PN Delta-DOR Signal Format Implementation

Zaid Towfic,* Thaddaeus Voss,* Mazen Shihabi,* and James Border†

ABSTRACT. — This article describes the design and implementation of the pseudorandom-noise (PN) delta differential one-way ranging (Delta-DOR) signal format on the Jet Propulsion Laboratory (JPL) Iris CubeSat Software Defined Radio. The Iris radio is now being used in deep space onboard the Mars Cube One (MarCO) spacecraft and will soon be utilized on multiple EM-1 missions. The spread-spectrum Delta-DOR format enables more accurate differential ranging measurements over the classical DOR tone format. The technique is applicable to deep space missions that require accurate navigation or require accurate angular position measurements for another purpose such as determining the ephemeris of a planet or small body. The classical Delta-DOR technique makes time delay measurements of spacecraft and quasar signals to determine spacecraft angular position in the radio reference frame defined by the quasar coordinates. The measurement system is configured to provide common-mode error cancellation as nearly as possible.

In the PN DOR mode (instead of a spacecraft modulating their downlink) with a sinusoidal signal, referred to as a DOR tone as done in the classical DOR mode, the sinusoidal tone is replaced with a spread spectrum signal. Using a spread spectrum DOR signal instead of a DOR tone enables cancellation of effects due to phase dispersion across the channels used to record quasar signals. This error source, referred to as phase dispersion or phase ripple, is currently the dominant measurement error for the classical DOR format. PN spreading will improve on classical DOR performance accuracy because by choosing the PN spreading code and shaping filter carefully, the spacecraft signal can be made to closely resemble the quasar signal resulting in reduction of the Delta-DOR error due to phase dispersion by 80 percent to 90 percent over the classical Delta-DOR approach.

The article will describe the choice of a Gold code sequence that possesses good autocorrelation properties as well as excellent cross-correlation properties that was used to spread the DOR tone, and the choice of root-raised-cosine (RRC) chip-shaping filter to reduce the amount of excess-bandwidth of the output waveform. The paper will also describe the path that was taken for the FPGA implementation on the Iris radio including a Simulink model of the PN Delta-DOR module that is used for automatic HDL code generation. This module was then integrated into the Iris firmware and verified.

* Flight Communications Systems Section.

† Tracking Systems and Applications 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. © 2019 California Institute of Technology. U.S. Government sponsorship acknowledged.

Future work includes the integration of the designed subsystem into the Universal Space Transponder and extension of the derived waveform to the Ka-band. We hope that the outlined PN Delta-DOR implementation can be integrated into an upcoming mission that utilizes the Iris radio (such as EM-1 missions).

I. Introduction and Requirements

The delta differential one-way ranging (Delta-DOR) technique makes time-delay measurements of spacecraft and quasar signals to determine the spacecraft's angular position in the radio reference frame as defined by the quasar coordinates. These measurements are often critical to deep space missions requiring accurate navigation, such as a spacecraft on a direct trajectory from Earth to a landing on Mars. The measurement system is configured to provide as much common-mode error cancellation as possible. The delta delay between spacecraft and quasar measurements has high accuracy and can be used to reduce errors from station clock offsets.

Improving Delta-DOR accuracy to meet more stringent navigation requirements remains a goal.

In current usage, spacecraft modulate their downlink with a sinusoidal signal, referred to as a DOR tone, to provide a wide span of bandwidth that enables a precise group-delay measurement. This wide bandwidth is especially needed to make precise group-delay measurements of weak quasar signals. For X-band downlinks, a DOR subcarrier frequency of about 19 MHz is typical, as this is within the spectrum allocation for deep space research at X-band. Quasar signals are typically recorded in channels with a bandwidth of 8 MHz.

During a Delta-DOR measurement session, quasars that are angularly close to the spacecraft are observed to provide common-mode cancellation of spatial effects including station coordinate errors and media calibration errors. Antennas slew back and forth between spacecraft and quasars to provide common-mode cancellation of temporal effects, including station clock drifts. Quasar signals are recorded in channels centered on the received frequencies of spacecraft DOR tones to provide common-mode cancellation of clock offsets and station group delays. However, the current architecture does not provide cancellation of effects due to phase dispersion across the channels used to record quasar signals. This error source, referred to as phase dispersion or phase ripple, is often the dominant measurement error for Delta-DOR. The per-channel phase error is typically in the range of 0.2 to 0.5 degree [1].

