Real-Time Wideband Telemetry Receiver Architecture and Performance

K. Andrews,[1] J. Gin,[1] N. Lay,[1] K. Quirk,[2] and M. Srinivasan[1]

We describe the architecture and algorithm development for a field programmable gate array (FPGA) wideband telemetry receiver prototype capable of processing data rates in excess of 100 megabits per second (Mbps). The high-speed parallel implementations of the matched filter, carrier phase tracking loop, and symbol timing recovery loop are discussed, along with simulation and hardware performance results.

I. Introduction

The current operational telemetry receiver system used by the Deep Space Network (DSN) was developed and optimized for the demodulation, tracking, and processing of low to medium data-rate deep-space signals. Its design supports variable data rates between 10 symbols per second and 26 megasymbols per second and incorporates a number of additional user-definable settings such as tracking loop bandwidths. Currently supported signaling formats include residual carrier and subcarrier-modulated telemetry and constellation orders up to 2 bits per symbol. In the past, there have not been significant efforts to incorporate operational support for a wide range of bandwidth-efficient modulations into the DSN. However, with the potential for migration of telemetry services to 31.8 to 32.3 GHz (Ka-band) downlink, both higher data rates and bandwidth-efficient signaling methods are undergoing renewed emphasis. These two elements, in part, form a key basis of current developments in next-generation DSN ground receiver technology. The receiver design will employ a flexible and upgradeable architecture as well as telemetry-processing algorithms that are capable of supporting downlink data rates in excess of 150 megabits per second (Mbps). In addition to the high-rate telemetry aspect, the receiver also targets the integration of preprocessing to support wideband radiometric signal measurement. Its architecture will rely upon reprogrammable hardware in order to implement a wide variety of telemetry, tracking, and radiometric science signal processing. The focus of this article is the initial development of wideband telemetry-processing designs.

---

[1] Communications Architectures and Research Section.

[2] Autonomy Control Section.

The research described in this publication was carried out by the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration.

In order to develop pragmatic, functional performance objectives for the receiver, a review of relevant requirements[3,4] and strategic telecommunications system studies[5,6] was performed. This survey revealed a number of driving requirements as well as worthy technical objectives that could help enable new DSN capabilities. In the telemetry processing arena, documented requirements call for support of single-carrier data rates up to 150 Mbps and processing of Consultative Committee for Space Data Systems (CCSDS)-defined standards for bandwidth-efficient modulations. In comparison, system studies examining the ability of DSN apertures to support high-rate links from lunar distances under modest transmit terminal assumptions assessed closeable links at data rates well in excess of 1 gigabit per second (Gbps). These bit rates would then directly correspond to symbol rates if one considers that the lowest code rate and constellation size respectively are 1/2 and two. Consequently, these are utilized as initial threshold and objective specifications for the receiver development effort, with a greater emphasis placed upon the lower, requirements-based target.[7] Additional telemetry-related specifications will be derived in the future from level-2 system requirements and analyses of achievable performance. Examples of these include loss allocations, co-channel and adjacent channel environments, and waveform distortion limits. In addition to telemetry-motivated requirements, processing of radiometric signals will impose its own set of constraints that include pre-compensating wideband signals based upon Doppler predicts, minimizing signal distortion, and incorporating time-tagging functions generally unavailable in commercial high-rate receivers. As a result, we are able to extend the receiver's applicability to support measurement of science observations such as signals from ground-based planetary radar. Furthermore, incorporation early in the design phase provides the desired functionality without requiring significant additional resources.

Another key goal of this development effort is achieving efficient technology infusion. This is partially accomplished with early identification and accommodation of key interfaces that map to existing and future telemetry, tracking, and science signal-processing systems. Baseline assumptions regarding the wideband receiver input interface assume two main possibilities based upon the sampling and front-end signal-processing architecture of the Large Array Prototype a full-band sampled input from a down-converted, 500-MHz-wide intermediate frequency (IF) or a lower sample rate, complex baseband, and a digital signal reconstructed from integer multiples of 1-MHz subbands [1]. Assumptions regarding output interfaces include several types: (1) hard- or soft-decision bit likelihoods passed to a forward error correcting (FEC) decoder, (2) digitally sampled data streams consistent with input interfaces to existing or independently developed systems (e.g., a 160-MHz interface to the Downlink Telemetry and Tracking (DTT) Subsystem[8] or a 1000-MHz interface to the Goldstone Solar System Radar digital receiver[9]), and (3) a reconstructed analog IF interface to support commercial off-the-shelf (COTS) receiver equipment in the event that there are operational situations for which this configuration provides a cost-effective or low-risk solution.

---

[3] *Deep Space Mission System, Level-2 Requirements*, document no. 820-001 (internal document), baseline 1.15, Jet Propulsion Laboratory, Pasadena, California, July 28, 2005.

[4] *Microwave Array Project, Level II Requirements for the Deep Space Network Array*, IND no. 900-004 (internal document), rev. B, Jet Propulsion Laboratory, Pasadena, California, December 31, 2004.

[5] L. Deutch, R. Preston, and B. Geldzahler, "A Vision for the Next Generation Deep Space Network," viewgraph presentation (internal document), Jet Propulsion Laboratory, Pasadena, California, December 2004.

[6] L. Deutch, R. Cesarone, G. Noreen, T. Ely, R. Hastrup, D. Morabito, D. Abraham, S. Weinstein, M. Sue, and F. Manshadi, "Lunar Network," viewgraph presentation (internal document), Jet Propulsion Laboratory, Pasadena, California, February 19 2004.

[7] The objective 1-GHz symbol rate is likely to be accommodated only in the 26-GHz band allocation, which has 1500 MHz of available spectrum versus the 500 MHz available in the deep-space 32-GHz band.

[8] *Downlink Tracking and Telemetry Subsystem (DTT), Requirements and Design Document: Volume 1*, DSMS no. 834-069 (internal document), rev. A, vol. 1, Jet Propulsion Laboratory, Pasadena, California, August 27, 2002.

[9] M. Slade, "GSSR R&TD FY05 Annual Report," JPL internal document, Jet Propulsion Laboratory, Pasadena, California, 2005.

II. Wideband Telemetry Receiver Architecture

Figure 1 depicts the top-level functional block diagram of the wideband telemetry receiver. The receiver must perform the functions of analog-to-digital signal conversion, frequency downconversion to baseband, carrier phase recovery, symbol timing recovery, and derivation of soft symbols. Note that a wideband channelizer may be present that extracts signals from various application subbands to the receiver. Furthermore, because the baseband algorithms are performed with a fixed number of samples per symbol, it is necessary to perform sample-rate conversion on the sampled signals in order to accommodate continuously variable telemetry data rates over an extremely wide range.

As previously noted, the front-end architecture is assumed to be consistent with the analog IFs and sample rates given in [1]. For a full-band sampled input, the input sampling rate is 1280 MHz. With a wideband receiver design supporting a normalized time resolution of 8 samples per symbol, this input rate corresponds to a maximum telemetry symbol rate of 160 Msymbols per second. Separate numerically controlled oscillators (NCOs) will be required to independently downconvert multiple carriers present on the same IF.

A key design component of the proposed receiver architecture is modularity. We would like each baseband application subsystem to operate as designed without hardware modification; hence, the processing rate for a particular block is fixed. This requires the sample rate of the digitized input signal to be adjusted according to the specified number of input samples per processing interval; e.g., telemetry-processing algorithms may be designed for 8 samples per symbol. With downlink telemetry rates varying continuously from 10 bps to greater than 100 Mbps, and an input sample rate of 1.28 gigasamples per second, the sample-rate downconversion factor varies from 1.6 to 16,000,000. The design and analysis of the sample-rate conversion subsystem will be documented in a separate article. For the purposes of this article, we assume that this step is accomplished without significant loss.

A large body of work exists on the development and analysis of various deep-space telemetry-processing algorithms [2–4], which shall not be duplicated here. We shall restrict ourselves to a brief discussion of the signal format used in the prototype receiver development as well as a summary of algorithm choices. The input to the analog-to-digital converter (ADC) is a suppressed-carrier quadrature phase-shift keying (QPSK) signal, given by

$$r(t) = \sqrt{2P}\left(d_I\left(t-\tau(t)\right)\cos\left(\omega_{\mathrm{IF}}t+\theta(t)\right) + d_Q\left(t-\tau(t)\right)\sin\left(\omega_{\mathrm{IF}}t+\theta(t)\right)\right) + n(t) \tag{1}$$

where P is the average signal power, ω_{IF} is the IF frequency, $\theta(t)$ is the received carrier phase, $\tau(t)$ is the received symbol timing offset, and $d_I(t)$ and $d_Q(t)$ are the QPSK in-phase (I) and quadrature (Q) data modulation waveforms given by

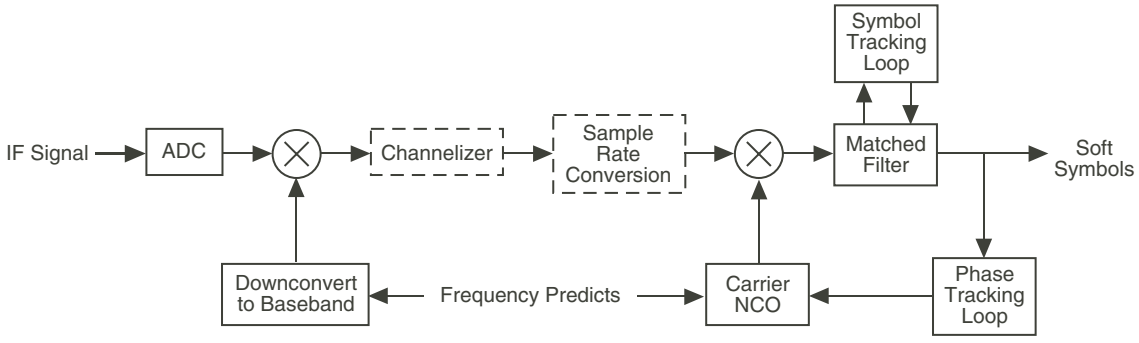

Fig. 1. Functional block diagram.

