delay (0.1- 30 ms)
- Skywave propagation delay (minDelay[1]-30 ms)
- Relative groundwave and skywave signal strengths
- Skywave Doppler Spread (0.1-10 Hz SS)

System Delay

The system delay is controllable from 0.1-5 ms in length. This delay can be used to simulate the processing delays anticipated from any external components not part of the laboratory test loop (e.g. communication security devices).

GW Simulation

The groundwave propagation delay is controlled from 0.1-30 ms. The incremental control occurs in steps equal to 139 μsec, which is equivalent to a range of 41 kms (22 nautical miles). When the propagation range and channel conditions support a skywave component, the groundwave component and/or skywave component will be attenuated to achieve the desired relative signal strength for the analog output. Although signal fading is being simulated, a constant time-averaged signal strength is maintained to ensure maximum usage of the limited dynamic range of the VSM analog interface. This composite signal is supplied to the TACNETSIM analog baseband subsystem which performs absolute signal strength control of numerous multi-node sources to achieve the specified signal-to-noise

---

[1] minDelay is determined by the system delay value. With a system delay of 0.0 ms, the minimum skywave delay time is equivalent to the Hilbert transform filter delay.

ratio. Noise and interference sources are added externally to TACNETSIM within the RF and/or analog baseband subsystems.

HF Skywave Simulation

Software routines are used to process the HF ionospheric propagation simulation outlined in Figure 3. Figure 6 presents an outline of the skywave component simulation process.

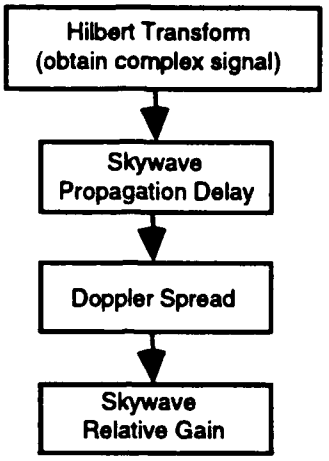


Figure 6: HF Skywave Simulation Process

Hilbert Transform

A Hilbert transform is performed on the real data to obtain the imaginary component of a complex signal representation. The impulse response of the Hilbert filter is delayed by half the filter length. Therefore, the filter is chosen to be of odd length so that the real signal component can delayed by exactly half the filter length. The input signal (real component) and its Hilbert transform (imaginary component) use separate delay line buffers. The delayed real component is then matched with the Hilbert transform filter output, which represents the imaginary signal, to form a complex signal pair.

Delay

Since the skywave simulation requires complex number arithmetic operations, two delay line buffers are implemented in software. The first delay line buffer is for the real

signal component, while the second is for the imaginary signal component. Figure 7 shows an overview of the delay line buffers and associated data pointers implemented by the software. Delay line buffers are implemented as circular buffers in software and the real-component delay line is shared by several processes with pointers to the appropriate data locations.

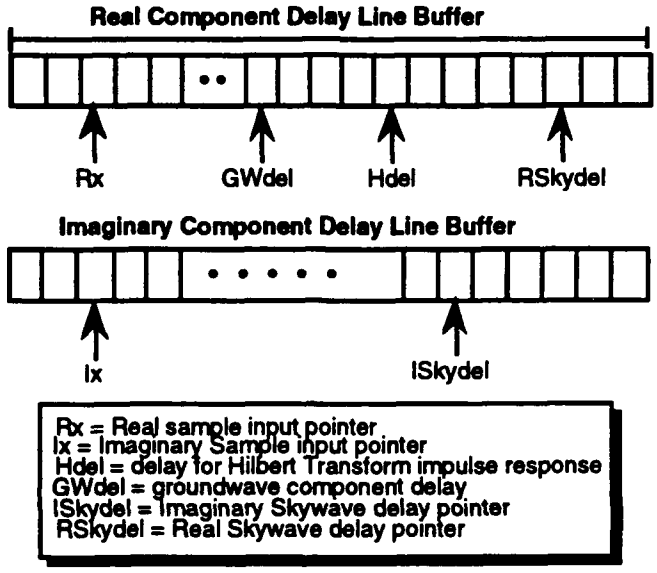


Figure 7: Delay Line Buffers and Data Pointers

For a given delay buffer size of N, the pointers in Figure 7 are calculated as follows.

