Performance Analysis of an all Digital BPSK Demodulator

John M. Gevargiz

Jet Propulsion Laboratory, Telecommunication Systems Section, Pasadena California, 91109

Abstract

A simulation algorithm was developed to analyze the performance of the binary phase shift key (BPSK) Costas loop coupled to the symbol sync loop (SSL). This algorithm was developed using COMDISCO's Signal Processing Work System. inputs to the algorithm included the signal parameters and receiver specifications, such as Costas and Symbol Sync Loop bandwidths and update rates. The analysis was performed for various signal-to-noise ratios and loop parameters.

The bit-error rate (BER), the SSL, Costas loop tracking variances, and the receiver acquisition time results include the two-way coupling effects between the Costas loop and the symbol sync loop. To provide statistical stability, final results were generated by averaging the simulation outputs; using various random noise seeds, initial carrier phases, and symbol timings.

The tracking variance and BER results were further compared to their theoretical and experimental counterparts. This comparison has shown that simulation results are within 0.25 dB of their theoretical and experimental counterparts. Based on these results, the simulation models can be used for predicting the demodulator's performance under various tracking conditions.

1. introduction

The existing Deep Space Network (I EN) receivers of the Jet l'repulsion I .aboratory (JPL) and National Aeronautics and Space Administration (NASA) will be replaced by a digital receiver called the Block V Receiver [1, 2]. The input frequencies to the Block V receiver are the S-band (2200 to 2300 MHz), X-band (8400" to 8500 M 117.) and Ka-band (31.8 to 32.4 G] Iz) downlink signals, received from the deep space satellites exploring the solar system under the direction of JPL and NASA. T'he Block V receiver supports data formats that use squarewave or sinewave subcarrier modulated symbols with a residual carrier, as well as a binary and quadrature phase shift keyed (BPSK and QPSK) suppressed carrier.

This receiver uses both software and hardware for processing the received downlink signals. T'he receiver sampling frequency is 160 M} Iz [1]. The high-sped processing is performed using the GaAs application-specific integrated circuit (ASIC) [3], and real-time software is used for implementing the receiver tracking loop filters and data detection algorithms. This architecture allows for implementing second- and third-order tracking loops that are easily re-configured with bandwidths that range from 0.1 Hz to 1.0 kHz.

The received signals from NASA/JPL deep space satellites contain science and engineering data, The DSN stations make every possible effort to minimize interference and to receive and decode the signal. Some of these efforts include using the predicts of the Doppler information and signal-to-noise ratio (SNR) at the ground stations to optimize the receiver parameters. Some of these parameters include the free running frequency and frequency rate of the numerical control oscillator (NCO), and bandwidths of the loop filters. T'he optimum receiver parameters are obtained using the analysis results of the receiver.

This paper discusses a simulation algorithm that is developed for studying the acquisition, tracking, and error performance of the BPSK demodulator, a major element of the Block V receiver. T'his demodulator consists of two coupled subsystems: the Costas loop and the symbol sync loop (SSL). T'performance analysis includes the convolutional encoder and the update filter. '1 'he update filters are used to update the SS1. and Costas loop-filter inputs at 1 kHz and 10 kHz, respectively.

Figure 1 illustrates a simplified block diagram of the digital receiver's BPSK demodulator. The simulation algorithm used for determining the performance of the SS1 . and Costas loop is implemented at the symbol rate to avoid excessive simulation time, using the signal processing worksystem (SPW) [4]. Therefore, each simulation cycle corresponds to one *symbol* increment i nstead of a *sample* increment. Figure 2 illustrates the SPW implementation of the system depicted in Fig. 1 that includes the signal generator, BPSK demodulator, lock detectors, and signal storage blocks. And, Fig. 3 illustrates the

detailed implementation of the BPSK demodulator; it shows the Costas loop, symbol sync loop, and convolutional decoder.

This roach significantly reduces the computation time as compared to simulating the system at the sample rate. For example, at the sampling frequency of 160 MHz and 400 kbps, there will be 400 samples per symbol; therefore the simulation at the sample rate will be approximate y 400 t i mes longer compared to the symbol rate simulation.

Sections 2 and 3 present the BPSK Costas loop and SS1, models, respectively. And, Section 4 presents the acquisition, tracking, and BER results.

2. BPSK Costas Loop Models

The functional block diagram of the suppressed carrier BPSK detector is shown in Fig. 1. For the sake of simplicity, symbol timing is referenced to the SSL NCO output. Hence the normalized received symbol time is denoted by a time dependent variable, $\tau_{SSL}(1)$. The time difference between the N(O timing and the received signal symbols is denoted by $\tau_{SSL}(t)T$. In practice, the NCO tracks the received symbols, seeking to diminish $\tau_{SSL}(t)$. Therefore, the BPSK waveform at the front end of the receiver is given by

$$S(t) = \sqrt{2}d(t + \tau_o T)\sin\left[\omega_o\left(t + \tau_o T\right) + \theta_o\right] + n(t) \tag{1}$$

where

$d(t) = \pm 1$	= equiprobable random symbols (convolutional encoded bits)
$T = 1/R_{symbol}$	= symbol period
ω_o	= BPSK carrier frequency
$\tau(t) = \tau_o - \tau_{SSL}(t)$	= symbol timing error normalized to a symbol period
$\tau_{SSL}(t)$	= SSL estimated symbol timing for time t
τ_o	. symbol timing, normalized to a symbol period, $-0.5 \langle \tau_o \leq 0.5$
θ_o	carrier phase, $-\pi \langle 9_0 \leq \pi$
$n(t)$	= additive white Gaussian noise

From Fig. 4, both analytical and simulation tools are used to obtain the performance of the BPSK Costas loop. The outputs of the integrate-and-dump filters are derived analytically, whereas simulation is used for obtaining the outputs of the loop filters, numerically controlled oscillator, update filter, and phase detector.