$$d_I(t) = \sum_{l=-\infty}^{\infty} d_l^{(I)} p(t - lT)$$
$$d_Q(t) = \sum_{l=-\infty}^{\infty} d_l^{(Q)} p(t - lT) \tag{2}$$

The in-phase and quadrature data bits $d_l^{(I)}$ and $d_l^{(Q)}$ take on values ± 1 with equal probability; T is the symbol duration; and $p(t)$ is a square-root raised-cosine (SRRC)-shaped Nyquist pulse with excess bandwidth α:

$$p(t) = \frac{4\alpha}{\pi\sqrt{T}} \frac{\cos\dfrac{(1+\alpha)\pi t}{T} + \dfrac{T}{4\alpha t}\sin\dfrac{(1-\alpha)\pi t}{T}}{1 - \left(\dfrac{4\alpha t}{T}\right)^2} \tag{3}$$

This pulse shape is shown in Fig. 2 for $\alpha = 0.35$. The noise process $n(t)$ is bandpass white Gaussian noise with one-sided power spectral density N_0. After sampling at frequency F_s, the digitized IF signal is mixed in quadrature to convert it to in-phase and quadrature baseband samples, given by

$$I(k) = \sqrt{2P} d_I\big(kT_s - \tau(kT_s)\big) \sin\phi(kT_s) + n_I(kT_s) \tag{4}$$

and

$$Q(k) = \sqrt{2P} d_Q\big(kT_s - \tau(kT_s)\big) \cos\phi(kT_s) + n_Q(kT_s) \tag{5}$$

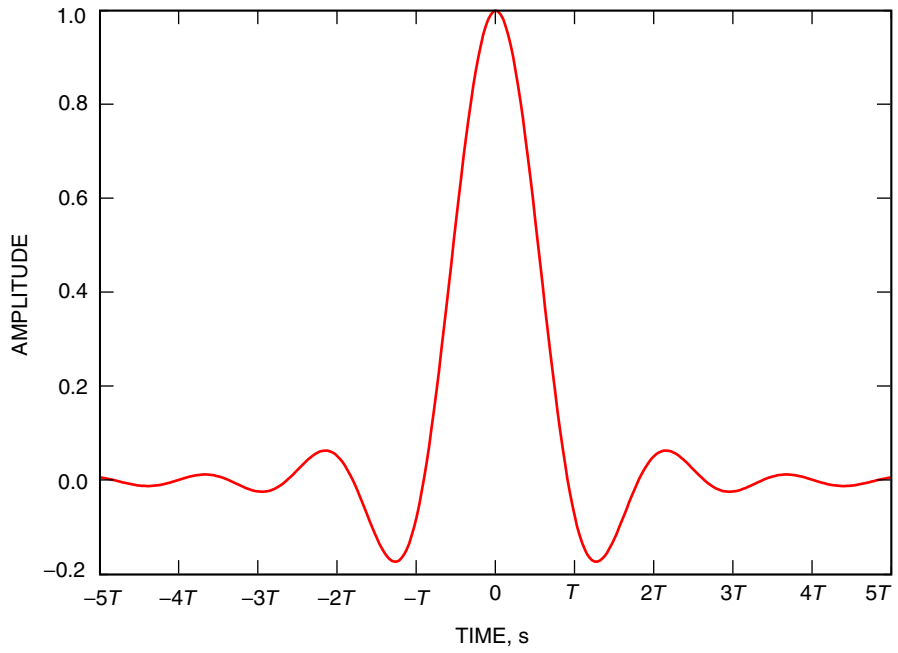

Fig. 2. Square-root raised-cosine pulse with $\alpha = 0.35$.

where $T_s = 1/F_s$ is the sampling interval and $\phi(kT_s)$ is the difference between the received carrier phase and the phase of the NCO. The quantities $n_I(kT_s)$ and $n_Q(kT_s)$ are independent zero-mean Gaussian variables with variance N_0F_s. Each of the baseband quadrature-sampled signals then is match-filtered with SRRC filters. A non-coherent symbol-timing recovery loop estimates τ and determines which match-filtered output sample to use as the soft-symbol output for each quadrature branch. The soft symbols are sent on to an error-correcting decoder as well as to the symbol-timing recovery loop and the carrier phase-tracking loop in order to form error signals for the timing and phase adjustments, to compensate for τ and ϕ, respectively.

III. Wideband Parallel Implementation

In order to process wideband telemetry signals with low-cost, low-power commercial hardware, the receiver is implemented in a parallel architecture as shown in Fig. 3. In this manner, only a single high-speed ADC along with a small number of other high-speed components are needed, while the bulk of the processing takes place at a lower rate. In the breadboard receiver, the ADC runs at 1.28 GHz; with a factor of 16 parallelization, the remainder of the hardware runs at 80 MHz.

Taking the samples from the ADC in a parallel fashion allows the field programmable gate array (FPGA) to run at a much lower clock rate as compared with the original sampling rate. The 16 samples from the ADC are taken into the FPGA and downconverted with a parallel downconvert, with cosine and sine values provided by a parallel NCO, to produce 16 parallel I and Q baseband samples. The baseband samples are processed by 33-tap parallel-root raised-cosine-matched filters, one for each arm. Following matched filtering, the samples are employed within a non-coherent symbol-tracking loop (Gardner symbol synchronizer), allowing the receiver to obtain symbol synchronization. The symbol synchronizer allows the receiver to determine which samples are designated as the soft decisions. Due to the parallel nature of the receiver, two decisions are obtained every clock cycle since the receiver is operating under

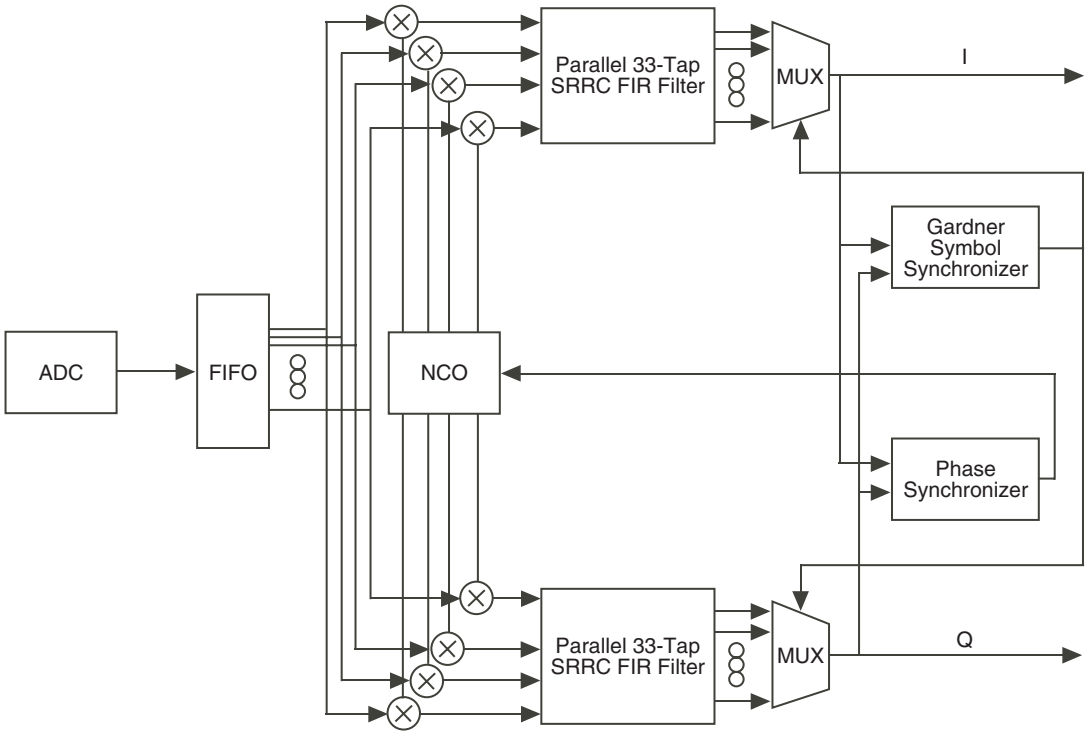

Fig. 3. Parallel telemetry receiver architecture.

the assumption of 8 samples per symbol.The two decisions result in two error signals that are averaged before being fed into the loop filter. The same two decisions are entered into a polarity-type Costas loop, which allows for the receiver to track out phase variations. Once again, the two decisions produce two error signals that are averaged, resulting in an error signal that is fed into the loop filter. The phase estimate is calculated and sent back to the parallel NCO, closing the loop, and producing new cosine and sine values that are multiplied by the input samples.

A. NCO Design

A numerically controlled oscillator (NCO) generates quantized samples of a sinusoid of a given frequency. The design is constrained by the spacing of the frequencies that can be generated and the fidelity of the resultant sampled sequence. A typical NCO design consists of an accumulator and a look-up table. At each clock, an offset, equal to the product of the desired frequency and the sample time, is added to the accumulator, producing the current phase of the sinusoid. This phase is used as the address into a look-up table containing quantized samples covering one cycle of the sinusoid. By increasing the offset that is accumulated, one steps through a cycle of the sinusoid in fewer samples, producing a higher-frequency output sequence; conversely, by decreasing the offset, more samples are required to complete a cycle, producing a lower-frequency output sequence.

The smallest realizable frequency, and thus the frequency resolution of the NCO, is determined by the smallest representable offset in the accumulator. For a hardware architecture where M samples are operated on in parallel, the frequency resolution is

$$f_{\min} = M \cdot 2^{-Q_p} f_s \quad \text{Hz}$$

where Q_p is the number of bits in the accumulator and f_s is the sample rate. The frequency resolution for accumulator sizes from 32 to 64 bits (4 to 8 bytes) were calculated for the current advanced receiver design, assuming a 1.3-GHz sample rate and $M = 16$ samples per clock (Table 1). A bit width of 40 was selected as providing adequate frequency and phase resolution.

