Galileo Ground Segment Reference Receiver Performance Characteristics

Neil Gerein
NovAtel Inc.
Calgary, Alberta, Canada
neil.gerein@novatel.ca

Co-Authors:
Allan Manz, NovAtel Inc., Canada
Michael Clayton, NovAtel Inc., Canada

1 Abstract

NovAtel, under contract to ESA, has initiated the development cycle for the high quality Ground Reference Receivers to be used in the future Galileo Sensor Stations (GSS). During the design process NovAtel is leveraging their experience as the world's leading supplier of Ground Reference Receivers to satellite augmentation systems in Europe, the USA, Australia, Japan and China.

The first step in this design process is the development of receiver requirements together with the confidence that these requirements can be met. The Binary Offset Carrier (BOC), multiplexed codes, multiple carrier frequencies, potential use of digital pulse blanking, and new high rate spreading codes make the design of a Galileo Reference Receiver challenging. To meet this challenge NovAtel is developing a bit level software simulation of a Ground Reference Receiver to verify performance characteristics during the requirements definition phase.

The critical performance characteristics of the Galileo Reference Receiver will be reviewed. An overview of the bit level software simulator will be presented. The expected tracking, multipath mitigation and interference rejection performance of the Galileo Reference Receiver will also be discussed.

2 Introduction

Approximately thirty Galileo Sensor Stations (GSS) will be distributed worldwide to provide measurements to the Galileo Control Centres (GCC). Each GSS will contain two to three reference receivers. The primary function of the receivers in a GSS is to consistently provide demodulated signal symbols and high precision pseudorange and carrier phase measurements. The ability to provide this information in less than ideal environments is also a requirement. In order to meet design assurance levels, many of the ancillary functions usually performed by a satellite based positioning receiver are eliminated, as they are not required for this application. The GSS receivers are optimized for fixed positions, continuous operation, and high quality reference oscillator inputs. Additionally, the network comprised of multiple receivers provides redundant information. This redundant information can be used to detect errors and improve performance with greater reliability and accuracy than is possible for a stand-alone receiver. Therefore, a receiver in a network can be more aggressive in collecting data, and thus provide more information, because of the additional safeguards provided though the network. Currently the development of the Galileo Reference Receiver (GRR) is in the requirements definition phase. NovAtel, under contract to ESA, is developing a high fidelity software simulator to be used to verify performance requirements during this phase. NovAtel is also developing a high-level architecture design for the GRR.

The high level conceptual design for the GRR is based on the NovAtel Common Reference Receiver (CRR) currently in development for WAAS, see Figure 1. A single RF/IF analog radio for every GNSS-2 frequency is implemented. In addition, the design optionally supports GNSS-1 frequencies. The tuned RF/IF radio supports the digitization of only the signals in the frequency band containing the desired transmitted signal. The digitized signals are then correlated by a number of parallel mechanisms as shown in Figure 2. Each mechanism is optimized to track one transmitted signal. The resultant correlation accumulations are used in code and carrier tracking control loops. The correlation accumulations are also used to extract the transmitted symbols. The state of the various tracking loops is periodically sampled at precise moments with respect to the time of the receiver. This information forms the basis of the pseudorange and carrier phase measurements that are output by the receiver. These measurements are accompanied by asynchronously gathered channel state information, such as channel tracking state, measured signal Doppler, estimated signal C/No, estimated carrier phase and pseudorange control loops errors, etc. The baseline system will track signals comprising the Open Service and Safety-of-Life Service. In total the receiver will have the ability to track 15 L1B (data) signals, 15 L1C (pilot) signals, 15 E5a-I (data) signals, 15 E5a-Q (pilot) signals, 15 E5b-I (data) signals, 15 E5b-Q (pilot) signals simultaneously. The receiver will also have the capability to support additional cards to track the L1A (PRS) and E6 signals.