Section 2.1 presents the analytical derivations of the Costas loop, and Section 2,2 presents the simulation models.

2.1 Costas Loop Analytical Models

From Fig. 1, the outputs of the in-phase (1) and quadrature-phase (Q) arm filters are given by,

$$\begin{aligned} S_I(t) &= d[t+\tau(t)T]\cos\{\Delta\omega(t)[t+\tau(t)T]+\theta(t)\}+n_I(t) \\ S_Q(t) &= d[t+\tau(t)T]\sin\{\Delta\omega(t)[t+\tau(t)T]+\theta(t)\}+n_Q(t) \end{aligned} \tag{2}$$

where

$\Delta\omega(t)=\omega_o-\omega_{nc}(t)$	= carrier frequency error
$\omega_{nc}(t)$	= NCO frequency
$\theta(t)=\theta_o-\theta_{nco}(t)$	= estimated carrier phase error
$\theta_{nco}(t)$	= NCO phase
$n_I(t)$ & $n_Q(t)$	= I and Q components of the AWGN

Since the system of Fig. 1 is a digital system, Eq. (2) can be expressed as

$$\begin{aligned} S_I(iT_s) &= d(iT_s+\tau_iT)\cos\left[\Delta\omega_i(iT_s+\tau_iT)+\theta_i\right]+n_I(iT_s) \\ S_Q(iT_s) &= d(iT_s+\tau_iT)\sin\left[\Delta\omega_i(iT_s+\tau_iT)+\theta_i\right]+n_Q(iT_s) \end{aligned} \tag{3}$$

where

i	= sampling index,
T_s	= sampling period = $1/F_s$
$\tau_i=\tau(iT_s)$	i^{th} sample of the timing error (SS1. NCO output)
$\Delta\omega_i = \Delta\omega(iT_s)$	= i^{th} sample of the frequency error (Costas loop NCO output)
$\theta_i = \theta(iT_s)$	= i^{th} sample of the carrier phase error (Costas loop NCO Output)

From Fig. 4, the phase of Eq. (3), $\left\{\Delta\omega_i(iT_s+\tau_iT)+\theta_i\right\}$, is analytically calculated by using the output of the NCO and the estimated symbol timing. This calculated phase term and symbol polarity are then used for calculating the outputs of the matched filters.

From Eq. (3), the outputs of the matched filters implemented using the integrate-and-clump filters can be expressed by

$$M_I(k) = \frac{1}{n}\sum_{m=nk}^{n(k+1)} S_I(mT_s) + \frac{1}{n}\sum_{m=nk}^{n(k+1)} n_I(mT_s)$$
$$M_Q(k) = \frac{1}{n}\sum_{m=nk}^{n(k+1)} S_Q(mT_s) + \frac{1}{n}\sum_{m=nk}^{n(k+1)} n_Q(mT_s) \tag{4}$$

where

$n = \text{Integer}(T/T_s)$	= number of samples per symbol
$S_I(iT_s)$ and $S_Q(iT_s)$	= outputs of the I and Q arm filters
$M_I(k)$ and $M_Q(k)$	= outputs of the IDF
k	= symbol count

For $\Delta\omega_i T \ll \frac{\pi}{2}$ in Eqs. (3) and (4), we can assume $\left[(iT_s + \tau_i T)\Delta\omega_i + \theta_i\right]$ does not change in a symbol period, hence

$$\cos\left[(iT_s + \tau_i T)\Delta\omega_i + \theta_i\right] \approx \cos\left\{(kT + \tau_k T)\Delta\omega_k + \theta_k\right\} \quad \text{for} \quad nk \le i \le n(k+1) \tag{5}$$

where kT and k are the end of the symbol period and the symbol count, respectively.

Figure 5 illustrates the relative symbol timing between the received symbols and the Costas matched filters. From Eqs. (4) and (5) and Fig. 5, we can show that when $\left|\tilde{T}_k - T\right| \ll \tau$ and $\tau_k \le 0$

$$M_I(k+1) \approx \cos\left\{(k+\tau_k)T\Delta\omega_k + \theta_k\right\}\left\{|\tau_k| d_{k-1} + \left[\left(\tilde{T}_k/T\right) - |\tau_k|\right] d_k\right\} + N_I(k)$$
$$M_Q(k+1) \approx \sin\left\{(k+\tau_k)T\Delta\omega_k + \theta_k\right\}\left\{|\tau_k| d_{k-1} + \left[\left(\tilde{T}_k/T\right) - |\tau_k|\right] d_k\right\} + N_Q(k) \tag{6}$$

but for $\tau_k > 0$

$$M_I(k+1) \approx \cos\left\{(k+\tau_k)T\Delta\omega_k + \theta_k\right\}\left\{|\tau_k| d_{k+1} + \left[\left(\tilde{T}_k/T\right) - |\tau_k|\right] d_k\right\} + N_I(k)$$
$$M_Q(k+1) \approx \sin\left\{(k+\tau_k)T\Delta\omega_k + \theta_k\right\}\left\{|\tau_k| d_{k+1} + \left[\left(\tilde{T}_k/T\right) - |\tau_k|\right] d_k\right\} + N_Q(k) \tag{7}$$

where

$$N_I(k) = \frac{1}{n}\sum_{m=nk}^{n(k+1)} n_I(mT_s)$$

$$N_Q(k) = \frac{1}{n}\sum_{m=nk}^{n(k+1)} n_Q(mT_s)$$

$\hat{T}_k$ is the estimated symbol duration; $d_{k-1}, d_k,$ *and* d_{k+1} are the former, the current, and the next symbol polarities, respectively.