Receiving systems used for deep space communications are among the most sensitive and tightly specified radio systems in existence. Even so, phase linearity across channels of several MHz bandwidth, at the level of a fraction of a degree of phase, cannot be assumed. However, common-mode cancellation of phase dispersion effects is possible for Delta-DOR by using a spread-spectrum DOR signal. The basic idea is to make the spacecraft signal more closely resemble the quasar white noise signal.

Consider Figure 1. The two spectra on the left-hand side are data from an actual Delta-DOR measurement. The top-left figure is the spectrum of receiver noise in a channel, with the

flat quasar signal buried in the noise. The bottom-left figure is the spectrum of a spacecraft DOR tone in a noise channel. The two spectra on the right-hand side are theoretical spectra of a DOR tone spread with a PN code. The top-right figure is the spectrum of a PN code with rectangular chips. The bottom-right figure is the spectrum of a PN code with shaped chips.

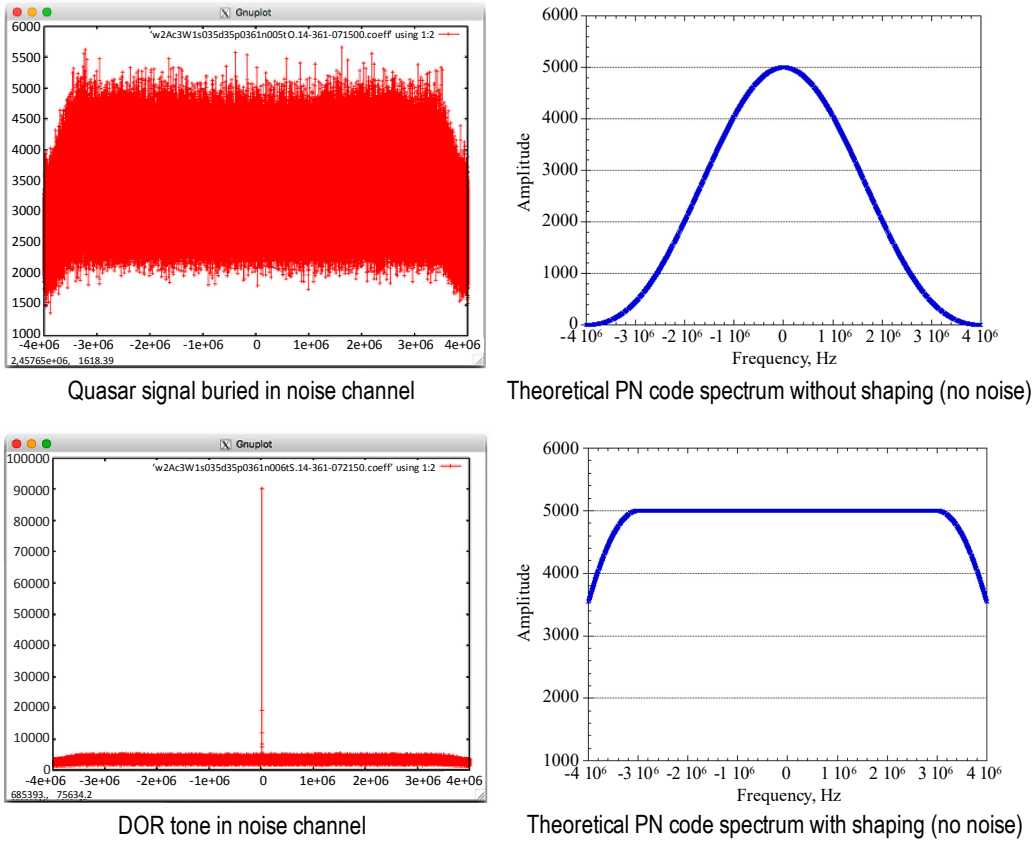

Figure 1. The two spectra on the left-hand side are data from an actual Delta-DOR measurement. The two spectra on the right-hand side are theoretical spectra of a DOR tone spread with a PN code.