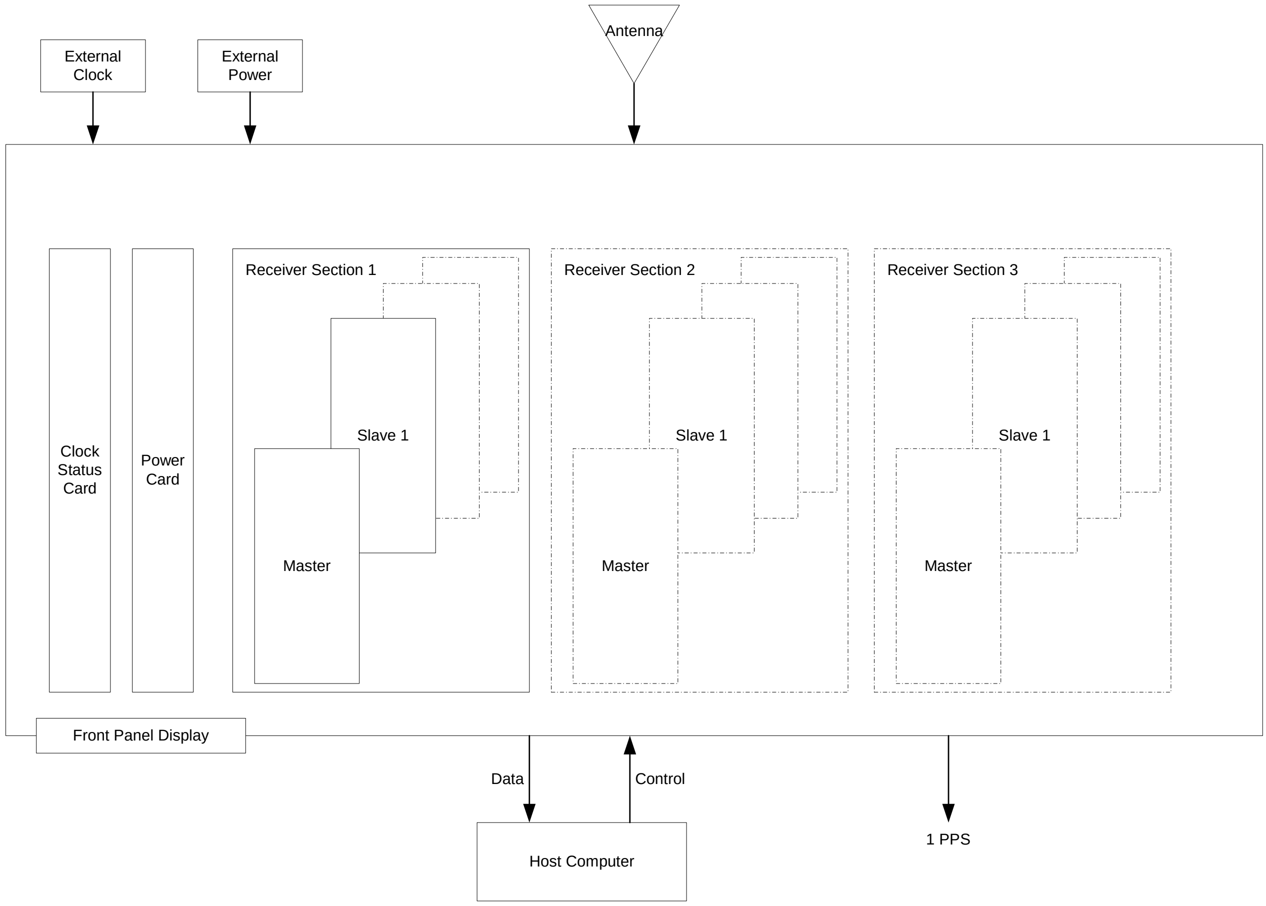

Figure 1 - GRR Functional Architecture

The decoding of the demodulated data and the use of this information in combination with the pseudorange measurements to compute a receiver Galileo time provides the means of computing unambiguous pseudorange measurements, generating a 1 PPS signal and facilitates the use of the receiver data at a network level.

The proposed receiver architecture provides for a flexible arrangement of hardware to accommodate future enhancements. This flexibility requires little or no changes in high level functional components, but provides the means of improving receiver performance such as: pseudorange and carrier phase measurement accuracy; multipath mitigation; signal distortion detection; and receiver throughput. For example, additional receiver cards can be added within the chassis to provide the extra correlators needed for the implementation of Signal Quality Monitoring (SQM) or Multipath Estimating Delay Lock Loop (MEDLL^{TM}). Multiple cards can share the same digitized data across the backplane, thus eliminating RF biases.

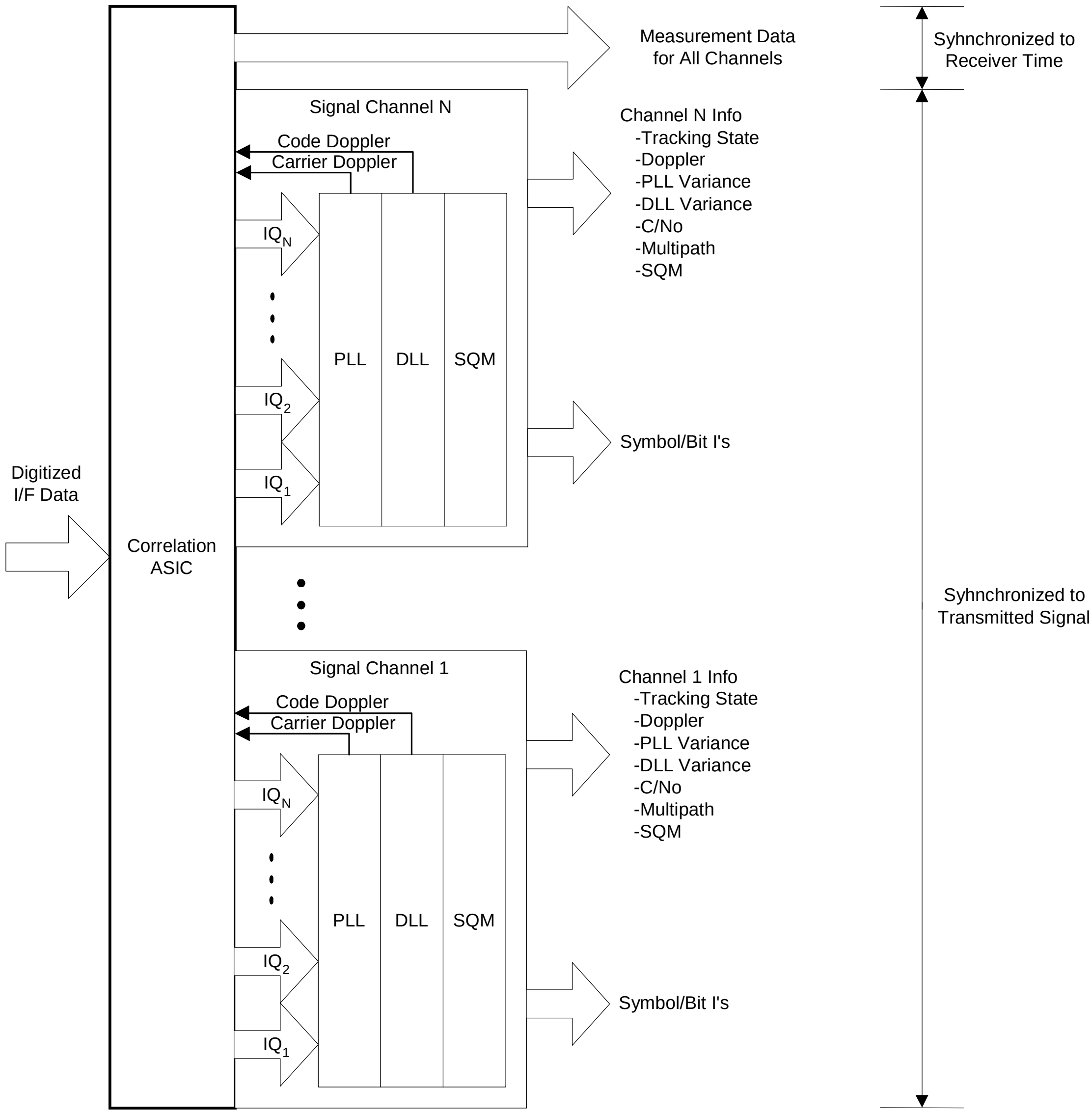

Figure 2 - Galileo Correlation / Signal Tracking Schematic

3 Critical Performances

A ground reference receiver has a very specific purpose – to act as a raw measurement engine to upstream processing. The purpose of the high fidelity software simulation is to verify that a ground reference receiver can meet the proposed requirements. This section reviews some of the GRR performance requirements that can be verified through simulation prior to receiver development.

3.1 Code and Carrier Tracking Error

Common measures of a GNSS receiver's performance are the 1 sigma code and carrier tracking noise and biases. The noise is dependent on tracking loop bandwidth, predetection integration time, discriminator spacing, front-end bandwidth, and other factors. The biases are dependent on signal corruptions such as multipath. Various numerical approximations exist for estimating both the code and carrier tracking noise and biases, and these estimations can be used to specify the receiver requirements.