$$Rx(i) = (Rx(i-1)+1) \bmod N$$

$$Ix(i) = (Ix(i-1)+1) \bmod N$$

$$Hdel(i) = (Rx(i)+(N-0.5FiltLength)) \bmod N$$

$$GWDel(i) = (Rx(i)+(N-GwDel)) \bmod N$$

$$RSkyDel(i) = (Rx(i)+(N-HDel-SkyDel) \bmod N$$

$$ISkyDel(i) = (Ix(i)+(N-SkyDel) \bmod N$$

In the above equation for RSkyDel(i), it can be shown that the minimum achievable skywave delay is equal to the Hilbert transform delay. This presents a limitation in the

simulation of short range communication links, where skywave propagation exists at near vertical incidence. If a shorthop skywave simulation is desired, an additional constant delay can be added to the groundwave delay. This allows the simulation of short relative delays between groundwave and skywave, but increases the overall system processing delay.

Doppler Spread

Following the tapped delay line, a set of independent tap gains are applied to the complex signal components. Tap gains are computed from a set of separately filtered and independently generated gaussian random variables with zero mean and equal variance. The tap gain process produces an overall fading-signal envelope with a Rayleigh distribution. The phase of the resultant signal should also have a uniform phase distribution, since independent Gaussian processes are applied in quadrature.

The performance of the doppler-spread processor is related to the accuracy of generating the independent Gaussian processes. The Gaussian distribution is approximated by first generating a set of uniformly distributed random variables and then summing these variables to approximate a normal distribution based upon the law of large numbers.[2] Two software routines have been examined to generate a set of 12-bit uniformly distributed random numbers. The first routine is based upon a 32-bit maximal length linear shift register, where the linear feedback register is shifted twelve times and the random number is taken as the result in the 12 LSBs. The second routine is based upon a 16-bit multiply algorithm similar to the standard C language library *rand()* function [7], implemented as the following equation:

$$\text{Ivar}(n) = (J * \text{Ivar}(n-1) + \text{constant}) \bmod 65535$$

The 12-bit number distributions resulting from these two algorithms where tested for the generation of 10,000 sample points. Appendix B presents the results, and shows that the

---

[2] If $U_1, U_2, \ldots, U_n$ is a sequence of n uniform deviates, then $X_n = \left(\sum_1^n U_i - \frac{n}{2}\right) * \left(\sqrt{\frac{n}{12}}\right)$ is asymptotically distributed as a normal random deviate [6].

distribution between the 11-bit range, 0-2047, for both algorithms is sufficiently close to uniform. The resulting Gaussian approximated distribution, based upon the sum of 12 random numbers, is shown to be of acceptable accuracy. The generated probability density function (pdf) is close to a 1% error in mean and standard deviation from a normalized Gaussian pdf. Given the accuracy of the resulting simulations, the 16-bit multiply algorithm has been chosen for its processing efficiency.

A programmable doppler spread bandwidth is realized by applying a low-pass filter to the independent Gaussian processes and varying the interpolation time between the filter outputs. A low-pass filter is applied to each Gaussian process and is periodically updated based upon a specific interpolator block size. The ideal Gaussian power spectrum resulting from the FIR filter output is given as

$$P(f) = \left[\frac{1}{\sqrt{2\pi\sigma_i^2}}\right] e^{\left[\frac{-f^2}{2\sigma_i^2}\right]} \qquad (1)$$

where σ_i^2 is the single-sided doppler spread specified in Hertz [5]. A FIR filter design program was developed and used to obtain a 99-tap 16-bit FIR filter accurate to within 1% of the ideal frequency spectrum.

A memory-efficient software routine to control the doppler-spread rate is realized by implementing a single, fixed FIR filter in combination with programmable interpolator sizes to perform a range of desired spread rates. Between filter updates, the tap gain values result from a linear interpolation between the old and the new filter output values. A linear interpolator is used and is equivalent to filtering the output by $(\sin x/x)^2$ with the first null appearing at the update rate. Together, the low pass filter and the linear interpolator apply the desired Doppler spread bandwidth characteristic to the output signal. For an accurate estimate of the desired Gaussian spectrum, it is recommended the update rate of doppler spread tap gain filter be greater than 32 times the desired spread rate.