By choosing the PN spreading code and shaping filter carefully, the spacecraft signal can closely resemble the quasar signal. Reduction of the Delta-DOR error due to phase dispersion can be reduced by 80 to 90 percent.

While PN spreading will improve Delta-DOR performance in the X-band, even more improvement is possible for the Ka-band. Since quasar flux is lower in the Ka-band and ground station system noise temperature is higher, more bandwidth is required for group-delay measurements in the Ka-band. This added bandwidth improves on the precision of quasar group-delay measurements at X-band. A DOR subcarrier frequency of about 160 MHz is envisaged for the Ka-band with a flat-spectrum signal filling a bandwidth of about 32 MHz centered on the subcarrier sidebands.

The next section discusses the general requirements of the PN Delta-DOR waveform, followed by a discussion of how the structure of the waveform will be influenced by these

requirements and how appropriate waveform parameters will be selected. We then outline the fixed-point design of the PN Delta-DOR module and discuss its integration into the JPL Iris CubeSat radio firmware. The Iris radio is now in deep space onboard the MarCO spacecraft and will soon be utilized on multiple EM-1 missions [2–3].

II. X-band PN Delta-DOR Iris Design

These are the key requirements for the X-band PN Delta-DOR implementation, as discussed in the previous section:

- 8 MHz spread-spectrum bandwidth on each side of the spectrum in order to match the quasar signal bandwidth during the Delta-DOR processing
- Each 8 MHz signal is used to spread the Iris 19 MHz classical Delta-DOR tone
- Code period >1 ms in order to maximize the integration time for the Delta-DOR PN sequence correlation (increasing the signal-to-noise ratio and ambiguity resolution capability)
- Code must have good autocorrelation properties in order to improve the time resolution during the Delta-DOR processing
- It is also desirable that the code have good cross-correlation properties
- A small roll-off factor of <0.3 must be used on the pulse-shaping filter in order to make the spectrum as flat as possible in the passband

The next sections describe a suitable code and chip-shaping filter to be used on the Iris processor to satisfy the above requirements. First, we determine the appropriate code rate in chips per second and the associated chip-shaping filter, and then we determine the code structure and primitive polynomials associated with the chosen code.

A. Code Rate

Given the requirements, and the operating frequency of the field programmable gate arrays (FPGAs) on the Iris transponder, a chip rate ≤8 MHz must be chosen. In addition, since the chips are going to be filtered by a root-raised-cosine (RRC) filter [4], we will observe that the following relationship between the output bandwidth of the filter and the input chip rate will hold:

$$BW = R_c(1+\beta), \tag{1}$$

where β denotes the roll-off factor of the filter.

Clearly, we will always have that the inequality:

$$R_c \leq BW \tag{2}$$

will be valid for any $\beta \geq 0$. In fact, as we will see in Section IV, it is possible to use the above upper-bound on the code rate to determine an appropriate lower-bound on the code length (which we will determine in the next subsection).

Now, we know that the chip rate must be an integer fraction of the FPGA clock frequency:

$$R_c = \frac{F_{\text{FPGA}}}{K}, \tag{3}$$

for some integer $K \in \mathbb{Z}$, which is the RRC interpolation ratio. Substituting the relationship of R_c into the bandwidth equation, we obtain:

$$BW = \frac{F_{\text{FPGA}}}{K}(1+\beta). \tag{4}$$

This equation has two degrees of freedom (K and β) and two already-determined constants (BW and F_{FPGA}). It is possible to create a table of possible choices for K and β and pick an appropriate choice that satisfies the constraints. This table is generated by sweeping K across the integers and solving for the appropriate β once we substitute $BW = 8$ MHz and the appropriate FPGA clock frequency F_{FPGA}.

It is important to note that a higher roll-off factor will allow the output signal to have a sharper histogram, thus allowing the design to allocate more power to the PN sequence than the carrier. A higher roll-off factor will reduce the tail of the filtered PN output distribution, thus allowing the gain to be increased further than in the case of a lower roll-off factor—as long as a wider transition band appears within the 8 MHz target bandwidth and can be tolerated.