The outputs of the matched filters in Fig. 4 can be obtained for each symbol using the estimates of the carrier phase, carrier frequency, and symbol timing. The matched filters are then followed by the update filter, loop filter, and NCO. The simulation models for the Costas filters and the NCO are described in Section 2.2.

2.2 Costas Loop Simulation Models

Figure 4 illustrates the analytical and simulation blocks that are used for the demodulator analysis. in this figure, the analytical and simulation blocks are denoted by dark-blocks and clear-blocks, respectively. From Fig. 4, the I and Q components of the matched filters are first multiplied and then averaged using the update filter. The output of this filter can be expressed as

$$\overline{M}_{IQ}(m) = \frac{1}{u}\sum_{k=mu+1}^{(m+1)u} M_I(k)M_Q(k) \qquad \text{for } R_{symbol} \geq R_{carrier\_update} \tag{8}$$

where

$u = \text{integer}\left(\dfrac{R_{symbol}}{R_{carrier\_update}}\right)$ = number of symbols per update time for Costas loop

$R_{carrier\_update}$ = Costas loop filter update rate (10 kHz, for this analysis)

n = update sample count for the Costas loop

It should be noted that the update filter use u samples as input to generate an output sample. Whereas the Costas loop is implemented at the symbol rate, the loop filter operates at a different rate called the update rate.

From Fig. 4, the output of the update rate filter is first followed by the second-order loop filter operating at the symbol rate, and the loop filter is followed by the NCO that operates at the symbol rate. The

output of the NCO is then used by the Costas phase detector for calculating the matched filter results. Also seen from Fig. 4 and Eqs. (7) and (8) is the coupling to the SS1,. The analytical and simulation models of the SS1, are described in Section 3.

3. Symbol Sync Loop Models

Figure 1 illustrates the functional block diagram of the symbol sync loop and the Costas loop. Similar to the Costas loop, both analytical and simulation tools are used to determine the performance of the SS1, The output of the mid-phase integrator is derived analytically, whereas simulation is used for obtaining the outputs of the update filter, loop filter, and the NCO.

Section 3.1 presents the analytical derivat ions of the SS1., and Sect ion 3.2 presents the simulation methods.

3.1 SS1, Analytical Models

Figure 6 illustrates the relative timing between the received symbols and the SS1. mitt-phaw integrator From Figs. 1 and 6, the output of the mid-phase integrator (windowed 11)1:) is given by

$$W_{IDF}(k) = \frac{1}{2\Psi n}\left\{\sum_{m=kn-J_{\Psi}}^{kn+J_{\Psi}} S_I(mT_s) + \sum_{m=kn-J_{\Psi}}^{kn+J_{\Psi}} n_I(iT_s)\right\} \tag{9}$$

where

J_{Ψ} = integer$\left(\frac{\Psi T}{T_s}\right)$ = number of samples per window, Y'

Y' = single-sidd window, normal ized by the symbol d u rat ion (Y' ≤ ().5)

$S_I(mT_s)$ = output of the in-phase> low-pass filter, given by Eq. 3

From the results of Eqs. (5) and (9), the output of the mici-phase integrator can be simplified to

$$W_{IDF}(k+1) = \cos\{(k+\tau_k)T\Delta\omega_k + \theta_k\}\sum_{m=kn-J_{\Psi}}^{kn+J_{\Psi}} d(mT_s - 1\ \tau_k T) - t\ NW,\ (J(-) \tag{lo}$$

where

$$N_{WI}(k) = \frac{1}{2\Psi n}\sum_{m=kn-J_{\Psi}}^{kn+J_{\Psi}} n_I(iT_s)$$

$\Delta\omega_k$, θ_k, and τ_k are the Costas loop sample frequency error and phase error, and the estimated SS1, timing error, respectively. From Fig. 6 and for T_{est} - T|((τ_k, Eq. (10) can be reduced to

$$
\begin{aligned}
&W_{IDF}(k\text{-}1\,1) \\
&\approx \cos\left[(k+\tau_k)T\Delta\omega_k \text{ -t } \theta_k\right]\{(Y^3 - \tau_k)d_{k-1} + (Y' \text{ -t } \tau_k)d_k\} \text{ -t } N_{WI}(k) \text{ for } |\tau_k| \langle \Psi' \\
&\approx 2\Psi' d_k \cos\left[(k+\tau_k)T\Delta\omega_k + \theta_k\right] + N_{WI}(k) \qquad for\ \tau_k\ )\ o \text{ and } ().5)\,|\tau_k| \rangle \Psi' \qquad (11) \\
&\approx 2\Psi' d_{k-1} \cos\left[(k+\tau_k)T\Delta\omega_k + \theta_k\right] + N_{WI}(k) \qquad for\ \ \tau_k \rangle\ \textbf{o and}\ (0.5)\ |\tau_k|)\ \Psi'
\end{aligned}
$$

The mid-phase integrator is then followed by the phase detector, update filter, loop filter, and the NCO. The simulation models for the SS1, filters and NCO are described in Section 3.2.

3.2 SS1, Simulation Models

Figure 4 illustrates the analytical and simulation blocks for the symbol sync loop. Similar to the Costas loop, the analytical and simulation blocks are denoted by dark-blocks and clear-blocks, respectively. From Fig. 4 and Eq. (1 1), the output of the SS1, update filter can be expressed as

$$
\overline{W}_{ssl}\ (l) = \sum_{k=lv+1}^{(l+1)v} \left\{ W_{IDF}(k) \left[\frac{sgn[M_I(k)] - sgn[M_I(k-1)]}{2} \right] \right\} \qquad (12)
$$

where

v = integer$\left(\dfrac{R_{symbol}}{R_{symbol\ synch\ update}}\right)$	= number of symbols per update time, for SS1
$R_{symbol\ synch\ update}$ –	SS1, update rate, 1 kHz,
$M_I(k)$	k^{th} detected symbol
l	update sample count for the SS1, update-filter

Similar to the Costas loop it should be noted that the update filter uses v samples as input to generate an output sample. Whereas the SS1, updates at the symbol rate, the loop filter operates at the update rate. From Fig. 4, the update filter is followed by the second-order loop filter and the NCO.