Signal Gain

When the propagation range and channel conditions support both groundwave and skywave components, the groundwave and/or skywave components are attenuated to achieve the desired relative signal strength for the analog output. As mentioned previously,

a constant composite signal strength is maintained in order to maximize usage of the limited dynamic range. Since a skywave component with doppler spread has a Rayleigh amplitude fading distribution, the ratio of the maximum amplitude to root-mean-square (rms) amplitude increases. In order to avoid the likelihood of output signal saturation, the rms output level is set in relation to the maximum output level to include a comfortable overhead signal strength margin of 3-6 dB.

Simulation Parameter Control

The external control of simulation parameters is an important issue in TACNETSIM design and operation, since the objective is to simulate dynamically changing network topologies and channel conditions. Numerous, time-varying communication channel and network parameters, including communication range, transmit power, sea condition, antenna angles, antenna types, and ionospheric conditions, are all translated into updated simulation control parameters for the channel simulation processor subsystem and other TACNETSIM subsystems during testing. We outline here the software control protocol and the specific programmable parameters updated within the simulation processor.

Simulation Parameters

The propagation delay of the communication channel is the first of several parameters controlled within the processor subsystem. Propagation delay is the difference between the transmission time and the reception time of a given signal. For groundwave components, propagation delay is related to the distance between the transmitting and receiving platforms along the surface of the earth. For tactical battle group communication links, this can be several hundred nautical miles. In the case of skywave components, propagation delay is a function of the virtual reflection height of the ionosphere and the distance between platforms. When it exists, the reflected signal path from the ionosphere is longer than the groundwave path and results in a multipath condition at the receiver.

The existence of a measurable, reflected ionospheric signal is a complex function of multiple variables, including range, latitude, diurnal effects, sunspot number, seasonal changes, antenna takeoff angle, transmission power, frequency, and other anomalous effects (e.g. sporadic E-layer). Although no absolute ionospheric communication model exists, several well-accepted software prediction packages are available to model the

expected effects given a set of input parameters. Some of these prediction packages can interface to real-time ionospheric sounding data to obtain a more accurate picture of the day-to-day fluctuations.

The TACNETSIM control subsystem makes use of ionospheric modeling software (e.g. IONCAP, PROPHET) to obtain predicted propagation models for its generated network communication scenario. The control subsystem translates the results from these software prediction models into a set of parameters for the simulation processor, including signal delay, signal strength, and doppler spread bandwidth. Antenna modeling can be included in the propagation modeling software to obtain more realistic results in terms of received signal strength of groundwave and skywave components. Communication channel quality conditions based upon CCIR standards (e.g. Poor, Fair, Good) can also be established on a link-by-link basis independent of any ionospheric modeling software.

Groundwave component simulation requires the determination of groundwave signal strength. Again, the TACNETSIM control subsystem uses a software prediction model (e.g. Barrick model) to obtain signal attenuation results for nautical range and sea state condition. The combined attenuation results from the groundwave and skywave modeling are translated into a relative signal strength parameter for the simulation processor. An absolute attenuation figure is translated by the control subsystem and communicated to the analog baseband subsystem.

Simulation Control Protocol

In this section, we describe the specific control protocol used by an external controller to communicate with the simulation processor subsystem. The external control hardware has been shown in Figure 5 and is based upon an 8-bit parallel data interface. The BIO status pin of the processor is used as a polled status line to trigger the transfer of an 8-bit control word. Upon receiving a new 8-bit control word, the simulation processor determines the type and value of the control word and takes appropriate action. Table 1 is the present set of control word bit assignments and their resultant action.