The fidelity of the NCO output is set by the length and width of the look-up table, which determine the granularity and quantization of the output sinusoid samples, respectively. An appropriate look-up table width can be selected by calculating the level of quantization noise at the output of the NCO and ADC sample-mixing operation. Given the number of bits in an ADC sample, the NCO output quantization is selected such that the product of the ADC sample and NCO output has a quantization noise less than that of the ADC sample in order not to reduce the number of effective bits in the ADC. For an ADC with Q_a bits of resolution and an unsigned number representation ranging from 0 to $1 - 2^{-Q_a}$, the quantization noise is modeled as being independent of the unquantized value and uniformly distributed in $(-2^{-Q_a}, 0]$. Prior to mixing with the NCO output, the unsigned ADC values are converted to 2's complement numbers through an operation mathematically equivalent to

Table 1. Frequency resolution versus accumulator size.

Q_p	$f_{\min}$, Hz
32	4.842
40	0.019
48	7.389×10^{-5}
56	2.886×10^{-7}
64	1.128×10^{-9}

$$f(x) = 2\left(x - \frac{1}{2}\right)$$

The sample values now range from -1 to $1 - 2^{-(Q_a-1)}$ with an independent quantization noise uniformly distributed in $(-2^{-(Q_a-1)}, 0]$. Assuming the same quantization model for the NCO look-up table with Q_n bits, the quantization noise is uniformly distributed between $(-2^{-(Q_n-1)}, 0]$ and is independent from both the unquantized NCO value and the ADC quantization noise. This assumption results in a worst-case quantization condition. Both the ADC sample, in the absence of all other noise, and the NCO output can be represented as sinusoids with an independent additive quantization noise. Their product is given by

$$\mathbf{z} = \big(\sin(\theta_a) + \mathbf{q}_a\big)\big(\sin(\theta_n) + \mathbf{q}_n\big)$$

where θ_n and θ_a are independent and uniformly distributed in $[0, 2\pi)$, and the quantization noise of the ADC and NCO, $\mathbf{q}_n$ and $\mathbf{q}_a$, are independent and uniformly distributed in $(-2^{-(Q_a-1)}, 0]$ and $(-2^{-(Q_n-1)}, 0]$, respectively. The resulting quantization noise,

$$\mathbf{q}_z = \sin(\theta_a)\mathbf{q}_n + \sin(\theta_n)\mathbf{q}_a + \mathbf{q}_a\mathbf{q}_n$$

has a variance

$$\sigma_{q_z}^2 = \frac{2}{3}\left(2^{-2Q_a} + 2^{-2Q_n}\right) + \frac{7}{9}2^{-2(Q_a+Q_n)}$$

and zero mean. The variance can be converted to an effective number of bits at the output of the multiplication,

$$Q_{z_e} = 1 - \log_2\left(\sqrt{12\sigma_{q_z}^2}\right)$$

The number of effective bits (bits above the quantization noise) at the output of the mixing operation is shown in Fig. 4 versus the number of ADC bits for numbers of NCO output bits ranging from 2 to 14. Given the number of ADC bits, the width of the NCO look-up table is selected by choosing the number of NCO output bits such that an increase in the number of NCO output bits does not significantly increase the effective number of bits at the output of the mixer. The current advanced receiver design has an 8-bit ADC; therefore, an NCO look-up table width of 10 bits was selected.

Using the mean-square-error (MSE) between the look-up-table values and a pure tone as a metric of the fidelity of the NCO output sequence,

$$\text{MSE} = \frac{1}{2\pi}\int_0^{2\pi}\left(\sin(\theta) - \frac{\left\lfloor \sin\left(\left\lfloor \theta\frac{2^L}{2\pi}\right\rfloor \frac{2\pi}{2^L}\right) 2^{Q_n-1}\right\rfloor}{2^{Q_n-1}}\right)^2 d\theta$$

one can select the appropriate length of the look-up table, 2^L, for a given width, Q_n. Figure 5 contains a plot of the MSE versus the number of address bits in the look-up table (output quantization levels), for look-up table widths from 2 to 14 bits. For a given output quantization, one chooses the look-up table

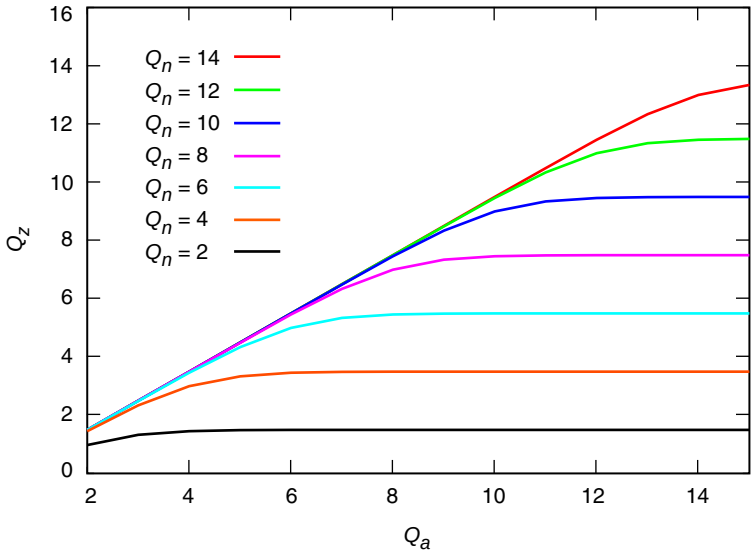

Fig. 4. Effective number of bits at the output of the mixer, Q_z, for a given ADC sample quantization, Q_a, and NCO output quantization, Q_n.

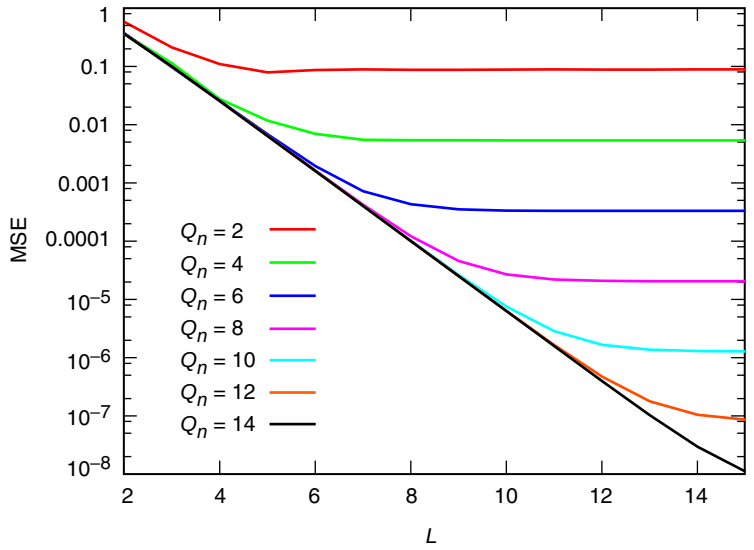

Fig. 5. Number of address bits in NCO look-up table versus the MSE between the NCO output and an infinite precision, infinite resolution sinusoid.

length such that any increase in the length provides little gain. For example, in the case of a 10-bit output quantization, there is very little reduction in the MSE for look-up table lengths beyond 2^{12} (12 address bits); a 2^{12} by 10 bit look-up table is selected for the current design of the advanced receiver.

B. Matched Filter

In order to have a spectrally efficient signaling scheme without incurring a significant performance loss due to inter-symbol interference (ISI) or the increased complexity necessary to combat ISI, Nyquist pulse shaping, in the form of a square-root raised-cosine (SRRC) transmit pulse shape and corresponding receiver filter, is used in this system. In a parallel architecture receiver, for each vector of M input

samples a corresponding vector of M filtered output samples must be calculated. For a finite impulse response (FIR) filter, this results in an M-fold increase in the complexity of the filter as the equivalent of a separate filter for each value of the output vector is necessary. In order to implement such a structure in a resource-limited device such as an FPGA, a fixed coefficient design using canonical signed digit (CSD) number representations is implemented.

The SRRC waveform is continuous and bandlimited; implementing a filter matched to this waveform as an FIR filter with fixed-point arithmetic will result in ISI and adjacent channel interference (ACI) due to the truncation and fixed point quantization. For a filter, h_r, of length N, the normalized peak and average ISIs incurred from the P adjacent symbols are given by

$$\mathrm{ISI}_{\mathrm{peak}} = \frac{\sum_{i=-P}^{P} \left| \sum_{j=-(N-1)/2}^{(N-1)/2} h_r[j] h_t(-jT_s - iT) \right|}{\sum_{j=-(N-1)/2}^{(N-1)/2} h_r[j] h_t(-jT_s)}$$

and

$$\mathrm{ISI}_{\mathrm{ave}} = \frac{\frac{1}{2^{2P+1}} \sum_{\vec{d}} \sum_{i=-P}^{P} d_i \sum_{j=-(N-1)/2}^{(N-1)/2} h_r[j] h_t(-jT_s - iT)}{\sum_{j=-(N-1)/2}^{(N-1)/2} h_r[j] h_t(-jT_s)}$$

where $h_t(t)$ is the transmitted pulse shape, T is the symbol time, and T_s is the sample time and $\vec{d}$ is a data vector of length $2P+1$. Note that this is a non-linear function of the length: longer filters don't necessarily give lower ISI. The average and peak ACIs are given by

$$\mathrm{ACI}_{\mathrm{peak}} = \frac{\max_{\frac{\pi(1+\alpha)T_s}{T} \le \omega < \pi} \{|H_r(\omega)|\}}{\frac{T}{\pi(1-\alpha)T_s} \int_0^{\pi(1-\alpha)T_s/T} |H_r(\omega)| d\omega}$$

and

$$\mathrm{ACI}_{\mathrm{ave}} = \frac{\frac{1}{\pi - [\pi(1+\alpha)T_s/2T]} \int_{\pi(1+\alpha)T_s/T}^{\pi} |H_r(\omega)| d\omega}{\frac{T}{\pi(1-\alpha)T_s} \int_0^{\pi(1-\alpha)T_s/T} |H_r(\omega)| d\omega}$$

where $H_r(\omega)$ is the frequency response of the receive filter. Using these metrics, one can select the minimum-length filter that provides adequate suppression of ISI and ACI. For a system with 8 samples per symbol and an SRRC filter with an excess bandwidth of 35 percent, $\alpha = 0.35$, the ISI and ACI metrics calculated for a set of filter sizes between 17 and 97 are given in Table 2. Note the ISI of a 49-tap filter is less than that of a 65-tap filter, demonstrating that a longer filter does not necessarily lower the ISI. For the prototype advanced receiver, a 33-tap filter length provides sufficient ISI and ACI rejection and is selected.