High fidelity receiver simulation can be used to confirm code and carrier tracking requirements.

The proposed E5a/E5b pilot signal 1-sigma code phase observable noise requirements due to thermal noise and interference in nominal conditions are given in Table 1. The proposed L1C noise requirements are given in Table 2. The proposed carrier phase observable 1-sigma noise requirement is the same for all frequencies, and is 0.005 meters.

Table 1 - Proposed E5a/E5b (pilot) Code Tracking Noise Requirements

C/N_0 (dB-Hz)	38.5	42.2	43.5
1-sigma pseudorange noise (cm)	20	12	11

Table 2 - Proposed L1C (pilot) Code Tracking Noise Requirements

C/N_0 (dB-Hz)	37.5	41.2	42.5
1-sigma pseudorange noise (cm)	20	13	11

3.2 Signal Quality Monitoring (SQM)

The SQM function monitors GNSS signals in space for anomalous behavior by accurately measuring the demodulated correlation function of the SIS. The receiver outputs accumulations at the specified correlation function values. The accumulations can also be based on an early-late calculation. The receiver collects accurate accumulation values and outputs them in a timely fashion. The receiver hardware must be capable of accurately tracking the correlation function at multiple correlation locations. The processing of the accumulation values (smoothing, removing biases and finally calculating the metrics) may be done by the receiver, or by a higher level processing computer. Processing the data at a network level provides the opportunity to increase the accuracy and reliability of the computed metrics. The points at which correlation values will be determined are defined relative to the punctual correlator. An example of a set of SQM correlator positions is given in Figure 3.

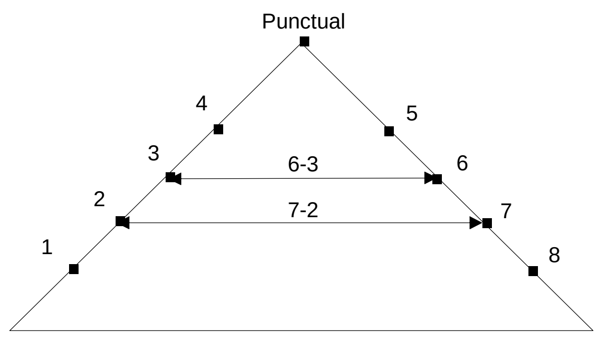

Figure 3 - Example Correlator Positions for SQM

Implementation of SQM requires cooperation between many different organizations and companies. Initially, a Failure Mode and Effect Analysis (FMEA) must be completed by the satellite manufacturer to identify possible threat models. At this point the receiver manufacturers and the various regulatory bodies governing safety of life operation must agree upon an acceptable range of discriminator functions, receiver radio bandwidths, and carrier phase smoothing or aiding time constants for both reference receivers and user receivers. Potential algorithms must be evaluated using the user receiver and reference receiver constraints against the various threat models. The results of the initial testing and evaluation might indicate that constraints must be re-evaluated and that the user and reference user constraints may differ. At the end of the process an algorithm and range of user and reference receiver constraints will be specified which bounds the acceptable differential pseudorange error between reference and user receivers.

The flexible architecture of the proposed GRR can accommodate all the currently known scenarios. Therefore, it is possible to delay finalizing the implementation details until later in the development schedule.

3.3 Multipath

Isolating a signal distortion due to errors at the satellite from multipath is exceedingly difficult at a local level. The effects of multipath at local sites can be mitigated at the network level if sufficient information is made available to the central processing facility. Nevertheless, in a ground reference receiver the local multipath effects can be mitigated through the use of techniques such as Narrow Correlator™ processing, and Multipath Estimating Delay Lock Loop (MEDLL™).

The proposed Galileo BOC(2,2) signals on L1 will be transmitted with the excess bandwidth required for Narrow Correlator™ processing. During the development of receiver requirements the effects of front-end filtering and correlator spacing will be studied by simulation. With a software simulator, new multipath models may be implemented and tested with less cost than implementing new models on a hardware simulator.

NovAtel has developed a multipath estimating technology known as MEDLL™. MEDLL™ uses a combination of hardware and software processing to directly measure the amplitude, phase and delay of each multipath component using maximum likelihood criteria[1]. Each estimated component is then subtracted from the measurement correlation function. MEDLL™ requires the receiver to sample the correlation function at multiple locations. Implementations of MEDLL™ using BOC(2,2) correlation functions can be simulated in software before hardware implementation.

The proposed E5a/E5b pilot signal peak code phase observable error requirements due to multipath in unfavourable conditions is given in Table 3. The proposed L1C error requirements in unfavourable conditions are given in Table 4.

Table 3 - Proposed E5a/E5b (pilot) Code Tracking Error Due To Multipath Requirements

C/N_0 (dB-Hz)	43.0	48.0
Carrier to Multipath Ratio (dB)	6.0	23.0
Peak pseudorange error (cm)	251	31

Table 4 - Proposed L1C (pilot) Code Tracking Error Due To Multipath Requirements

C/N_0 (dB-Hz)	43.0	48.0
Carrier to Multipath Ratio (dB)	6.0	23.0
Peak pseudorange error (cm)	166	23

3.4 Interference Mitigation

The effect of Radio Frequency Interference (RFI) is to reduce the C/N_0 level of the received signals. If the C/N_0 level drops below the tracking threshold a loss of lock will occur. Care can be taken with the design of the receiver tracking loops to reduce the effect of RFI.
The following generalizations can be made with regard to tracking loops:

- The pre-detection integration period can be as short as possible under high dynamic stress. However, because a ground reference receiver is stationary the pre-detection integration period can be increased to improve the tracking threshold for weak signals and during periods of RFI.
- A narrow bandwidth loop filter will filter out more noise (hence improve the RFI capability). A wide bandwidth loop filter settles faster but is only desirable under high dynamic stress.
- The loop order is sensitive to the same order of dynamics (i.e first order is sensitive to velocity stress, second order is sensitive to acceleration stress, third order is sensitive to jerk stress).

One of the key features of the proposed Galileo signal structure is the use of pilot signals (i.e. no data modulation). The GNSS-1 receiver designer has traditionally been limited to using Costas Loop PLL discriminators that are insensitive to 180-degree phase reversals due to data modulation. Since the pilot signals have no data, and therefore no 180-degree phase reversals, a true four-quadrant arctangent PLL discriminator can be used. This means the pre-detection integration period can be extended beyond the data period, improving the receiver's performance in the presence of RFI. The tracking error threshold of the true PLL (full 360-degrees) is double that of the Costas PLL, and therefore reduces the power needed for tracking by 6 dB. If the receiver is stationary and has a high quality clock, as is the case for a ground reference station receiver, then narrowing the PLL bandwidth is a viable solution for interference mitigation[2].