8-BIT CONTROL WORD	ACTION
00000000	Reset Simulator Command
00011111	Update Parameters Command
001xxxxx	Set System Delay (xxxxx = delay)
010xxxxx	Set Groundwave Delay (xxxxx = table value)
011xxxxx	Set Skywave Delay (xxxxx = table value)
100xxxxx	Set Doppler Spread (xxxxx = table value)
101xxxxx	Set Relative Signal Strength (xxxxx = table value)
110xxxxx	Presently Unused
111xxxxx	Presently Unused

Table 1: Control Word Protocol Assignments

System Error Control

There are several error control procedures contained within the simulation software to ensure proper operation. Perhaps the most important is the non-real-time error trap, which upon detection of an attempt to execute a simulation in non-real-time, shuts down the output of the system. This prevents non-real-time processing errors from translating into communication system errors during testing.

Due to the fixed-point processing and probabilistic nature of the simulation, there is a possibility of undesirable arithmetic underflows and/or overflows occurring in the system. All software algorithms have been designed to avoid this situation by proper pre-scaling and post-scaling of numbers. In the unlikely event an overflow does occur, the processor has been set in saturation mode. Attempts to represent numbers outside of the processor's range cause the number to be set as the highest positive or lowest negative number. This avoids the disastrous analog result from small fixed-point overflows, where a large positive number can become a negative number.

FUTURE PLANS

In this report, we addressed the initial design and development of the simulation processor subsystem for the NRL TACNETSIM narrowband network simulator. The initial objective has been to achieve a narrowband simulation processor for an HF IBG

narrowband network. This simulation included one groundwave and one skywave component for each communication path. The future development of a long haul skywave simulation to support several different skywave components is planned.

As mentioned previously, the VSM processor board has limited dynamic range and analog bandwidth (i.e. 3.6 kHz). Future plans exist to upgrade the TACNETSIM system by replacing the VSM-based simulation processor subsystem with a set of high performance DSP processor cards and multi-channel analog interface boards. High performance floating point DSP cards (e.g. TIGER 30) enable multiple communication link simulations to be performed on a single card. In combination with multiple channel analog interface boards, the future processor subsystem will have greater flexibility and increased analog bandwidth. The increased analog bandwidth shall allow simulation of wider bandwidth communication systems, such as 25 kHz tactical satellite channels.

CONCLUSION

A low cost digital signal processor (DSP) based communication channel simulator has been designed and implemented. An array of these channel simulator processors will form a subsystem of the multi-node tactical network simulator (TACNETSIM) being developed by the Information Technology Division at the Naval Research Laboratory (NRL). The present software simulates an HF IBG communication network, where each communication path contains a groundwave and/or skywave propagation component. The skywave component is simulated using the Watterson ionospheric propagation model. Simulation parameters are software controllable and are externally updated over time to perform dynamic communication network modeling.

The array of independently controlled simulation processors used in TACNETSIM allows a new type of on-line dynamic network channel simulation to be performed. Actual tactical communication network scenarios can be modeled and then simulated in a laboratory environment using actual transmit and receive components within the simulation loop. Communication network performance can be stressed under repeatable, controlled conditions to assist in network system design and analysis. In addition, future program cost savings are realized by reducing the need for expensive communication network field testing efforts.

ACKNOWLEDGMENTS

The author is indebted to R. Brian Adamson for his assistance in the development of the simulation approach taken and for the development of the tap gain FIR filter design program. Doug Waggy provided valuable assistance in performing real-time implementation and performance studies. Additional thanks to the NRL Tactical Networks Section for continuous support and encouragement of this project.

REFERENCES

1 C. C. Watterson, J. R. Juroshek, and W.D. Bensema, "Experimental verification of an ionospheric channel model", ESSA Tech. Report, ERL 112-ITS 80, 1968.

2 David L. Tate, "The Variable Speed Modem Modulator Design", NRL Report 9169, March 20, 1989.

3 Joseph P. Macker, "Software Assessment for an HF Channel Simulator using the VSM DSP Hardware Module", Memorandum for the CSD Project File, June 1992.