B. Code Length

The driving requirement behind the code length is the fact that the code period must be at least 1 ms. Considering the code rate obtained in the previous section, the code length is related to the code period via the following equality:

$$2^N - 1 = T_c \cdot R_c, \tag{5}$$

where T_c denotes the code period in seconds, and $2^N - 1$ denotes the code length in chips. Given the requirement that $T_c > 10^{-3}\ s$, we have:

$$T_c = \frac{2^N - 1}{R_c} > 10^{-3} s. \tag{6}$$

It is possible to obtain a sufficient condition on the code period via the inequality $R_c \leq BW$:

$$T_c = \frac{2^N - 1}{R_c} \geq \frac{2^N - 1}{BW} = \frac{2^N - 1}{8\,\text{MHz}} > 10^{-3}\ s \tag{7}$$

which implies that:

$$N > 12.966 \tag{8}$$

where $N \in \mathbb{Z}$, since it represents the number of bits in the linear feedback shift register (LFSR). Thus, the above inequality is automatically satisfied when:

$$N \geq 13 \tag{9}$$

which implies that the code length is

$$2^N - 1 \geq 8191\,. \tag{10}$$

In this case, observe that the code length was completely determined by the bandwidth and the code period requirement of 1 ms, and does not explicitly depend on the code rate.

C. Primitive Polynomial Pairs

Given the code length obtained in Section B, we must obtain the actual PN sequence to be used in the Delta-DOR tone spreading. There are generally two PN sequences that are considered for such an application:

(1) Maximal length sequences (M-sequence)

(2) Gold codes

M-sequences have very good autocorrelation properties, meaning that the autocorrelation plot of the M-sequence against itself will yield a very high peak at the origin (in fact, the peak in the absence of noise will be the length of sequence) and very low off-center correlations. However, the M-sequence has a drawback in that it is not immediately very well suited for cross-correlation against *other* M-sequences (e.g., in the case of direct-sequence spread-spectrum multi-user applications). The Gold codes were introduced for the purpose of mitigating this drawback. These codes are generated by implementing an exclusive or (XOR) operation on very specifically picked M-sequence polynomials to produce up to $2^N - 1$ sequences that exhibit good cross-correlation performance against one another and good autocorrelation performance by themselves. However, the drawback is the added complexity to implement them (essentially the implementation of two M-sequences) and a slightly worse autocorrelation performance than M-sequences. To see this, we picked a Gold sequence with $N = 11$ as well as an M-sequence of the same length. The autocorrelation performance is shown in Figure 2. We observe that while both sequences exhibit good pseudorandom behavior, the M-sequence autocorrelation is slightly better behaved than that of the Gold sequence. However, we note that good cross-correlation performance is desired if different codes will be used for different nearby spacecraft (such as adjacent CubeSats).

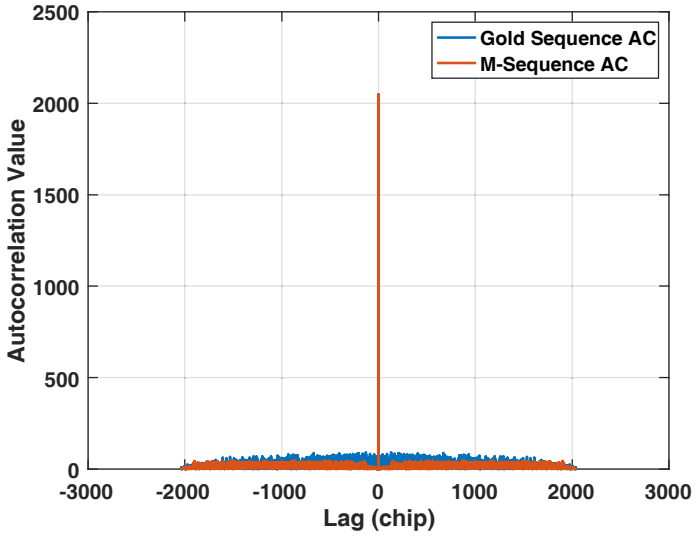

Figure 2. Autocorrelation performance of an M-sequence and a Gold sequence.