The software simulation described in this paper simulates the proposed spreading codes to be generated by the GNSS-2 satellites. This allows the designer to study the effects an interfering signal has on a specific spreading code spectrum.

The pulsed interference from Distance Measuring Equipment (DME)/Tactical Air Navigation (TACAN) in the E5a/L5 and E5b frequency bands is of concern. The use of digital pulse blanking has been shown to mitigate the effects of the pulsed interference from DME/TACAN sources[3]. Because digital pulse blanking operates on a sample-by-sample basis it is suitable to use high fidelity software simulation to study its performance in a ground reference receiver environment.

Definition of the expected interference environments for the GSS is ongoing. The preliminary proposed values for in-band interference are given in Table 5, and the proposed out-of-band interference is given in Table 6.

Table 5 - Proposed In-Band Interference Assumptions

Nominal in-band interference	-141.3 dB_W in any 1 MHz
Extreme in-band interference	-131.3 dB_W in any 1 MHz

Table 6 - Proposed Out-of-Band Interference Assumptions

Frequency (MHz)	**Total Interference/Minimum Desired Signal Power Ratio (I/S)**
f<1127.95	100 dB
1127.95<f<1164.45	100-2*(f-1127.95) dB
1188.45<f<1192.07	27 dB
1216.07<f<1237.41	27+2*(f-1216.07) dB
1237.41<f<1258.75	69.7-2*(f-1237.41) dB
1298.75<f<1335.25	27+2*(f-1298.75) dB
1335.25<f<1522.552	100 dB
1522.552<f<1559.052	100-2*(f-1522.522) dB
1591.788<f<1628.29	27+2*(f-1591.788) dB
f>1628.29	100 dB

4 Software Simulator

4.1 Simulator Overview

To aid in the development of the software simulator, NovAtel purchased the commercial MATLAB GPS Signal Simulation Toolbox from NAVSYS Corporation. The NAVSYS Toolbox is a collection of source code files that can be used to study the effects of the GPS C/A code satellites on a conventional GPS receiver. NovAtel is using the core building blocks of the NAVSYS Toolbox, along with building blocks modified for the Galileo signal structure, to develop simulations of the GRR. In this section we provide an overview of the software simulation.

The simulation consists of two main steps: 1) signal generation, and 2) tracking the received signal. Figure 4 is the high-level flow diagram of the signal generation step[4]. The user's initial position and time are used to determine the pseudorange to each satellite in view. To decrease the amount of time needed for simulation the user may select a subset of the visible satellites. The user specified spreading codes are generated and modulated with the navigation message and a carrier signal. Interfering signals can be added if desired. The composite signals are then passed through a receiver front-end software module, where the signal is filtered to a finite bandwidth and sampled. The output of the receiver front-end block is a vector of samples that a receiver ASIC would "see" at the output of an analogue-to-digital (A/D) converter. This vector of digital samples is saved in a Digital Signal Format (DSF) file for later input into the receiver simulation.

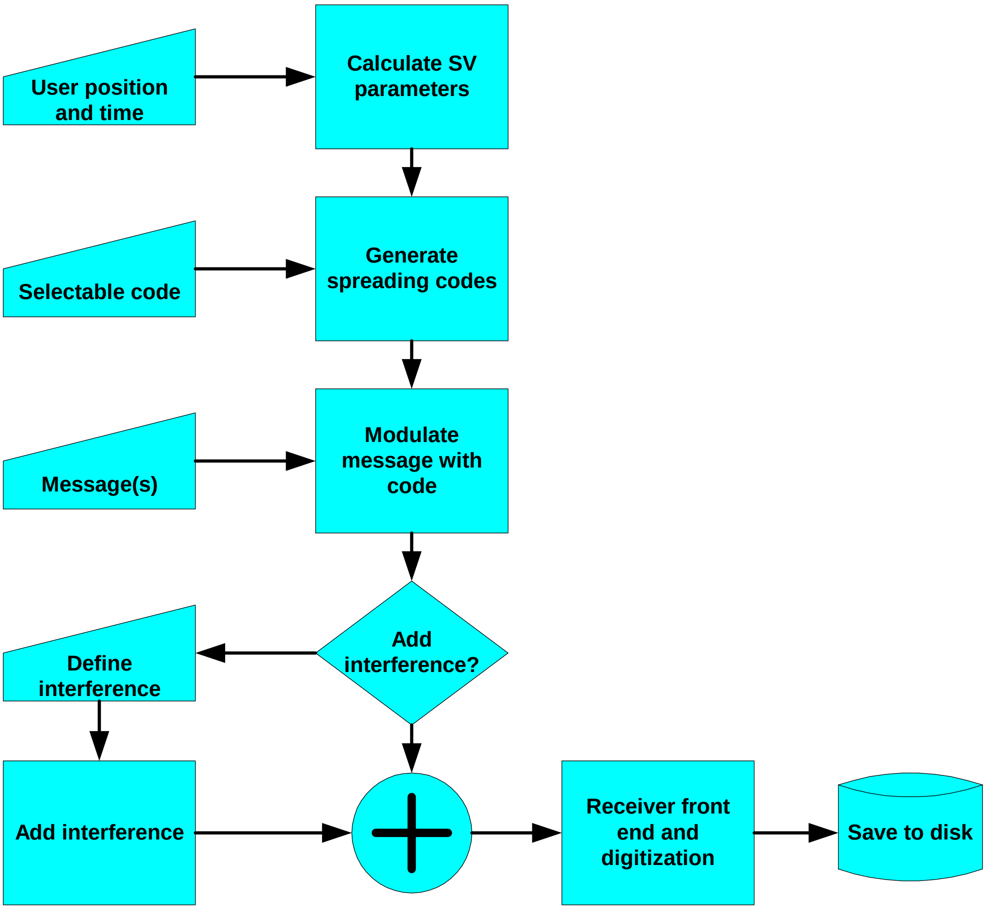

Figure 4 - Simulation Signal Generation Flow Diagram

The simulation signal generation shown in Figure 4 has a number of attractive features. The MATLAB programming language allows for new spreading codes and signal characteristics to be added with relative ease. Creating additional pseudoranges during the "Calculate SV parameters" step allows the simulation of multipath signals. Interference signals can also be defined. Simulating the filtering and sampling effects of the receiver front-end creates an accurate representation of the signal for baseband processing. Saving the digital samples to disk allows the user to compare different baseband processing configurations with the exact same set of A/D samples.