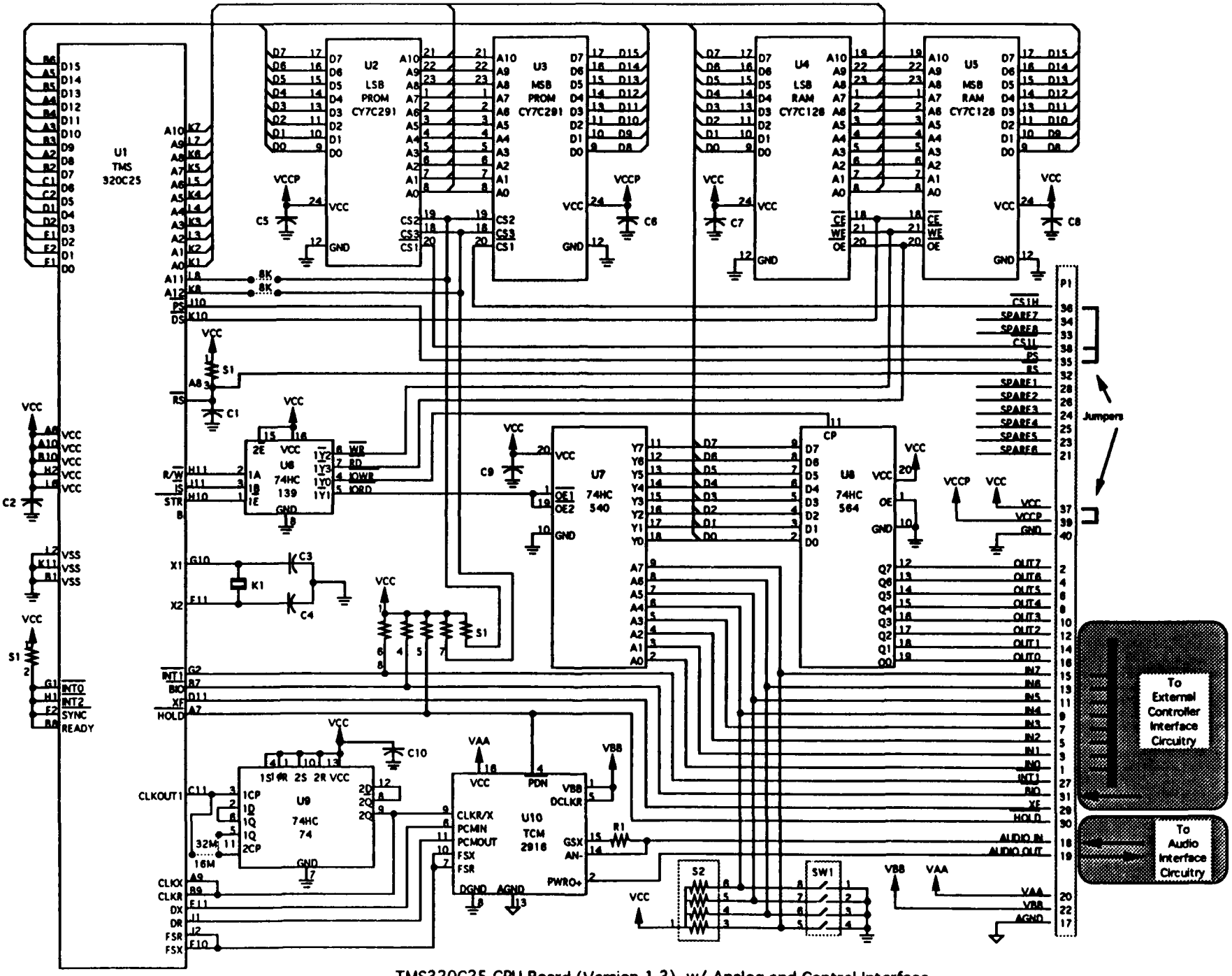
4 Telecommunications Circuit Data Book, Texas Instruments, 1986.

5 C. C. Watterson,, Ax, G.G., Demmer, L.J., and Johnson, C.H., " An Ionospheric Channel Simulator", ESSA Research Laboratories Technical Memorandum ERLTM-ITS 198, September 1969.

6 Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables, U.S. Department of Commerce, National Bureau of Standards, Applied Mathematics Series 55, U.S. Govt. Printing Office, May 1968.