Nevertheless, it was decided that the implementation use a Gold sequence with $N \geq 13$ as outlined in Section II.B. The next task is to pick a primitive polynomial pair, i.e., the generating polynomial of two M-sequences that are specifically chosen according to the following theorem [5]:

Theorem

Let f_1 and f_t be a preferred pair of primitive polynomials of degree N whose corresponding shift registers generate maximal linear sequences of period $2^N - 1$ and whose cross-correlation function θ satisfies the inequality:

$$|\theta| \leq \begin{cases} 2^{(N+1)/2} + 1 & \text{for } N \text{ odd} \\ 2^{(N+2)/2} + 1 & \text{for } N \text{ even } N \neq mod\ 4 \end{cases} \tag{11}$$

Then the shift register corresponding to the product polynomial $f_1 \cdot f_t$ will generate $2^N + 1$ different sequences each period $2^N - 1$ and such that the cross-correlation function θ of any pair of such sequences satisfies the above inequality.

For our case, it is ideal to pick an odd $N \geq 13$ and test the first line of the correlation constraint above. A table of all possible M-sequence generating polynomials for any $N < 33$ can be found in [6] (it should be noted that reversing the order of the taps in any of the tables will result in another valid M-sequence generator as well, but those reversed taps are not listed in the table). By verifying the inequality in the theorem, a valid preferred polynomial pair for a given odd N can be found.

The structure of the Gold code generation is shown in Figure 3. The initial states are stored in the delay elements of the block diagram. The top PN sequence generator creates an M-sequence with one of the two primitive polynomials, $p(z)$, found through verifying the theorem, while the bottom PN sequence generator creates an M-sequence with the second

primitive polynomial, $q(z)$, found by verifying the theorem. The two sequences are then combined via an XOR operation. The resulting PN sequence is a Gold sequence due to the choice of $p(z)$ and $q(z)$. The final block converts 0/+1 binary PN sequence to a −1/+1 binary sequence.

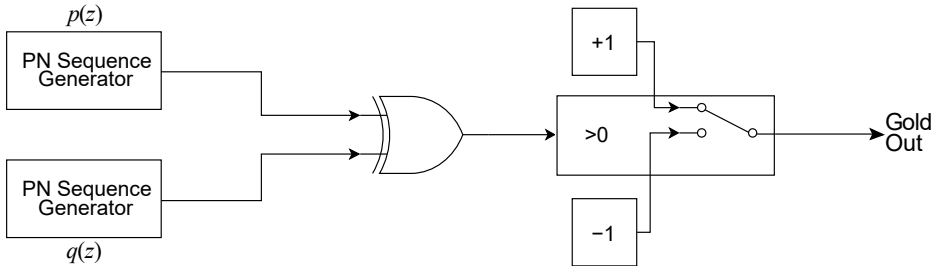

Figure 3. The Gold code generation block diagram.

D. Chip-Shaping Filter

The chip-shaping filter limits the bandwidth of the PN sequence, but increases the dynamic range of the time-domain signal. Based on the discussion in Section II.A, a roll-off factor that is related to the FPGA clock frequency and the samples per chip (K) is determined. The final parameter of the RRC filter is the filter span. For this, the requirement is to use a 16-chip span. Higher spans help in flattening the passband frequency response while suppressing the stop-band further. To see this, we consider 2-, 4-, 8-, and 16-chip span RRC filters in Figure 4. The penalty of using higher spans is that longer filters require higher complexity in the FPGA implementation (more multipliers).