The receiver tracking flow diagram is shown in Figure 5. The receiver simulation consists of three major steps: 1) reading data from an existing DSF file, 2) processing the data through the receiver tracking loops, 3) update the tracking states. The latter two steps will now be described in detail.

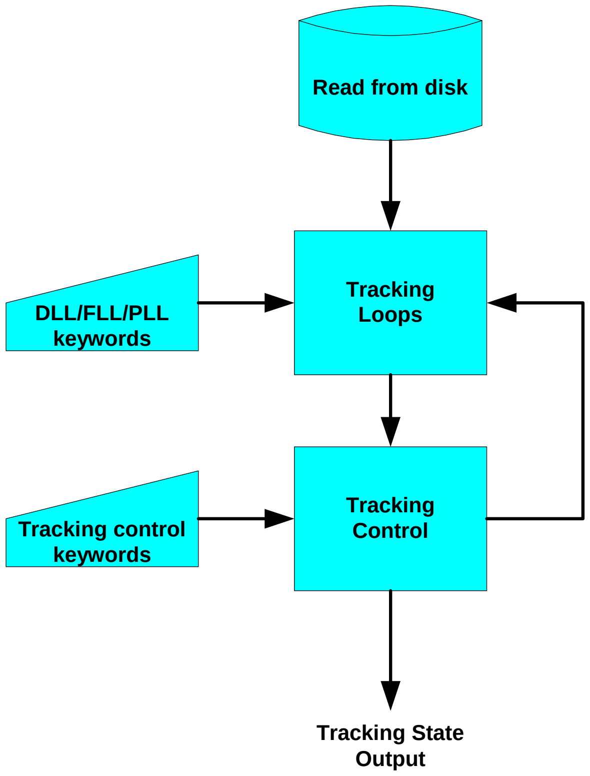

Figure 5 - Simulation Tracking Flow Diagram

The tracking loops consist of traditional delay lock loops (DLL), frequency lock loops (FLL), and phase lock loops (PLL). The MATLAB high level programming language offers considerable flexibility for development. For example, an almost unlimited number of correlators may be implemented, at relatively arbitrary locations along the correlation function.

The tracking control block shown in Figure 5 is used to transition between tracking states based on user-defined thresholds. The tracking state is defined as a 3 digit number, with the 100's place representing the carrier tracking state, the 10's place representing the code tracking state, and the 1's place representing the search state. A diagram illustrating the various tracking state transitions is shown in Figure 6. A typical test run starts with the receiver simulation in a wide search (state 001). After the user-defined acquisition declare threshold is reached the receiver starts the DLL and advances the code state. The carrier tracking loop is also started. The transitions between the different carrier tracking states is controlled through calculation of a locksum. The locksum returns a value between 0 and 1 that indicates the level of frequency and phase error in the tracking loop. As shown in Figure 6, the carrier loop first implements a wide FLL, then transitions to a narrow FLL/wide PLL, and finally to a narrow PLL. If the received signal contains navigation data then the carrier loop will attempt bit sync to transition to the final narrow PLL state.

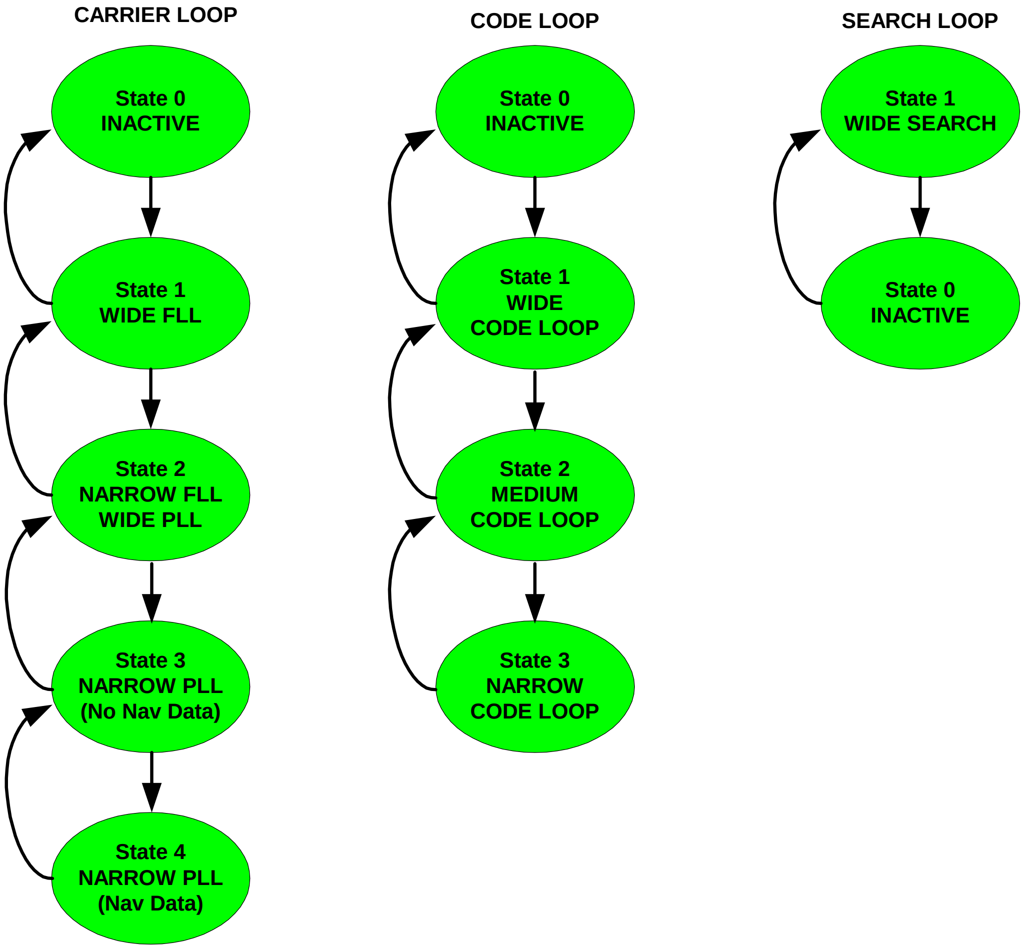

Figure 6 - Tracking States

User-defined keywords are used to control the signal generation and receiver simulation. Signal generation keywords include sample rate, front-end bandwidth, intermediate mixing frequency and noise figure. Receiver simulation keywords include bandwidths for code and carrier tracking loops, thresholds for advancing and reversing the code and carrier tracking states, and filter time constants used in the calculation of the carrier tracking loop locksum. Examples of user-defined keywords are given in Table 7.