Figure 7 illustrates the overall simulation algorithm. The results Of the simulation are the acquisition, tracking, and bit-error rate statistics. These results are presented for various loop parameters in Section 4.

4. Simulation Results

Figure 7 illustrates the simulation setup for investigating the performance of the system depicted in Fig. 1. Table 1 summarizes the signal and the receiver parameters and the corresponding values that are used in this analysis. For each set of parameters, the simulation is iterated for different noise seeds and the initial carrier phase and symbol timing.

The receiver is declared in lock if the phase detector outputs meet the following lock criterion:

$$|\theta_k| \le \frac{\pi}{4} \quad \text{or} \quad |\theta_k| \le \pi + \frac{\pi}{4} \qquad \text{for the BPSK Costas loop and,}$$

$$|\tau_k| \le \frac{T}{4} \qquad \text{for the symbol sync loop}$$

and stays in lock for a period of $10/B_l$ for B_l, the loop bandwidth.

Figure 8 illustrates the BPSK Costas loop tracking variance versus E_b/N_0 for various Costas loop and SS1, bandwidths. Figure 8 also presents the theoretical tracking variance for the BPSK Costas loop. The comparison shows the simulation results to be within a small margin of their theoretical counterpart. It should be noted that the theoretical calculations do not include the coupling effect between the SSL and Costas loop. Figure 9 similarly illustrates SSL tracking variance versus E_b/N_0 for various receiver band widths.

Figures 10 and 11 present the BPSK Costas loop and SSL man time of carrier and symbol timing acquisition in seconds, versus E_b/N_0, respectively. The mean time of acquisition data is obtained by averaging the acquisition time for various noise seeds, initial carrier phase, and initial symbol timing; Figure 12 illustrates the cumulative distribution function (CDF) of the Costas and SS1, acquisition time for the coupled loops.

Figure 13 presents the bit-error-rate results versus E_b/N_0. Also shown in this figure are the experimental BER results from Ref. 5 for convolutional coded data with rate = 1/2 and k= 7. The comparison shows the simulation results to be within 0.25 dB of the measured data.

Acknowledgment

The research described in this paper was carried out at the Jet Prepulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration. The technical support of M. Aung, F. Davarian, 1'. Kinman and 1'. Robbins in the development of this analysis is gratefully acknowledged.

References:

[1] J. Berner, "An Extremely Sensitive Digital Receiver for Deep Space Satellite Communications," Eleventh Annual International Phoenix Conference on Computers and Communications, pp 577-584, Scottsdale, AZ, April 1-3, 1992.

[?] S. Hinedi, "NASA's Next Generation All-Digital Deep Space Network Broadband Receiver," *IEEE Transaction on Communications*, vol. 41 no. 1, pp. 246-257, January 1993.

[3] J. S. Graham, M. S. Thompson, and R. A. Johnson," A GaAs ASIC Chip Set for NASA's Deep Space Network, " 1992 IEEE ASIC Conference and Exhibit, pp 57-63.

[4] COMDISCO Systems Inc., 919 East Hillsdale Blvd., Foster City, CA 94404.

[5] J. Yuen, *Deep Space Telecommunications Systems Engineering*, Jet Propulsion Laboratory, pp. 223-226, July 1992.

Table 1. Simulation **Parameters.**

System Parameters:
Bit Rate . 20 kbps
Costas up-date rate = 10 kHz
SS1. up-date rate = 1kHz
SS1. Window = 0.5 , normalized to a symbol duration
SS1. frequency error = 0.0117.
Number of iterations: 10
Number of simulated symbols: 10,000000
Coding method: convolutional using Rate = 1/2 K= 7

Initial Conditions:
Initial. carrier phase , - $\pi/2 \leq \theta_o \leq$ n/2, randomly selected using uniform distribution
Initial symbol timing, $-0.5 \leq \tau_o \leq 0.5$, randomly selected using uniform distribution
Noise seed: The noise seed is changed for the number of iterations

Variables:
Costas loop bandwidth = 2,5, 10,20,40,80, and 100 Hz
SS1, loop bandwidth = 2,5, and 10 Hz
Carrier frequency error = 1,2.5,5, 10,20,40, and 50 Hz
E_b/N_o = 3 - 5 dB