7 Kernighan, B.W., Ritchie, D.M., " The C Programming Language", ATT Bell Laboratories, Prentice Hall, 1988.

APPENDIX A:
SIMULATION PROCESSOR HARDWARE

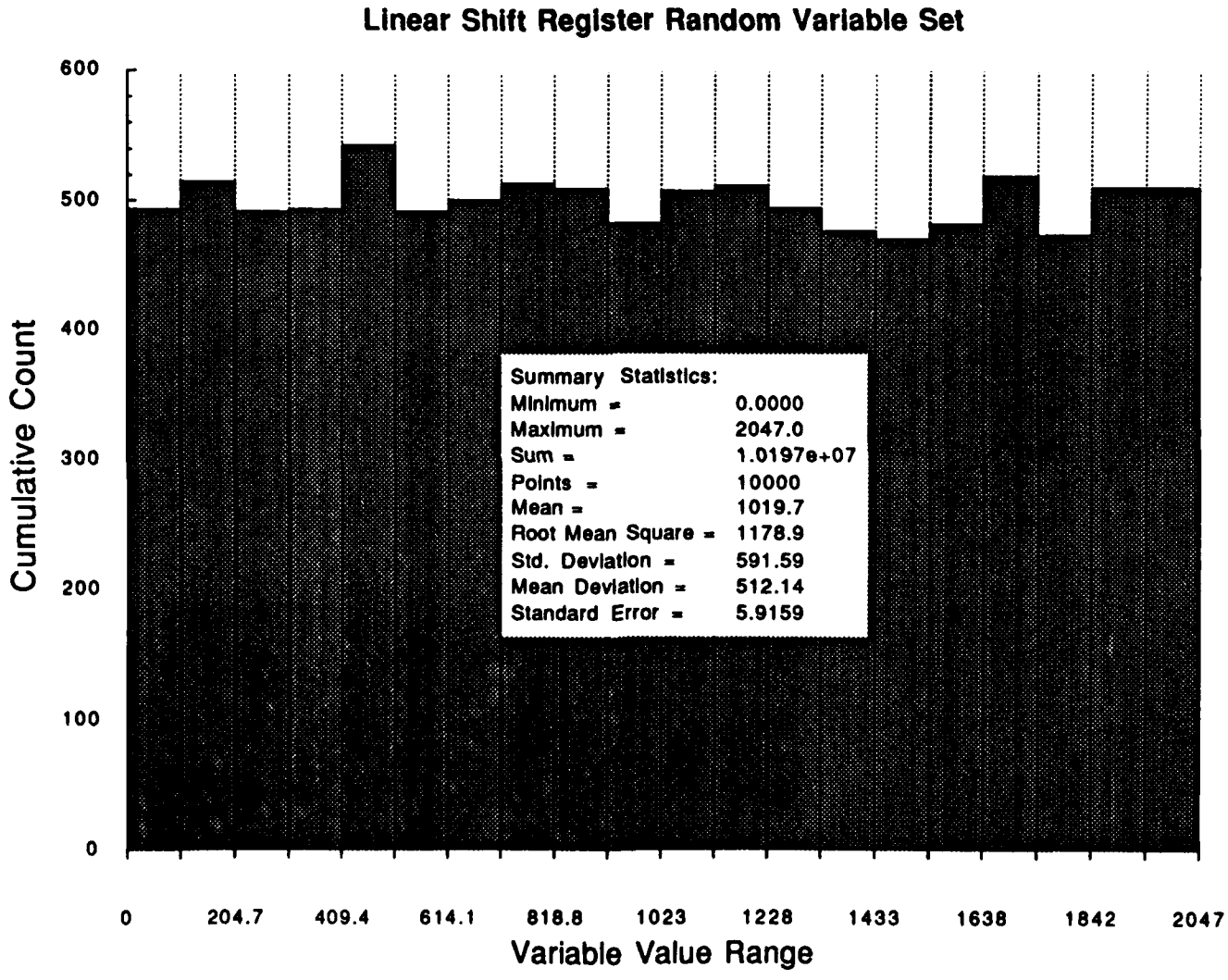


TMS320C25 CPU Board (Version 1.3) w/ Analog and Control Interface

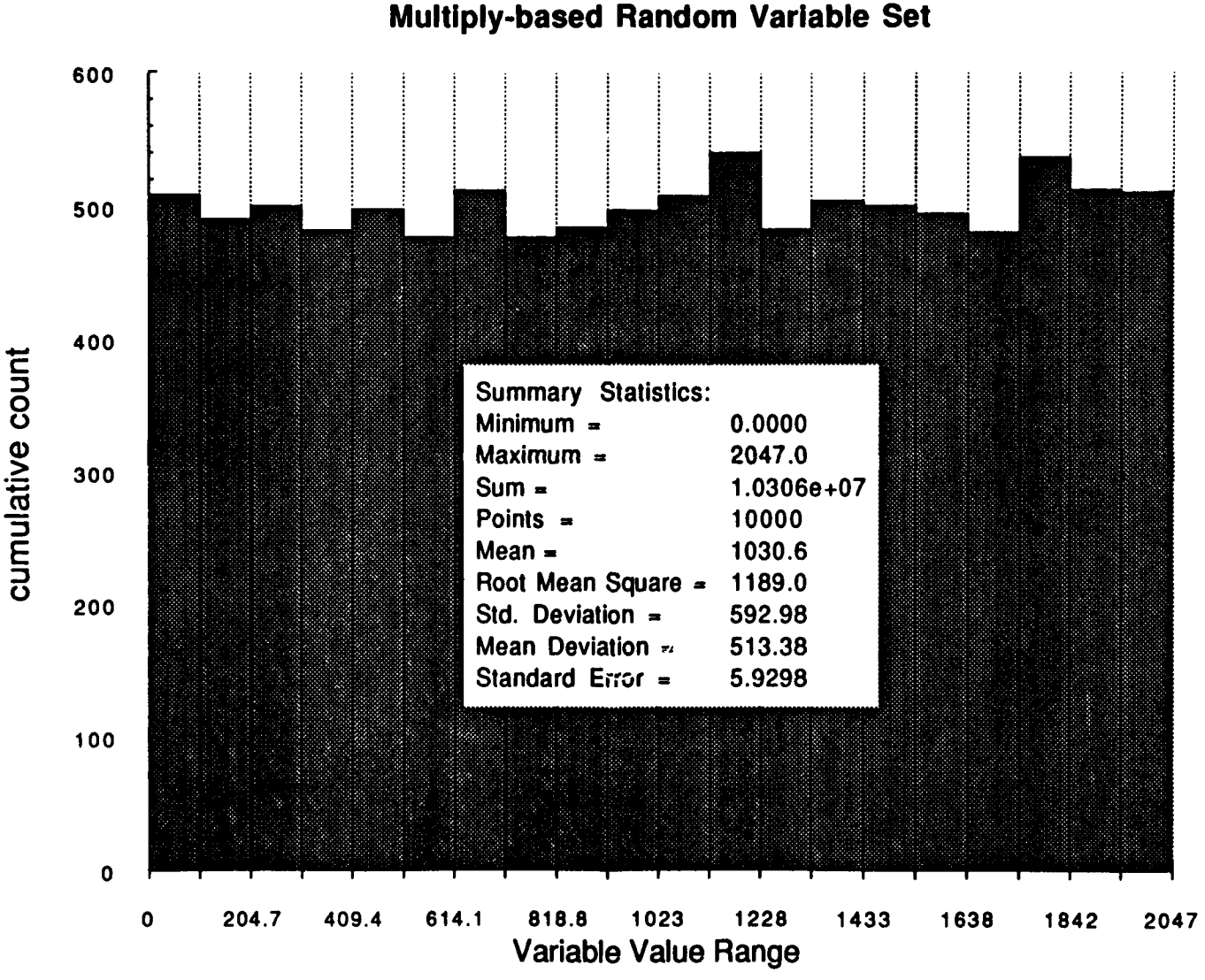
APPENDIX B:

RANDOM VARIABLE PROCESS SIMULATION RESULTS

RESULTS FROM RANDOM VARIABLE GENERATION SOFTWARE BASED UPON A 32-BIT MAXIMAL LENGTH SHIFT REGISTER



RESULTS FROM RANDOM VARIABLE GENERATION SOFTWARE BASED UPON A 16-BIT MULTIPLY ALGORITHM



RESULTS FROM GAUSSIAN VARIABLE GENERATOR BASED UPON A SUM OF 12 UNIFORM VARIABLES