Table 7 - Example Keywords

Keyword	Description
CODE_TYPE	Spreading code type, ex. CA, L1B, L1C, E5AI, E5AQ
FREQ_SAMPLE	Sampling rate of the A/D converter
IF_FREQ	Intermediate frequency of the RF front end
AD_BITS	Number of A/D converter bits to simulation
BW	Front end bandwidth
NOISE_FIGURE	Low noise amplifier noise figure
CORRELATOR_SPACING	Correlator spacing in chips
BDLL	DLL bandwidth
BFLL1	Wide FLL bandwidth
BFLL2	Narrow FLL bandwidth
BPLL	PLL bandwidth
INTEGRATION_TIME	Predetection integration time
T_LOCK_SUM	Carrier loop locksum threshold for PLL pull in
AFC_ALPHA_1	Carrier loop locksum time constant for wide FLL state
LOG_RATE	The rate at which simulation data is logged

The output from the receiver simulation is a tracking state output vector, with a frequency defined by LOG_RATE. The contents of the tracking state output vector are shown in Table 8.

Table 8 - Tracking State Output Vector

Time (sec)
Satellite PRN
Pseudorange (meters)
Sum of 1 millisecond I^2+Q^2
DLL RMS tracking error (chips)
Carrier phase (cycles)
Doppler frequency (Hz)
Frequency rate of change (Hz/msec)
Carrier phase RMS error (cycles)
Carrier PLL locksum
Carrier PLL time since last lost lock
C/N0 in dB-Hz
Current track state

4.2 Simulation Examples

In this section outlines two simulation examples. The high fidelity simulation allows the user to modify a number of receiver parameters and retest, without the expense of modifying analogue and digital hardware.

The first examples simulate the effect of the front-end bandwidth on the receiver code tracking noise. Algebraic approximations are provided by Betz[5] to estimate the expected noise as a function of front-end bandwidth and early-minus-late discriminator spacing. Betz identifies three cases: 1) Spacing limited, where the noise depends primarily on the early-late spacing and not the front-end bandwidth, 2) Bandwidth limited, where the noise depends primarily on the front-end bandwidth and not the early-late spacing, and 3) A transition region between the other two cases. These cases are shown below in equations 1 to 3 respectively, where *D* is the normalized early-late spacing, *b* is the normalized front end bandwidth, *T* is the pre-detection integration time, B_L is the DLL bandwidth, and C/N_0 is the carrier to noise ratio.

$$\left(\frac{\sigma_\tau^2}{T_c^2}\right)_{NELP} \cong \frac{B_L(1-0.5B_L T)}{2\frac{C}{N_0}} D \left[1+\frac{2}{T\frac{C}{N_0}(2-D)}\right] \quad \pi \le Db \tag{1}$$

$$\left(\frac{\sigma_\tau^2}{T_c^2}\right)_{NELP} \cong \frac{B_L(1-0.5B_L T)}{2\frac{C}{N_0}} \left(\frac{1}{b}\right) \left[1+\frac{1}{T\frac{C}{N_0}}\right] \quad Db \le 1 \tag{2}$$

$$\left(\frac{\sigma_\tau^2}{T_c^2}\right)_{NELP} \cong \frac{B_L(1-0.5B_L T)}{2\frac{C}{N_0}} \left[\frac{1}{b}+\frac{b}{\pi-1}\left(D-\frac{1}{b}\right)^2\right] \left[1+\frac{2}{T\frac{C}{N_0}(2-D)}\right] \quad 1 < Db < \pi \tag{3}$$

Figure 7 shows the spacing limited case described by equation 1. For this example the normalized bandwidth is set to 10 and the early-late discriminator spacing is set to 1 chip.

Figure 8 shows the bandwidth limited case described by equation 2. For this example the normalized bandwidth is set to 2 and the early-late discriminator spacing is set to 0.2 chips. For reference the spacing limited case described by equation 1 is shown as the solid line.

Figure 9 shows the transition region case described by equation 3. For this example the normalized bandwidth is set to 2 and the early-late discriminator spacing is set to 1 chip. For reference the spacing limited case described by equation 1 is shown as the solid line.