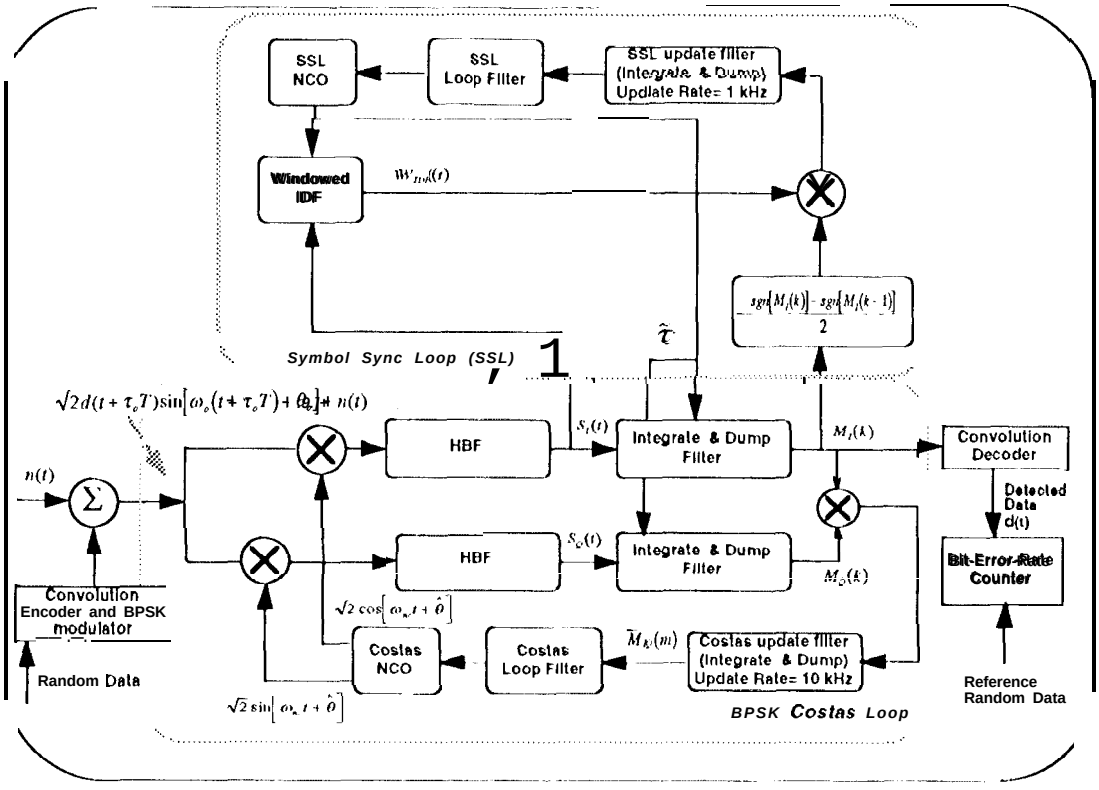

Figure 1. BPSK Costas Loop and Symbol Sync Loop Block Diagram

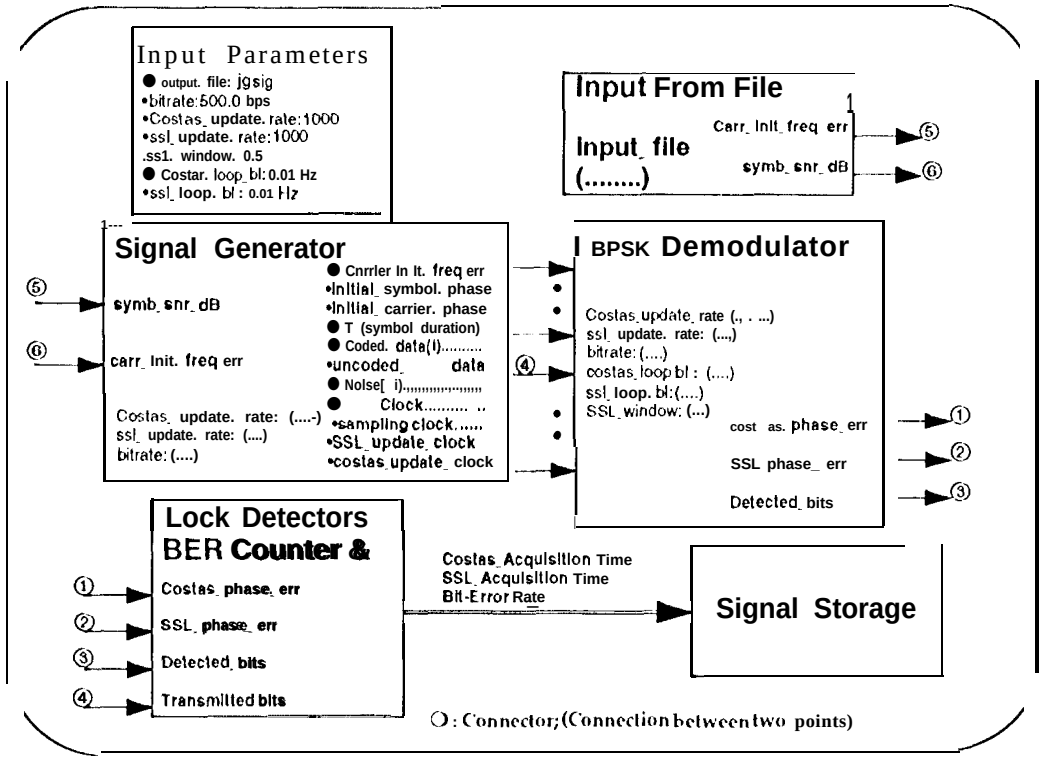

Figure 2. System Level Simulation of the BPSK Demodulator, Using SPW

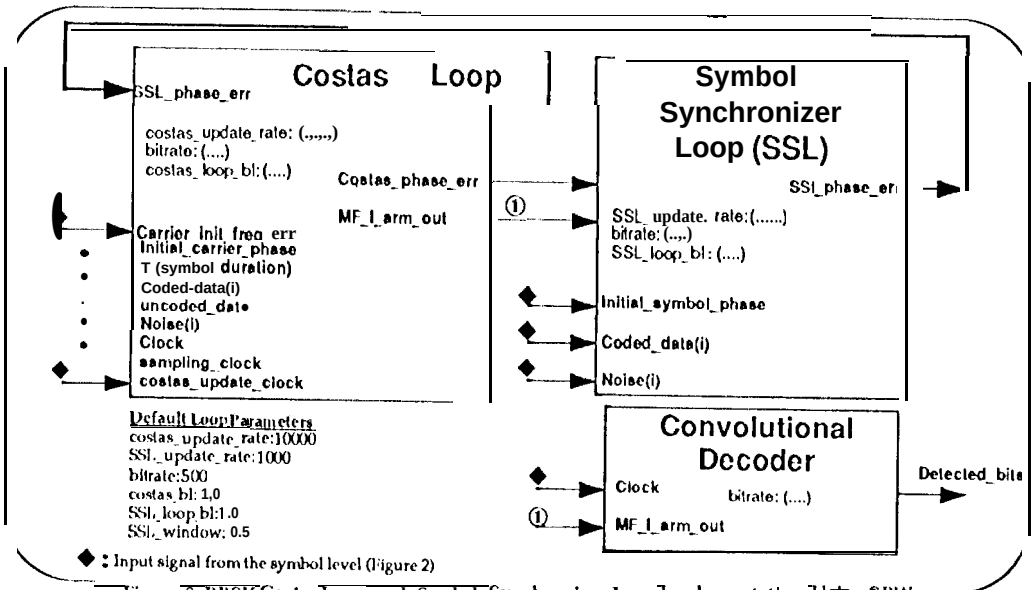

Figure 3. BPSK Costas Loop and Symbol Synchronizer Loop Implementation, Using SPW