Rather than quantizing the filter coefficients to a uniformly sampled quantization space and implementing the filter convolution with fixed-point multipliers, the filter coefficients are mapped to the non-uniformly spaced quantized CSD number representation, where the convolution can be performed without multipliers [5]. Canonical signed digit numbers use a trinary alphabet $(-1, 0, 1)$ as digits to base 2 number such that each CSD number does not have adjacent non-zero digits and is of minimum Hamming weight [6, pp. 149–152]. The multiplication of a binary number with a CSD number, therefore, can be implemented with a small number of shift, add, and subtract operations, reducing the hardware complexity of the filter.

Since not all numbers are representable as CSD numbers of a given quantization, mapping to the CSD space therefore will result in a distortion of the filter frequency response. To select the level of CSD quantization, the length of the CSD number, and the maximum number of non-zero digits, the MSE between the CSD quantized filter coefficients and the infinite precision set of coefficients are calculated. One then can select the quantization level that provides the best performance versus complexity trade-off. For the 33-tap SRRC with a 35 percent excess bandwidth filter, the MSE for CSD quantization levels of 8 through 12 digits with 2 through 5 non-zero digits were calculated and are given in Table 3. Observe that there is no gain in going from 4 to 5 non-zero digits; therefore, the 12-4 CSD quantization is selected for the prototype advanced receiver. The impulse and frequency responses of the 12-4 CSD quantized filter along with those of an infinite precision filter are shown in Figs. 6 and 7.

The number of bits used in operations throughout the filter are chosen such that the minimum number needed to maintain the effective number of bits (not add to the quantization) and avoid overflow is used. The input samples to the filter result from the mixing operation of the NCO output and the ADC samples. For an 8-bit ADC and 10-bit NCO, the 18-bit product has approximately 8 effective bits, as shown in

Table 2. Adjacent channel interference and inter-symbol interference versus matched filter size for SRRC pulse, $\alpha = 0.35$.

Filter set length	Maximum ACI, dB	Average ACI, dB	Maximum ISI, dB	Average ISI, dB
17	−12.11	−32.33	−14.39	−17.90
33	−17.90	−39.05	−24.15	−29.34
49	−23.07	−47.48	−51.36	−59.93
65	−25.09	−56.87	−36.89	−42.25
81	−27.00	−54.86	−51.84	−58.80
97	−29.10	−59.07	−65.62	−73.85

Table 3. Mean-squared error between CSD quantized filter and ideal filter versus quantization level.

CSD coefficient width	Maximum number of non-zero digits			
	2	3	4	5
8	2.31×10^{-3}	5.55×10^{-4}	5.55×10^{-4}	5.55×10^{-4}
9	2.44×10^{-3}	1.68×10^{-4}	1.68×10^{-4}	1.68×10^{-4}
10	2.62×10^{-3}	6.68×10^{-5}	3.53×10^{-5}	3.53×10^{-5}
11	2.56×10^{-3}	4.73×10^{-5}	1.01×10^{-5}	1.01×10^{-5}
12	2.56×10^{-3}	4.54×10^{-5}	3.17×10^{-6}	3.17×10^{-6}

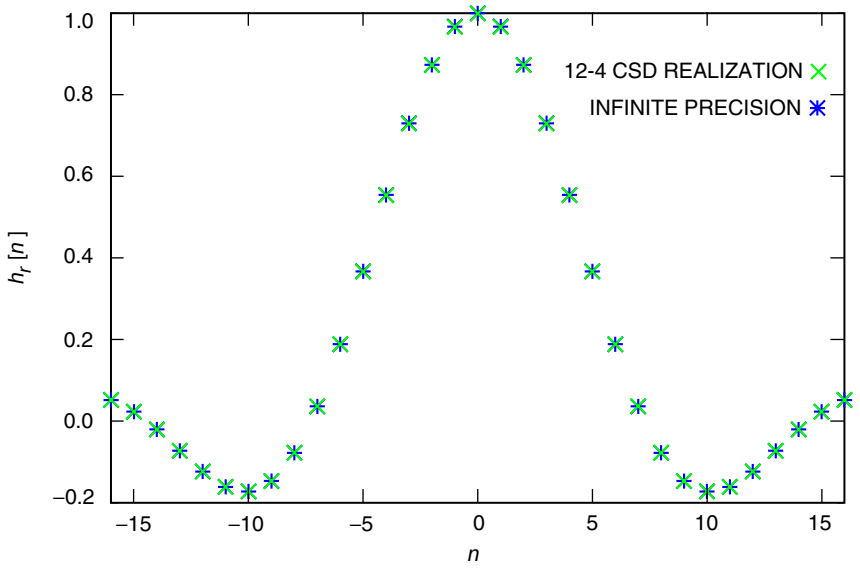

Fig. 6. The impulse response of a 12-4 CSD quantized and unquantized 33-tap SRCC filter with a 35 percent excess bandwidth (square-root raised-cosine, $\alpha = 0.35$, FIR filter).

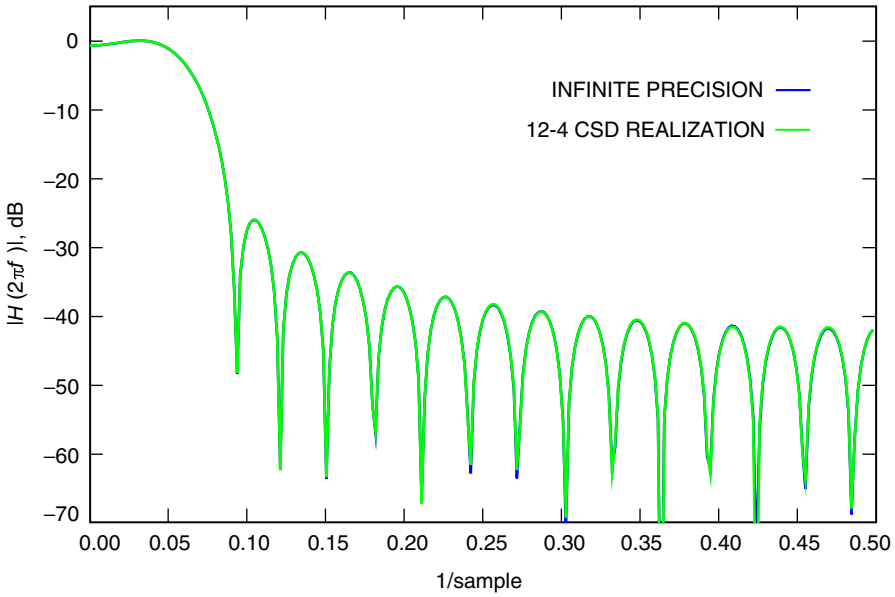

Fig. 7. The frequency response of a 12-4 CSD quantized and unquantized 33-tap SRRC filter with a 35 percent excess bandwidth ($\alpha = 0.35$).

Fig. 4. In order to keep the noise resulting from truncation of the 18-bit product below the level of the quantization noise, the 18-bit product is truncated to 10 bits. To mitigate overflow at the output of the FIR filter, after truncation to 10 bits, the input samples are scaled. In two's complement arithmetic, when adding a group of numbers (such as the tap-weight products), overflow of intermediate sums does not affect the final result as long as the final result is representable. A scaling factor of

$$\frac{1}{\sqrt{\sum_{k=-(N-1)/2}^{(N-1)/2} |h_r[k]|^2}}$$

is chosen to bound the total energy at the filter coefficient products to be less than the input energy [7, pp. 359–363]; this is not sufficient to prevent overflow at the output, but it does reduce the occurrence. Evaluating this for the selected CSD coefficients and choosing the smallest representative power of two, the scaling can be implemented by shifting the input samples by 2 bits. The product of the scaled input values and the N tap weights generates a set of 24 bit numbers. In truncating these products and their sum (the filter output), it is desired to keep the resulting truncation noise level below that of the quantization noise at the filter output:

$$\frac{N2^{-2(Q_t-1)}}{12} + \frac{2^{-2(Q_o-1)}}{12} \leq \frac{2^{-2(Q_i+Q_s+Q_b-1)}}{12}$$

where Q_t is the number of bits the products are truncated to, Q_o is the number of bits in the truncated output, Q_i is the number of bits in a filter input sample, Q_s is the number of bits the input sample is shifted to prevent overflow, and Q_b is the number of bits desired to separate the truncation and quantization noise. For the advanced receiver, where $Q_i = 10$ and $Q_s = 2$, in order to satisfy a $Q_b = 3$ bit buffer between the truncation and quantization noise, an output size of $Q_o = 16$ bits with an internal quantization of $Q_t = 18$ bits was selected.

C. Symbol-Timing Recovery

The symbol-timing recovery loop estimates the received symbol timing offset τ and uses the estimate in order to determine which match-filtered signal output to use as soft-symbol decision statistics. The existing Block V receiver generally uses the data transition tracking loop (DTTL) to perform symbol synchronization. The DTTL is a coherent tracking algorithm that requires carrier phase recovery prior to symbol tracking; however, carrier phase tracking in the absence of a residual carrier tone generally requires symbol synchronization. In DSN practice, the symbol and carrier tracking loops are initiated together in a type of "bootstrapping" process [8]. In the current receiver architecture, we utilize a non-coherent symbol-timing recovery loop that precedes carrier phase recovery.