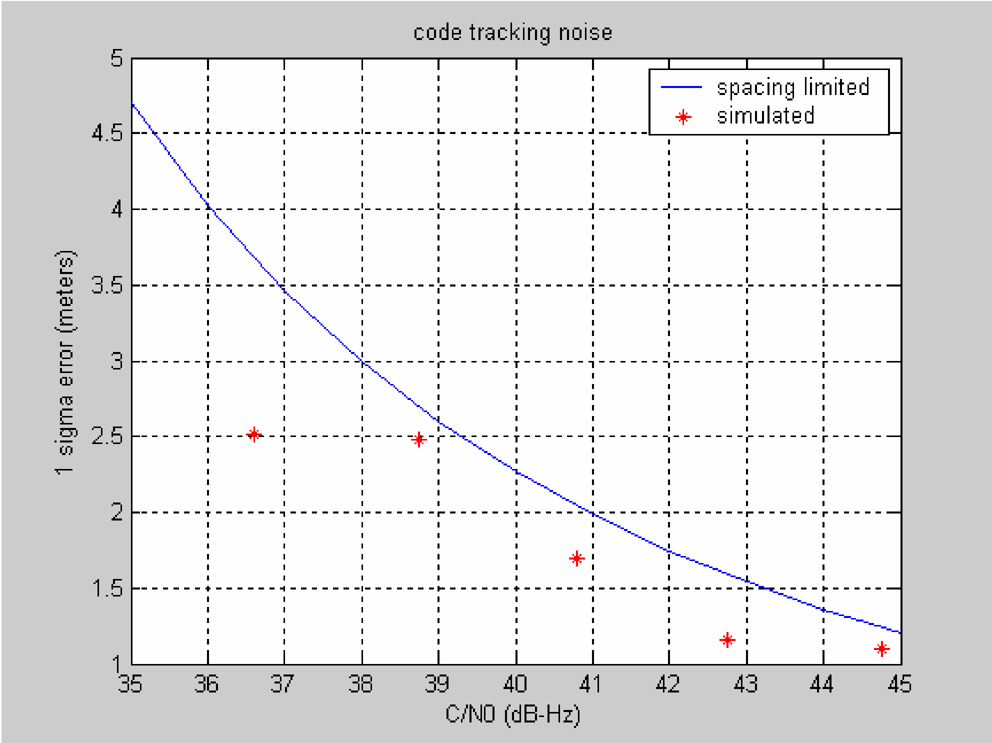

Figure 7 - Spacing Limited Tracking

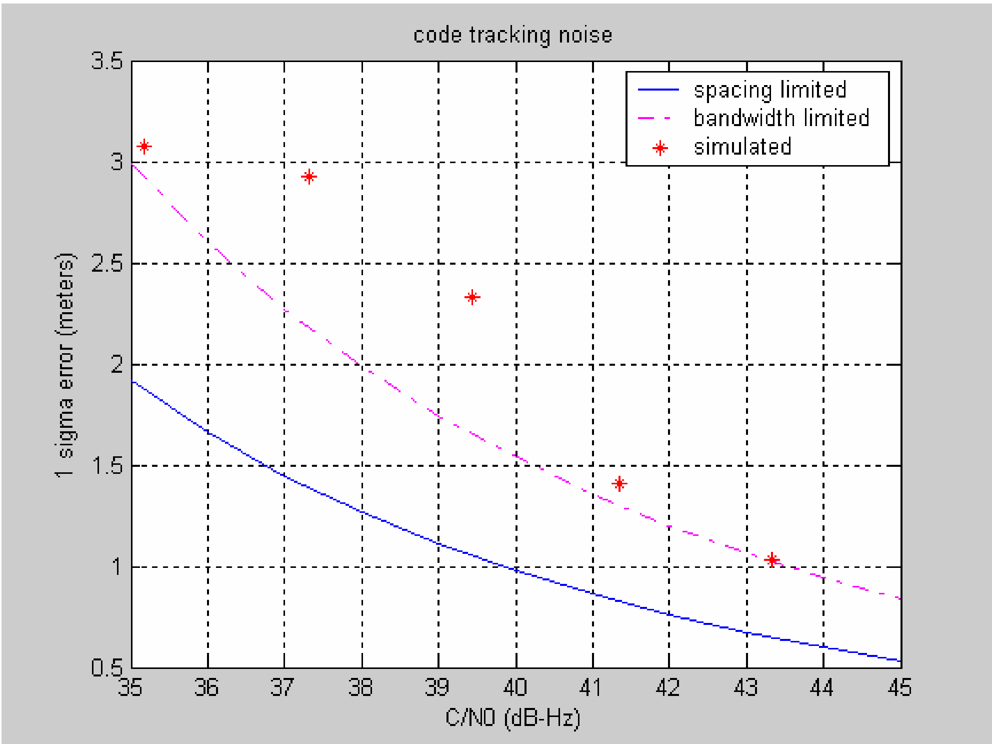

Figure 8 - Bandwidth Limited Tracking

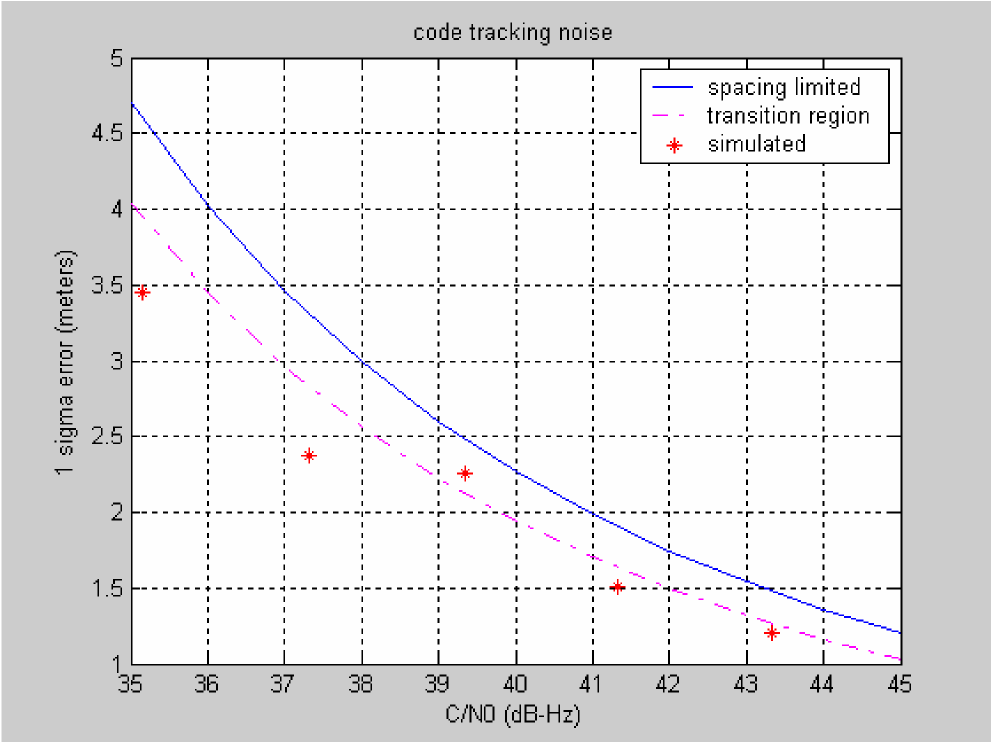

Figure 9 - Transition Region Tracking

The second example concerns the use of new spreading codes. The testing of new codes using hardware correlators involves implementation on a field programmable gate array (FPGA) or a custom made Application Specific Integrated Circuit (ASIC). With software simulation the receiver designer can gain insight and design experience with new GNSS-2 signals without the expense of hardware design, implementation, and debugging. This is shown in Figure 10 and Figure 11 below. For these figures the solid line is the expected code tracking noise calculated using equation 1, and the solid dots are the 1 sigma code tracking noise output from the receiver simulation. The spreading code has a chipping rate of 2.046 MHz, and a code length of 8184 chips (code period of 4 milliseconds) for the results presented in Figure 10. The same spreading code was modulated with a 2.046 MHz square wave to generate the BOC(2,2) code and used to create the results in Figure 11. In both figures the receiver front-end bandwidth was set to 20 MHz, the early-minus-late correlator spacing was set to 0.4 chips, the DLL bandwidth was set to 1 Hz, and the pre-detection integration time was set to 1 millisecond. The BOC(2,2) signal in Figure 11 has improved tracking performance when compared with the BPSK(2) signal in Figure 10, due to the “sharper” autocorrelation function of BOC(2,2).