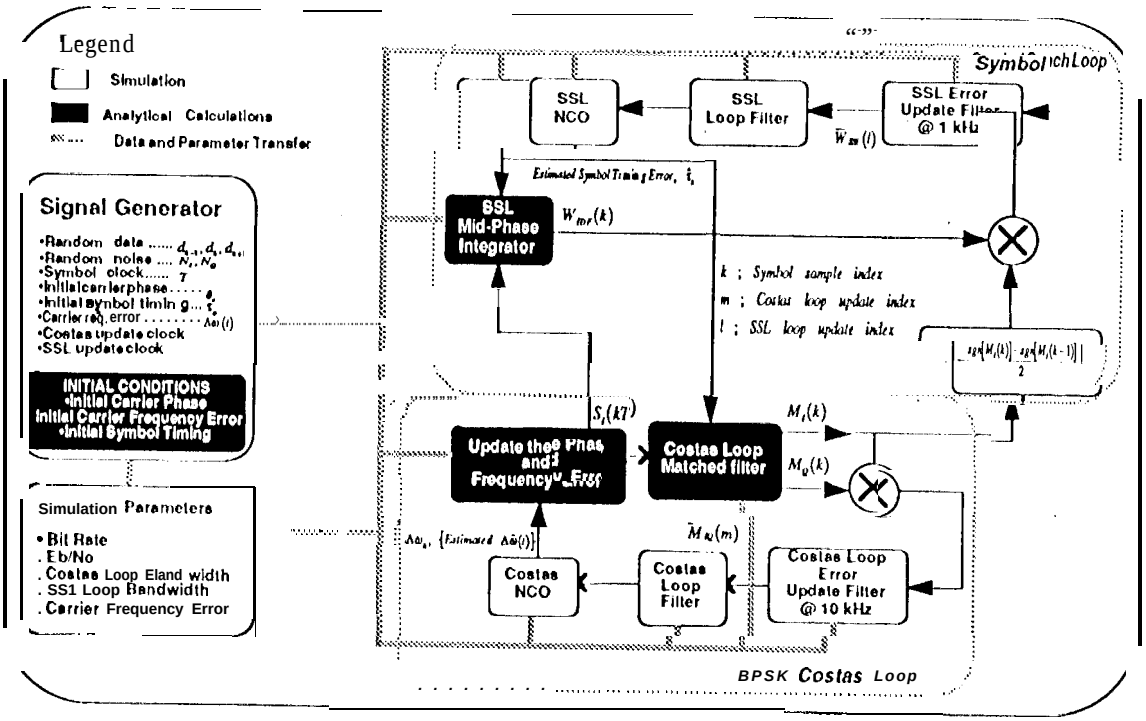

Figure 4. SPW Implementation of the BPSK Demodulator

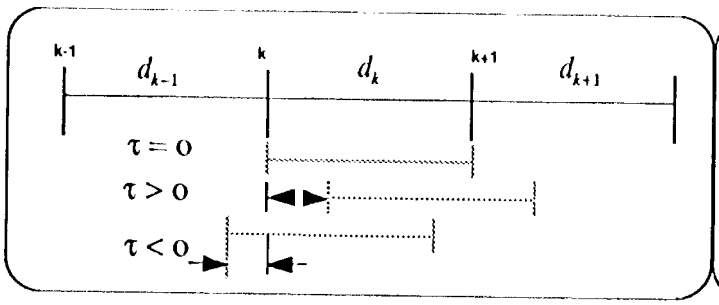

Figure 5. Costas Loop Symbol Timing

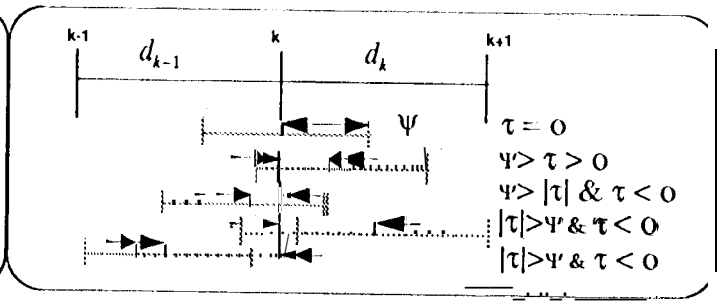

Figure 6. SS1. hlid-1'base Intergrator Timing

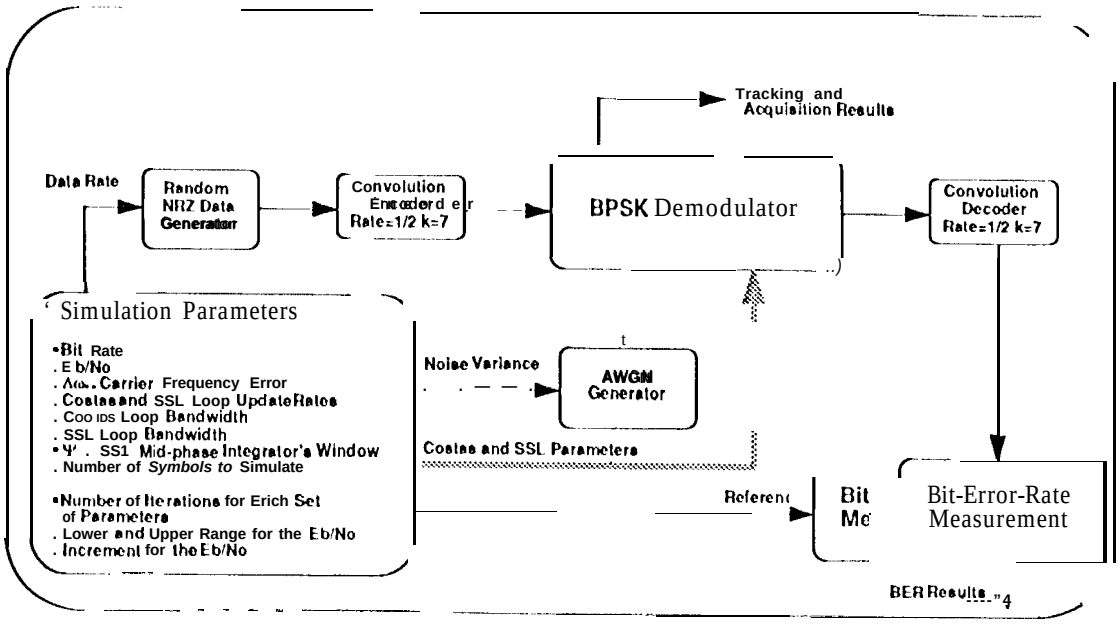