A block diagram of the non-coherent Gardner timing loop is shown in Fig. 8. A detailed description of the derivation of this structure may be found in [9]. The timing-error detector signal is given by

$$e_t(k) = I\left(k - \left\lfloor \frac{N_s}{2} \right\rfloor\right)\big(I(k) - I(k-N_s)\big) + Q\left(k - \left\lfloor \frac{N_s}{2} \right\rfloor\right)\big(Q(k) - Q(k-N_s)\big) \tag{6}$$

where $N_s = T/T_s$ is the number of samples per symbol, and $I(k)$ and $Q(k)$ are the match-filtered samples of the in-phase and quadrature baseband signals. The timing-error signal thus is formed from the difference between adjacent symbol values, which provide information about the derivative of the data signal, in conjunction with the midpoint sample, which indicates the magnitude of the timing error. The timing error signal e_t can easily be shown to be independent of carrier phase and may be used for both binary phase-shift keying (BPSK) and QPSK signals. However, for BPSK, once carrier phase is acquired, one of the two channels will contain only noise and thereafter should be disconnected for tracking operation. After loop filtering, the error signal is used to correct timing by adjusting the selection of samples out of the matched filter. In this implementation, the resolution of the timing correction thus is limited to the sampling frequency.

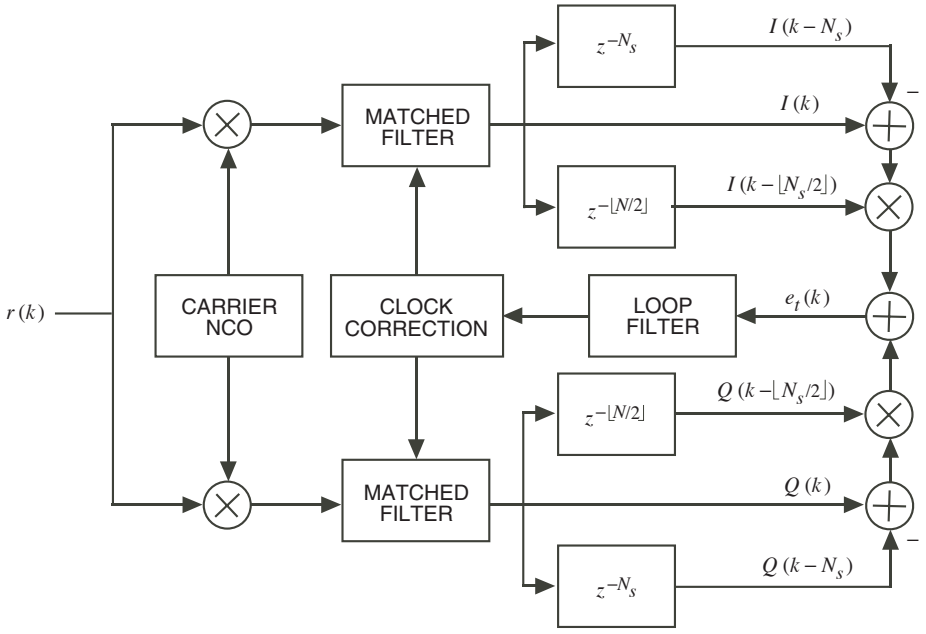

Fig. 8. Gardner timing recovery loop.

In order to analyze this loop and design the loop filter, we examine the digital baseband equivalent loop diagram shown in Fig. 9, described by the loop equation

$$\hat{\tau}(k) = A\big(\tau(k-L) - \hat{\tau}(k-L)\big) * f(k) \tag{7}$$

The quantity $\hat{\tau}(k)$ is the estimate of the timing offset τ at time k; A is the loop gain factor; and L is the bulk delay through the loop. The loop filter $f(k)$ is a second-order filter with digital transfer function

$$F(z) = \frac{\alpha(1-z^{-1}) + \beta}{(1-z^{-1})^2} \tag{8}$$

where α and β are loop coefficients that must be designed to obtain the desired response and loop bandwidth. The loop bandwidth is calculated as

$$B_l = \frac{1}{2}\int_{-1/2T_u}^{1/2T_u} |H(e^{j2\pi f T_u})|^2 df \tag{9}$$

where T_u is the loop update time and $H(f) = H(z)|_{z=e^{j2\pi f T_u}}$ is the loop transfer function, which may be obtained from Eq. (7) as

$$H(z) = \frac{AF(z)}{z^L + AF(z)} \tag{10}$$

Substituting Eq. (8) into Eq. (10), we can carry out the integration in Eq. (9) numerically. In the implementation of the symbol timing loop, $A = 0.015625$, $L = 20$, and $T_u = 1.25 \times 10^{-8}$ seconds. From

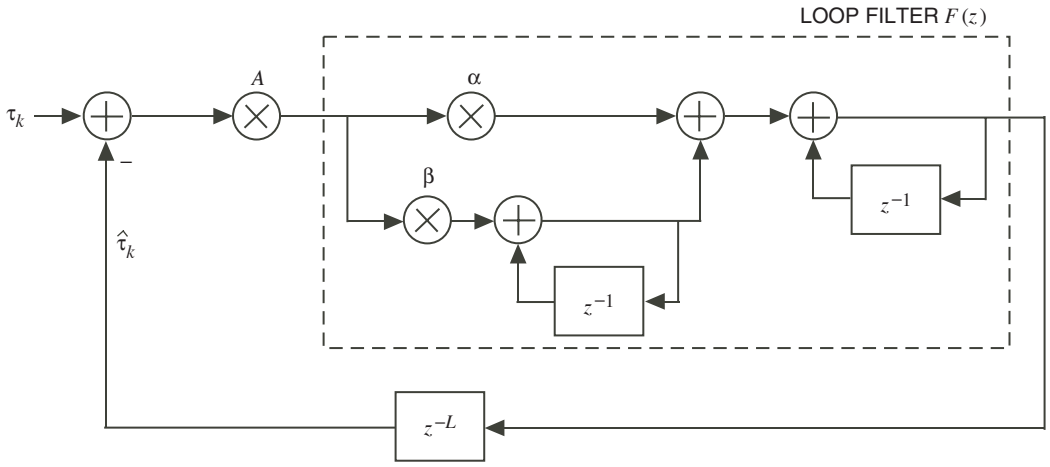

Fig. 9. Baseband equivalent loop.

analog loop analysis [10], the filter coefficients α and β may be calculated for a second-order critically damped loop with design loop bandwidth B_l^* as

$$\alpha = \frac{16}{5}\frac{B_l^* T_u}{A}$$
$$\beta = \frac{\alpha^2 A}{4} \tag{11}$$

The analog analysis only holds for $B_l T_u < 0.01$, however, so for larger digital bandwidths, we use the equations in Eq. (11) as a starting point and modify them by trial and error to obtain filter coefficients.

The performance of the symbol-timing recovery loop may be characterized by the timing-error variance, or jitter. The modified Cramer–Rao lower bound (MCRB) on normalized timing jitter for symbol-timing recovery via feedback loops and raised cosines is given by [11]

$$\text{MCRB} = \frac{B_l T_u}{2\left(\frac{1}{3}\pi^2(1+3\alpha^2) - 8\alpha^2\right)\frac{E_s}{N_o}} \tag{12}$$

where E_s/N_o is the symbol signal-to-noise ratio (SNR), which is twice the bit SNR for QPSK.

D. Carrier Phase Recovery

In the past, most deep-space missions have transmitted signal formats containing a residual carrier component in order to provide a reference for carrier phase tracking using a traditional phase-locked loop [3]. In the case when the residual carrier becomes fully suppressed, a Costas loop becomes necessary for tracking the data-modulated carrier. For the QPSK signals that are considered here, the Costas cross-over loop (also known as polarity-type Costas loop) [12] is used to estimate and track carrier phase. The Costas cross-over loop is a high SNR approximation to the maximum-likelihood (ML) phase estimator and is simpler to implement than the ML algorithm or its low SNR approximation. A functional block diagram is shown in Fig. 10. Here the phase error signal is formed by cross-multiplying the outputs of the

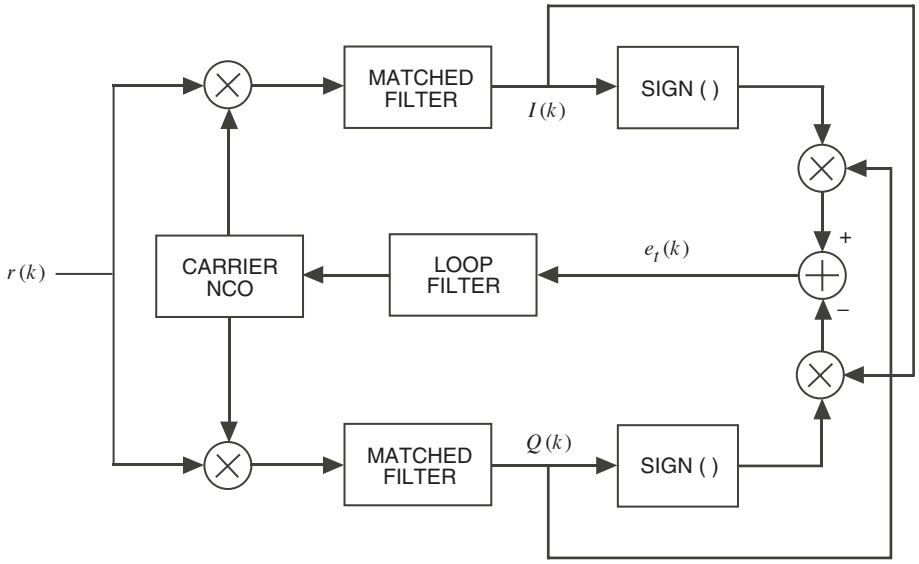

Fig. 10. Costas cross-over loop for carrier phase tracking.

in-phase and quadrature matched filters with the signs of those quantities from the opposite branches. The error signal thus formed is proportional to $sin(4\phi)$, where ϕ is the phase error. This results in a four-phase ambiguity that must be resolved in order to process the baseband symbols.

The baseband equivalent loop may also be represented by Fig. 9, substituting the loop input phase θ and the estimated phase $\hat{\theta}$ for the input and estimated time offsets τ and $\hat{\tau}$. In the implementation of the phase tracking loop, the loop gain is given by $A = 0.125$, and the bulk loop delay is given by $L = 29$. The loop bandwidth and loop transfer function are calculated as for the symbol tracking loop in Eqs. (9) and (10). As for the symbol-tracking loop filter, the analog loop analysis may be used as a starting point for obtaining loop filter coefficients.