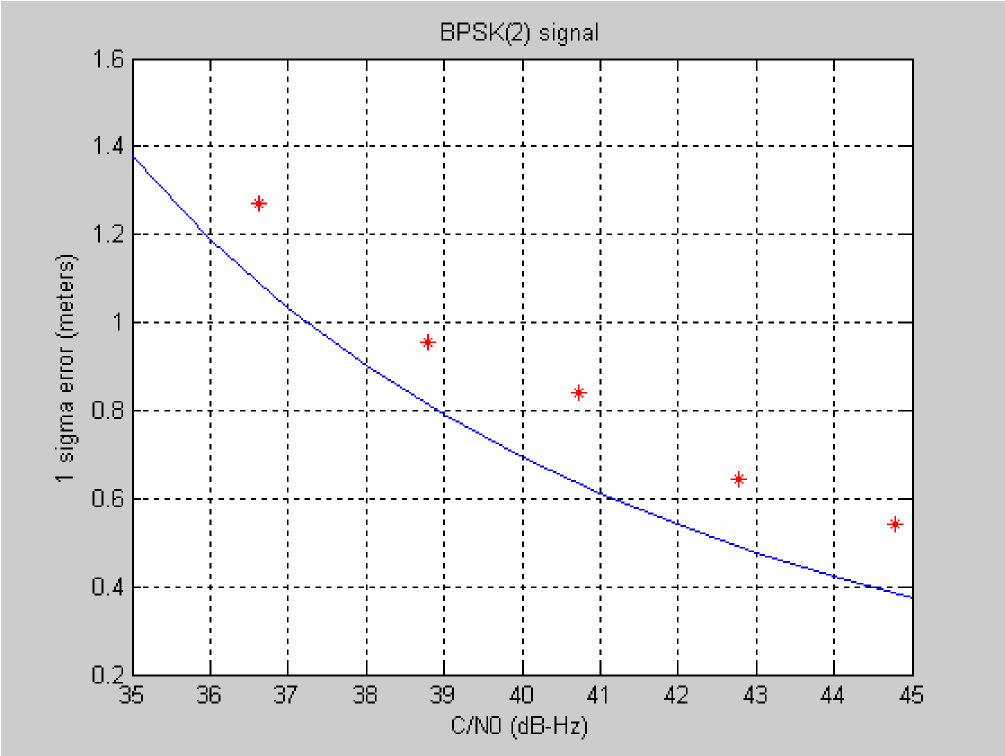

Figure 10 - Code Tracking Error (1 sigma) for BPSK 2.046 MHz chipping rate

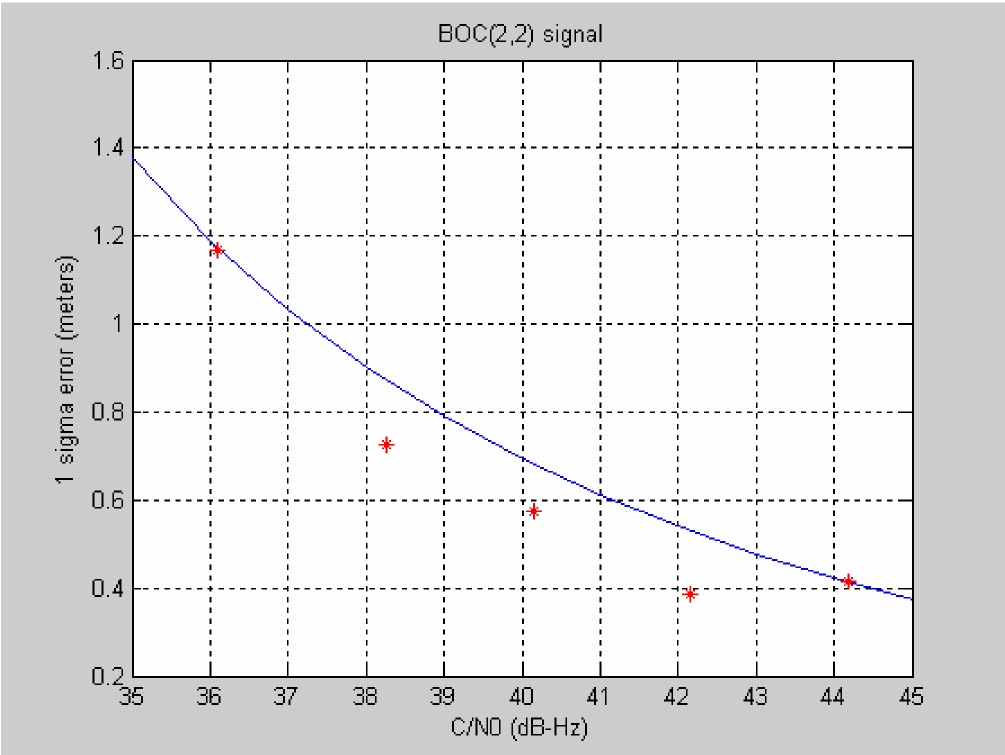

Figure 11 - Code Tracking Error (1 sigma) for BOC(2,2)

5 Concluding Remarks and Future Work

The powerful signal generation capabilities of the simulator can accommodate changes in the Galileo signal structure, new classes of interference, and updated multipath models. The simulation of the receiver design allows for testing of tracking loop changes between simulation runs while using the same set of A/D samples for each test.

Work is continuing on the high fidelity software simulator development, with testing of all critical performance requirements scheduled in the upcoming months.

6 References

[1] B. Townsend, P. Fenton, K. Van Dierendonck, D.J.R van Nee, "L1 Carrier Phase Multipath Error Reduction Using MEDLL Technology," *Proceeding of ION GPS-95,* Palm Springs, CA, September 1995

[2] A. Manz, et. al., "Improving WAAS Receiver Radio Frequency Interference Rejection", *Proceedings of the ION GPS-2000,* Salt Lake City, Utah, September 2000

[3] J. Grabowski, C. Hegarty, "Characterization of L5 Receiver Performance Using Digital Pulse Blanking", *Proceedings of the ION GPS-2002,* Portland, OR, September 2002

[4] A. Brown, N. Gerein, K. Taylor, "Modeling and Simulation of GPS Using Software Signal Generation and Digital Signal Reconstruction," *Proceedings of ION Technical Meeting*, Anaheim, CA, January 2000.

[5] J.W. Betz, K.R. Kolodziejski, "Extended Theory of Early-Late Code Tracking for a Bandlimited GPS Receiver", *Navigation: Journal of the Institute of Navigation,* Vol. 47, No. 3, Fall 2000