Figure 7. Performance Analysis Algorithm

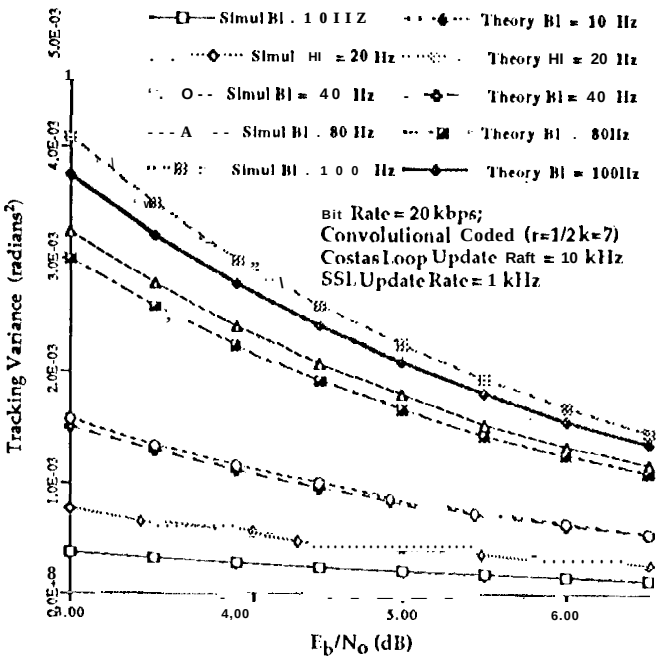

Figure 8. BPSK Costas Loop Tracking Variance

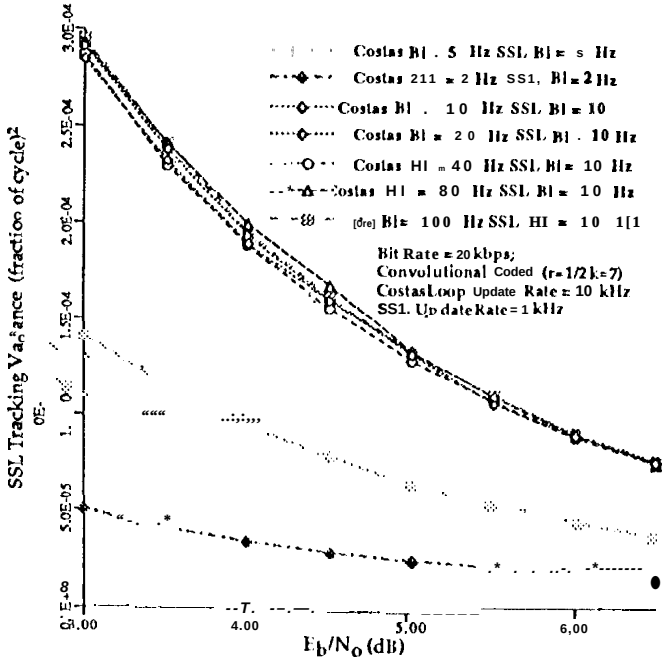

Figure 9. Symbol Synchronizer Loop Tracking Variance

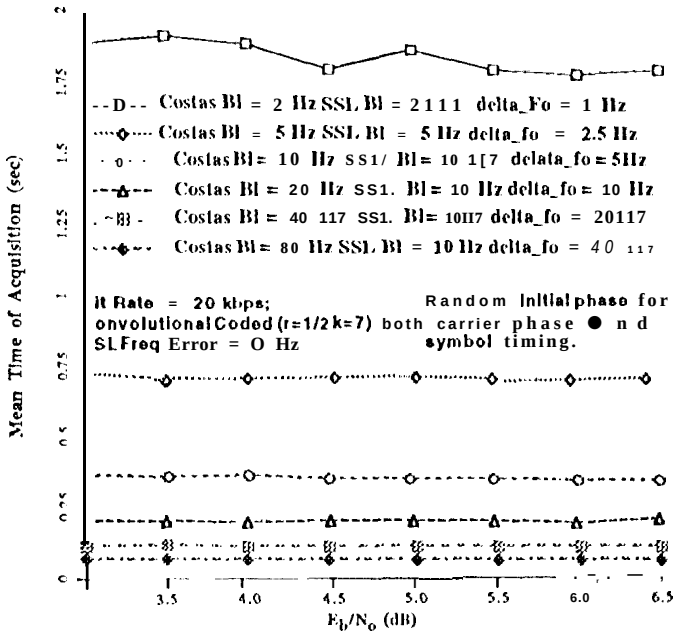

Figure 10. BPSK Costas Loop Mean Time of Acquisition Versus E_b/N_0