The normalized phase tracking error performance for a polarity-type Costas loop is given by

$$\left(\frac{\sigma_\phi}{2\pi}\right)^2 = \frac{B_l T_u}{2\mathcal{S}_L \dfrac{E_s}{N_o}} \tag{13}$$

where the squaring loss $\mathcal{S}_\mathcal{L}$ is given by

$$\mathcal{S}_L = \frac{\left(\operatorname{erf}\left(\sqrt{\dfrac{E_s}{N_o}}\right) - \sqrt{\dfrac{4E_s}{\pi N_o}}\, e^{-(E_s/N_o)}\right)^2}{\left(1 + 2\dfrac{E_s}{N_o} - \sqrt{\dfrac{2}{\pi}}\, e^{-(E_s/N_o)} + \sqrt{2\dfrac{E_s}{N_o}}\operatorname{erf}\left(\sqrt{\dfrac{E_s}{N_o}}\right)\right)^2} \tag{14}$$

IV. Performance Results

In order to validate the receiver design and evaluate performance, fixed-point software simulations were performed that replicate the parallel architecture, including NCO, matched filter, and symbol and phase tracking loops. Uncoded bit-error rates (BERs) were evaluated as well as tracking-error variances. Finally, coded performance was tested for various block length low-density parity-check (LDPC) codes using a separate software decoder.

Figure 11 shows the simulated uncoded BER as a function of the bit SNR, E_b/N_o. Here, both symbol timing and carrier phase are assumed to be known, so the tracking loops are turned off, and we simply evaluate the design of the matched filter. We see that there is essentially no loss in the fixed point simulation results with respect to the theoretical BER given by BER $= Q(\sqrt{2E_b/N_o})$, where $Q(\cdot)$ is the complementary Gaussian distribution function.

Figures 12 and 13 demonstrate the performance of the Gardner symbol tracking loop. For these results, the carrier phase was assumed to be known perfectly, and only the symbol tracking loop was turned on. Two digital loop bandwidth values of $B_l T_u = 0.00138$ and $B_l T_u = 0.0125$ (or analog loop bandwidths $B_l = 100$ kHz and $B_l = 1$ MHz for an update time of $T_u = 1.25 \times 10^{-8}$ s) were tested. Figure 12 shows the normalized timing-error variance $(\sigma_t/T_{sym})^2$ as a function of the bit SNR. Also plotted is the MCRB on the timing-error variance. We observe that the simulated timing-error variance appears to reach a floor as the SNR increases; while this is expected due to the limited resolution of the timing correction, it is unclear why the value of the floor differs for the two loop bandwidths. We also note that at very low SNR the error variance makes a jump; this is simply an artifact of incorrect variance measurement when the timing offset "wraps around" from zero to T_{sym}. This is borne out by the results in Fig. 13, which show less than a 0.2-dB loss from symbol-timing error relative to the ideal BER curve, even in the low SNR region.

Figures 14 and 15 show analogous results for the Costas cross-over phase tracking loop. Here the symbol timing was assumed to be known perfectly, and loop bandwidth values similar to those used for the symbol-tracking results were used. In Fig. 14, the simulated normalized phase-tracking error variance $(\sigma_\phi/2\pi)^2$ is plotted along with the theoretical phase jitter as a function of bit SNR. As the SNR increases, we see good correspondence between simulation and theory. However, for lower SNR values, the

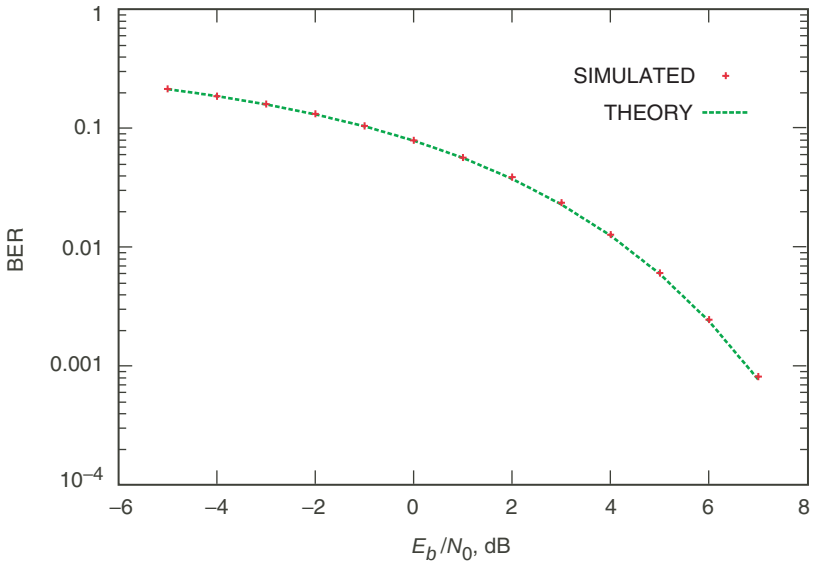

Fig. 11. Uncoded BER with matched filter, perfect tracking assumed (phase tracking off, symbol tracking off).

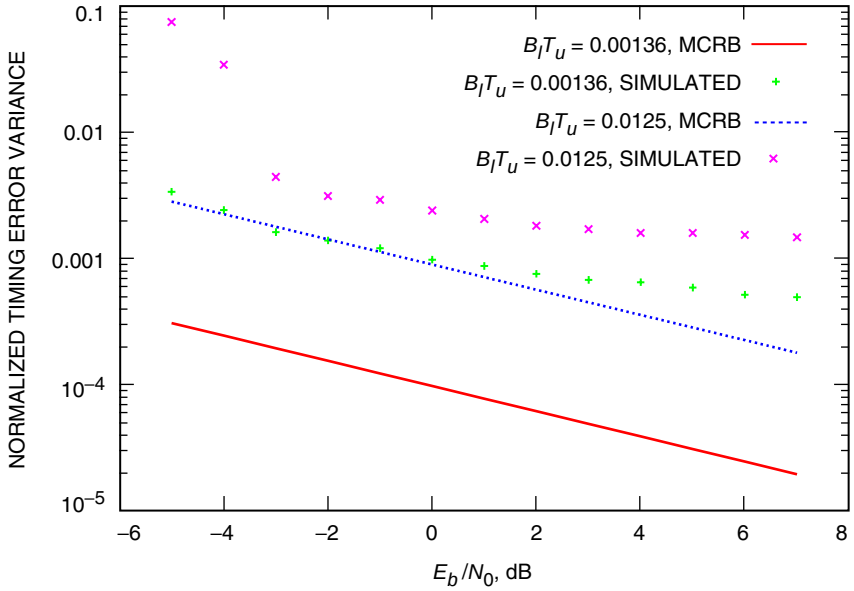

Fig. 12. Symbol-tracking error variance versus E_b/N_0, carrier phase known perfectly (phase tracking off, symbol tracking on).

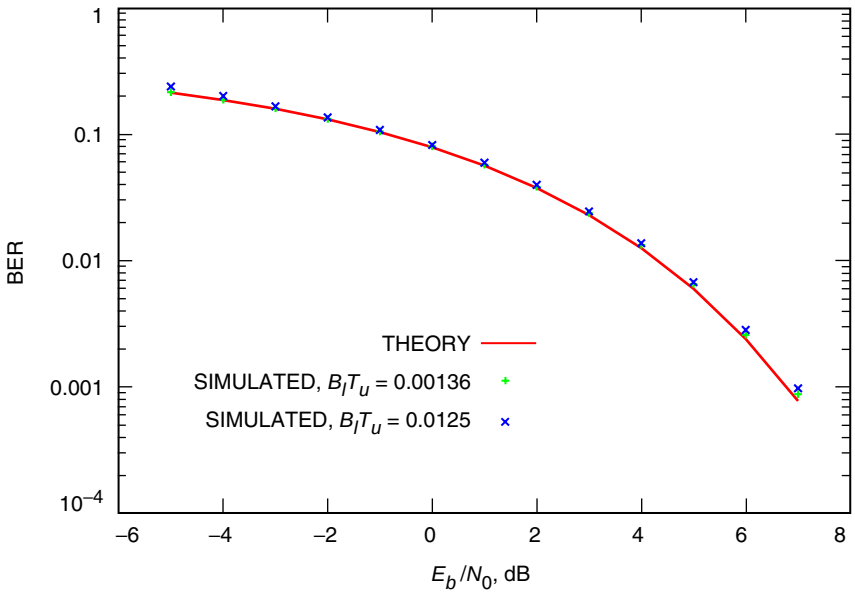

Fig. 13. Uncoded BER with symbol tracking loop on and phase tracking loop off (phase known perfectly).

simulated phase error variance diverges from the lower bound. Some of this is due to incorrect variance measurement as the phase wraps around from zero to 2π, but for $E_b/N_o < -2$ dB there are real problems with tracking when a loop bandwidth of $B_l T_u = 0.0126$ is used, as may be seen from Fig. 15, in which the BER is degraded by several decibels under these conditions. When $E_b/N_o > -3$ dB, or if a lower loop bandwidth of $B_l T_u = 0.00126$ is used, the losses are limited to less than 0.2 dB. If the loop bandwidth is constrained to be greater than 0.001 due to channel dynamics, the low SNR carrier phase-tracking structure [12] should be considered for implementation.