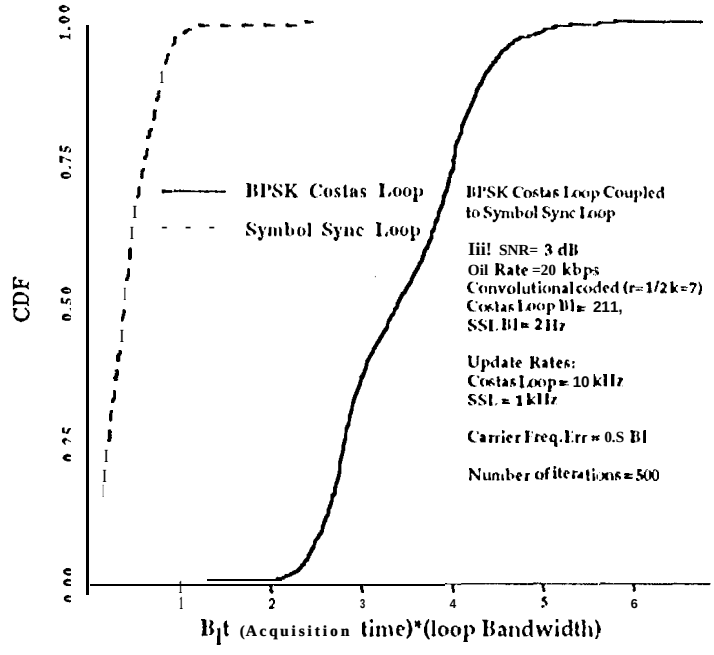

Figure 12. Cumulative Distribution Function of the Acquisition Time

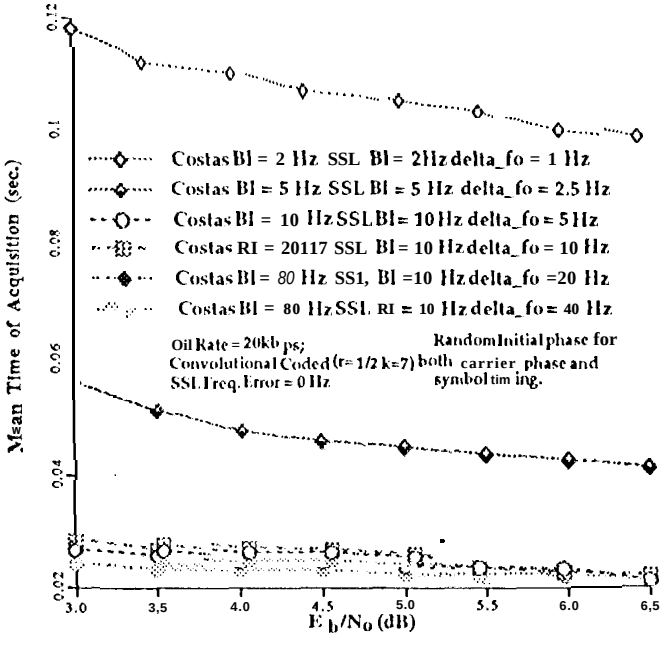

Figure 11. Symbol Synchronizer Loop Mean Time of Acquisition (see)

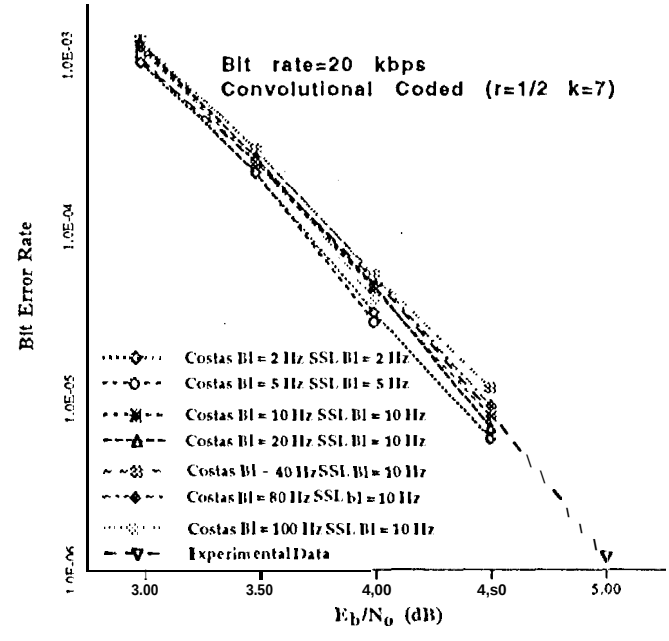

Figure 13. BPSK Demodulator Bit-Error-Rate Results