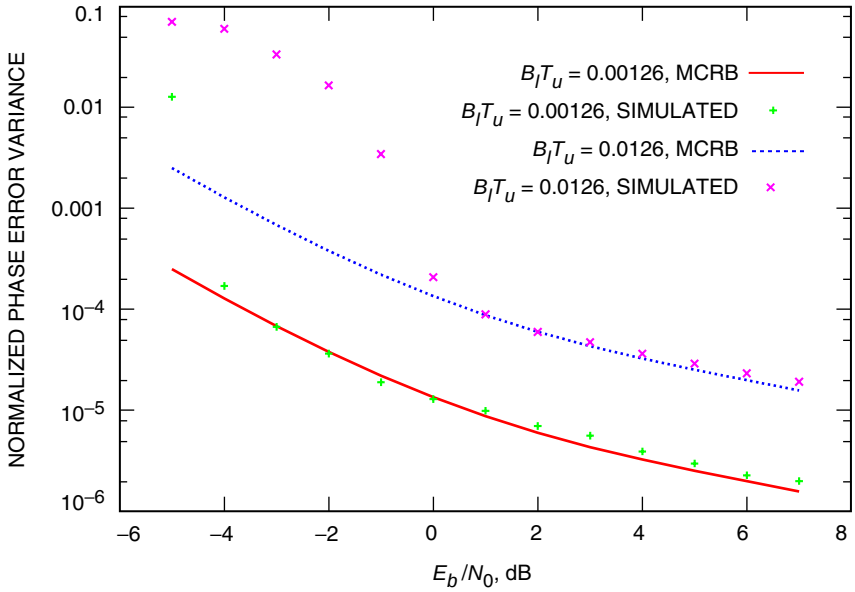

Fig. 14. Phase tracking error variance versus E_b/N_0, symbol timing known perfectly (phase tracking on, symbol tracking off).

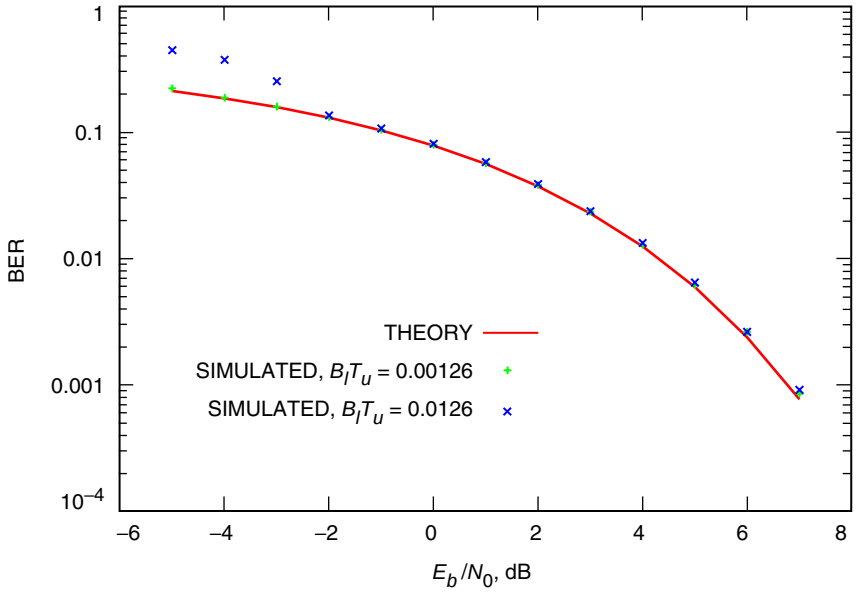

Fig. 15. Uncoded BER with phase tracking loop on and symbol tracking loop off (timing known perfectly).

In Fig. 16, we show uncoded BER performance when both symbol-timing and carrier-phase tracking loops are turned on with identical loop bandwidths. Along with the ideal curve, the two cases of $B_l T_u = 0.001$ and $B_l T_u = 0.01$ are tested for a static phase offset, showing losses at the low end of the scale for the higher loop bandwidth (again due to the phase-tracking loop). Furthermore, an additional curve shows bit-error rate performance when a 5-kHz IF input frequency offset is added and the loop bandwidth is set to $B_l T_u = 0.001$, showing that the second-order loop is able to track the dynamic phase when $E_b/N_o > -4$ dB.

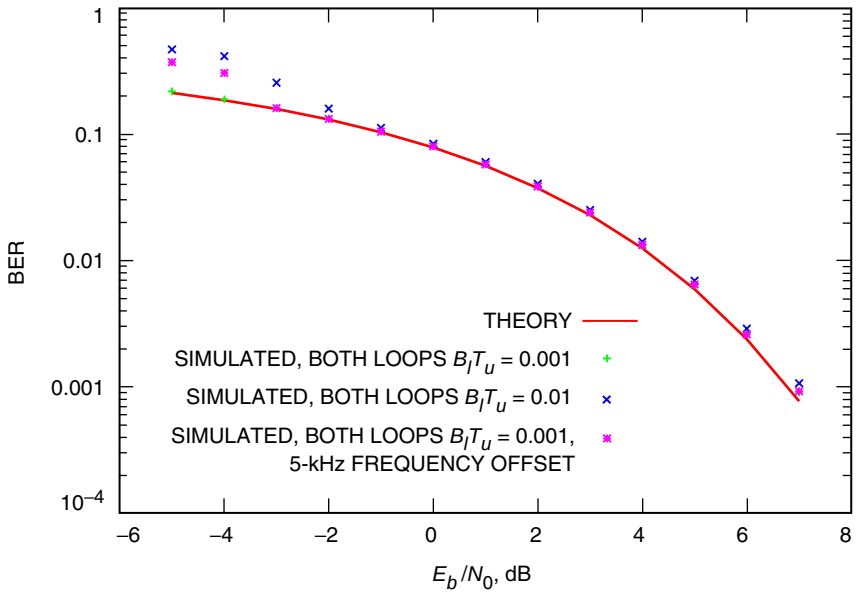

Fig. 16. Uncoded BER with both tracking loops on (fixed-point receiver simulation).

In order to provide validation of the real-time digital processing algorithms, test vectors representing sampled waveforms were employed as inputs to the FPGA hardware. These test vectors corresponded to 8-bit amplitude-quantized sequences of a digitized intermediate frequency (IF) waveform. During this preliminary evaluation, the hardware was operated with an effective sample rate of 1300 MHz, a QPSK symbol rate of 162.5 MHz, and an IF of approximately 400 MHz. This yielded a normalized time resolution of 8 samples per symbol. Both carrier-tracking and symbol-synchronizer loop circuits were enabled, and their loop bandwidths were both set to 1.3 MHz, $B_lT_u = 0.016$. In Fig. 17, the uncoded bit-error rates are shown for both the digital hardware and theoretical QPSK performance results. As can be seen from this figure, the real-time hardware performance matches the 0.25-dB degradation observed in the software simulation for the 10^{-3} BER regime and slightly larger loss of approximately 0.75 dB as the uncoded BER approaches 10^{-5}. In SNR operating regimes where error correction will be employed (typically greater than 1 percent uncoded BER for a variety of code rates), the implementation loss of the digital receiver is expected to be in the 0.1 dB range.

In addition to the tracking-error variance and uncoded BER simulations, matched-filter output soft symbols were recorded and fed into a software LDPC decoder. The coded performance was evaluated without implementing an LDPC encoder, so the soft symbols were multiplied by their corresponding bit values in order to generate the all-zero codeword for decoder testing. Figures 18 through 20 show the resulting frame-error rates for three different block lengths of 1024, 4096, and 16384, with each graph showing curves for the three code rates of 1/2, 2/3, and 4/5. The LDPC codes are from the AR4JA family of LDPC codes [13,14]. The theoretical performance curves for these codes were obtained from documented tests of FPGA hardware and software decoders [13,14]. Although more operating points must be tested to form more complete curves, we observe approximately 0.2-dB losses for the receiver data.

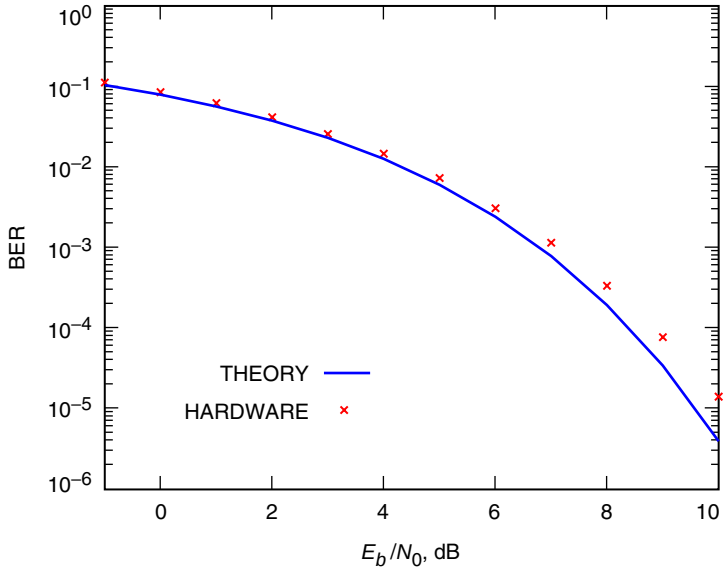

Fig. 17. QPSK FPGA hardware uncoded BER with both tracking loops on, $B_l T_u = 0.016$.

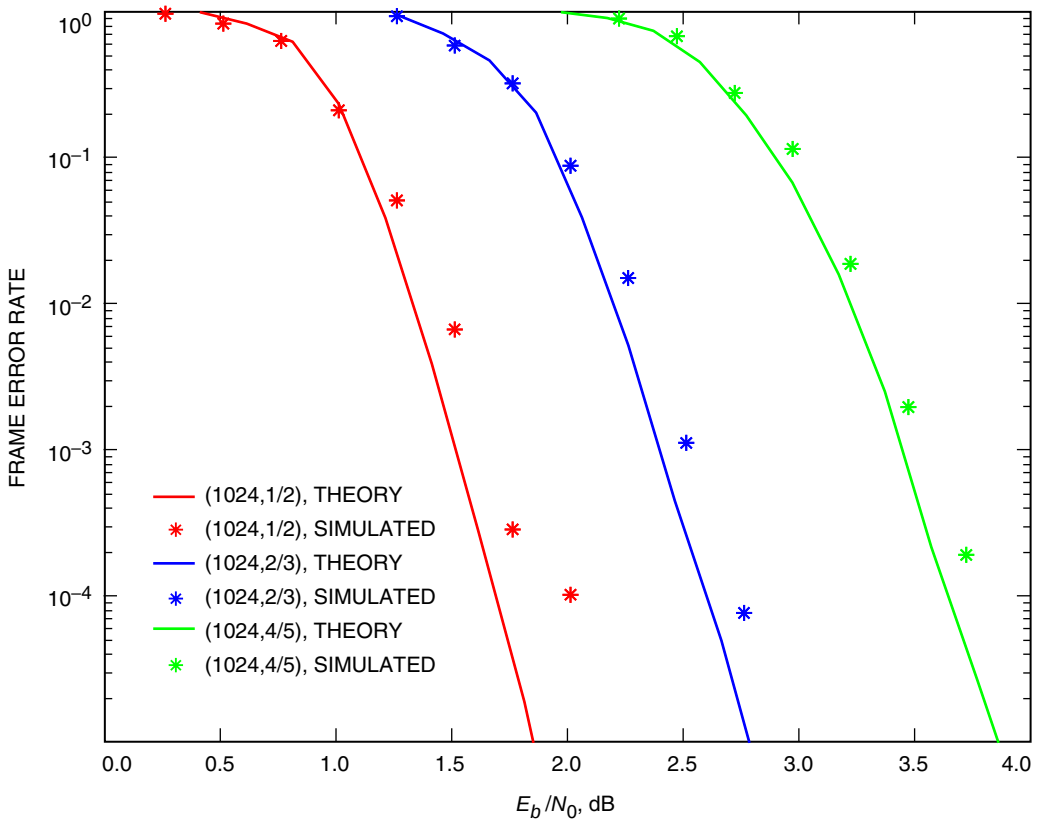

Fig. 18. LDPC decoded frame error rates, block length 1024, both tracking loops on.

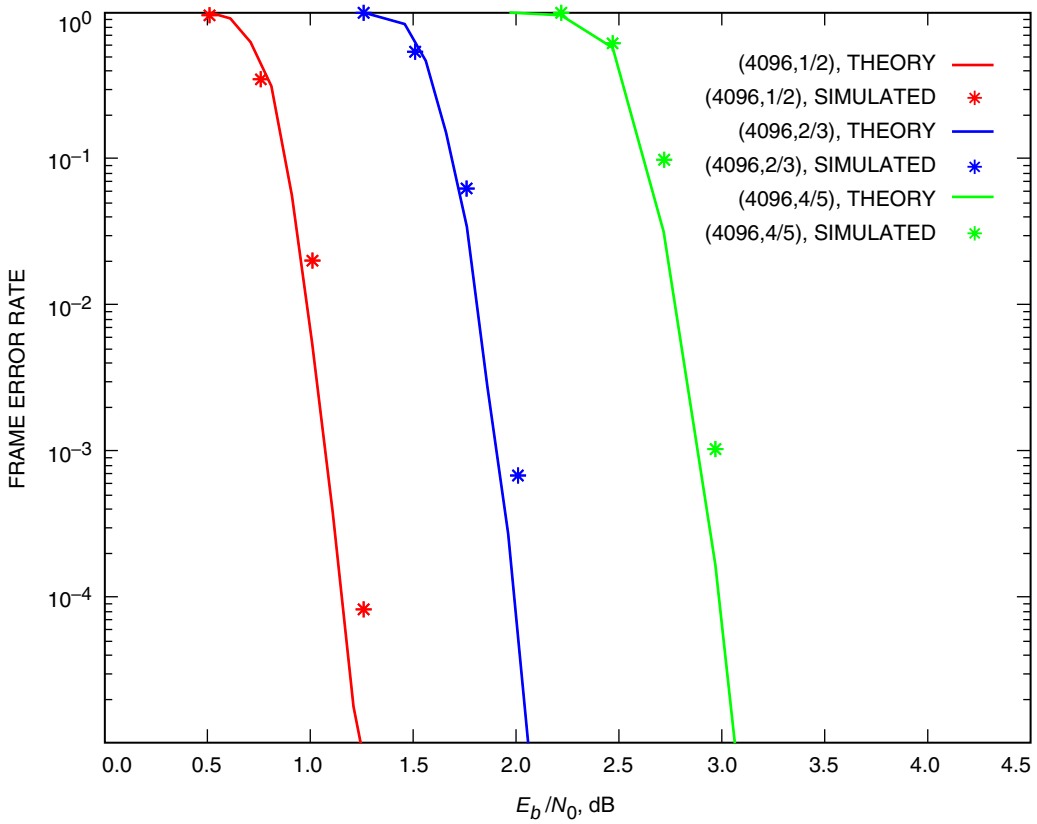

Fig. 19. LDPC decoded frame error rates, block length 4096, both tracking loops on.

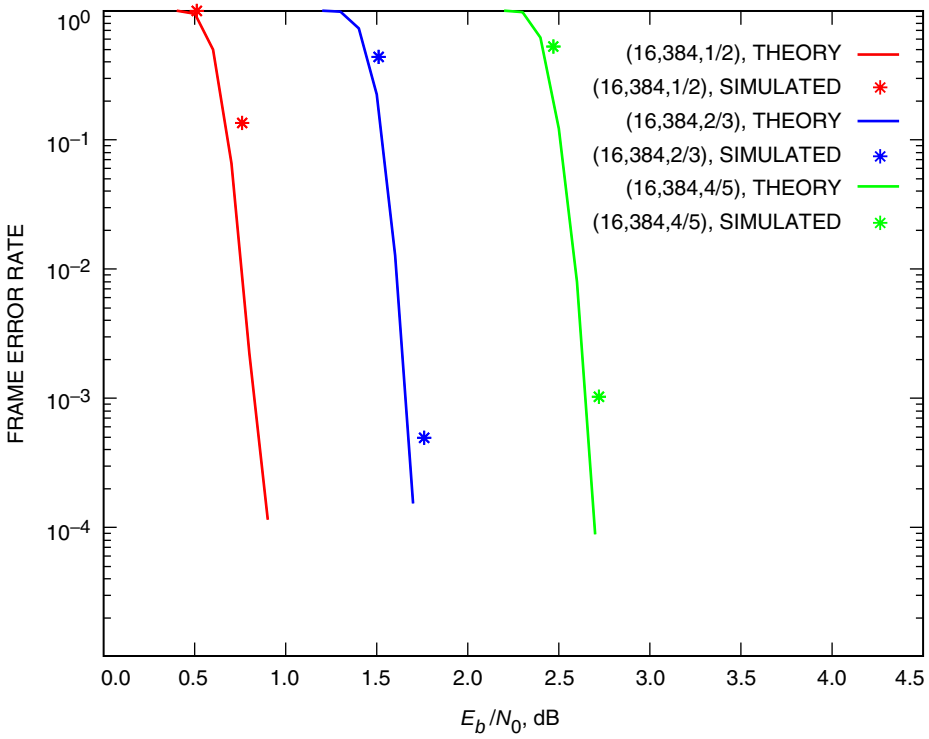

Fig. 20. LDPC decoded frame error rates, block length 16,384, both tracking loops on.

V. Conclusions

A preliminary architecture for a wideband telemetry receiver has been presented, along with an explanation of the design choices for development of the NCO, matched filter, and symbol and phase tracking loops for a square-root raised-cosine shaped data pulse modulated fully suppressed carrier QPSK signal. Uncoded and coded simulation results show bit-error probability performance within 0.2 dB of theoretical models, as does preliminary FPGA testing.

References

[1] R. Navarro and J. Bunton, "Signal Processing in the Deep Space Array Network," *The Interplanetary Network Progress Report*, vol. 42-157, Jet Propulsion Laboratory, Pasadena, California, pp. 1–17, May 15, 2004. http://ipnpr/progress_report/42-157/157N.pdf

[2] S. Hinedi, "A Functional Description of the Advanced Receiver," *The Telecommunications and Data Acquisition Progress Report 42-100, October–December 1989*, Jet Propulsion Laboratory, Pasadena, California, pp. 131–149, February 15, 1990. http://ipnpr/progress_report/42-100/100K.PDF

[3] J. Yuen, ed., *Deep Space Telecommunications Systems Engineering*, New York: Plenum Press, 1983.

[4] M. K. Simon, S. M. Hinedi, and W. C. Lindsey, *Digital Communication Techniques*, Englewood Cliffs, New Jersey: Prentice Hall, 1995.

[5] H. Samueli, "An Improved Search Algorithm for the Design of Multiplierless FIR Filters with Powers-of-Two Coefficients," *IEEE Transactions on Circuits and Systems*, vol. 36, pp. 1044–1047, July 1989.

[6] K. Hwang, *Computer Arithmetic, Principles, Architecture, and Design*, New York: Wiley, 1979.

[7] A. V. Oppenheim and R. W. Schafer, *Discrete-Time Signal Processing*, New Jersey: Prentice-Hall, Inc., 1989.

[8] *DSMS Telecommunications Link Design Handbook*, 810-005, "34-m and 70-m Telemetry Reception," 207, rev. A, Jet Propulsion Laboratory, Pasadena, California, June 13 2003.

[9] F. Gardner, "A BPSK/QPSK Timing-Error Detector for Sampled Receivers," *IEEE Transactions on Communications*, vol. 34, pp. 423–429, May 1986.

[10] S. Aguirre and W. J. Hurd, "Design and Performance of Sampled Data Loops for Carrier and Subcarrier Tracking," *The Telecommunications and Data Acquisition Progress Report 42-79, July–September 1984*, Jet Propulsion Laboratory, Pasadena, California, pp. 81–95, November 15, 1984. http://ipnpr/progress_report/42-79/79H.PDF

[11] U. Mengali and A. N. D'Andrea, *Synchronization Techniques for Digital Receivers*, New York: Plenum Press, 1997.

[12] M. K. Simon, "On the Optimality of the MAP Estimation Loop for Carrier Phase Tracking BPSK and QPSK Signals," *IEEE Transactions on Communications*, vol. COM-27, pp. 158–165, January 1979.

[13] D. Divsalar, S. Dolinar, and C. Jones, "Low-Rate LDPC Codes with Simple Protograph Structure," *IEEE International Symposium on Information Theory*, Adelaide, Australia, pp. 1622–1626, September 2005.

[14] D. Divsalar, C. Jones, S. Dolinar, and J. Thorpe, "Protograph Based LDPC Codes with Minimum Distance Linearly Growing with Block Size," *IEEE Global Communications Conference (GLOBECOM)*, Saint Louis, Missouri, pp. 1152–1156, December 2005.