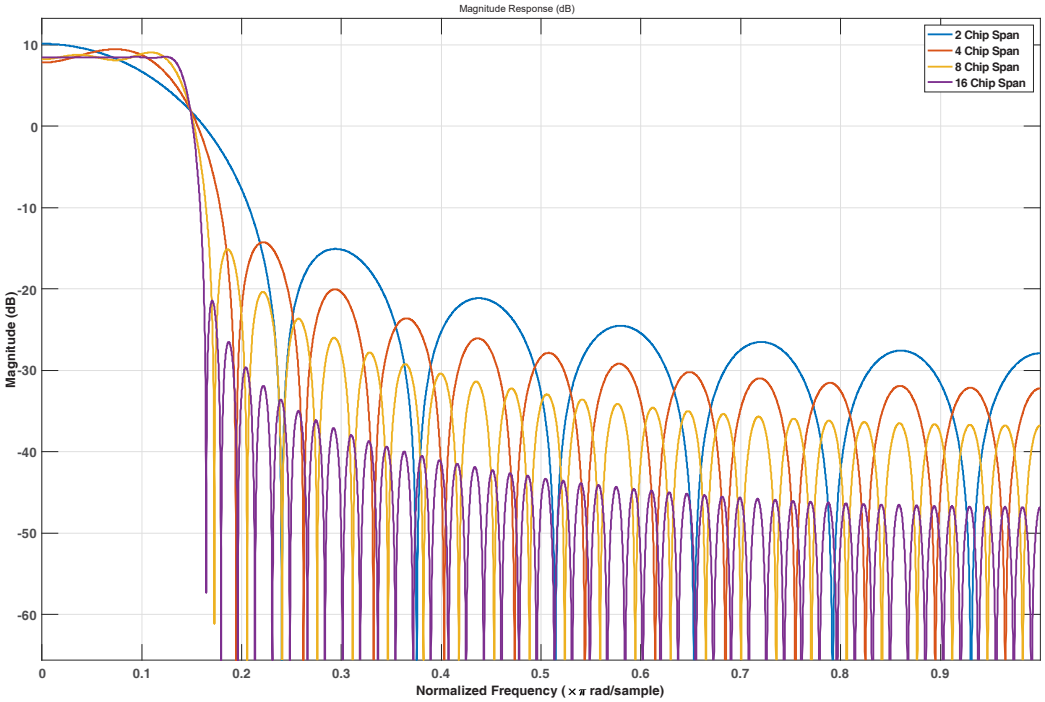

Figure 4. RRC filters with different spans. Higher spans suppress the stop-band further and flatten the passband.

III. X-band PN Delta-DOR Iris Implementation

A. Floating-point Simulation (MATLAB)

A floating-point model was generated to model the PN Delta-DOR generation process. A Gold code sequence was generated according to the discussion in Section II.C. An RRC filter was then created that spans 16 chips with the excess-bandwidth β (chosen in Section II.A) and an interpolation factor of K samples per chip. The output of the RRC filter is then modulated by a real sinusoid with a frequency of approximately 19 MHz, the traditional Delta-DOR sinusoidal subcarrier frequency [1]. Figure 5 shows the signal flow of the PN Delta-DOR generation subsystem. Step (B) represents the output of the code generation process shown in Figure 3. The output of that process is then fed into the RRC filter that interpolates the code by a factor of K, making the bus clock at the FPGA frequency (Step C). Finally, the output of the RRC filter is multiplied by the 19 MHz traditional Delta-DOR subcarrier (Step A) to yield the final PN Delta-DOR subsystem output (Step D). Figure 6 shows the expected output of each stage of this PN Delta-DOR generation process.

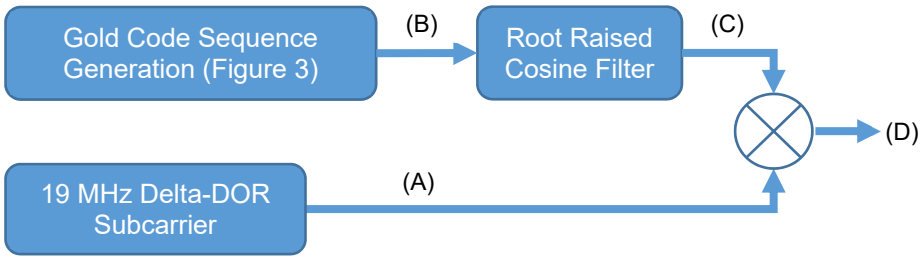

Figure 5. Signal flow of the PN Delta-DOR generation process. Step (A) is the 19 MHz Delta-DOR subcarrier that is traditionally generated for the classical Delta-DOR function. Step (B) is the Gold code sequence at the chip rate R_c, while Step (C) is the output of the RRC filter at the FPGA clock rate. Finally, Step (D) is the modulated output of the PN Delta-DOR generation system.

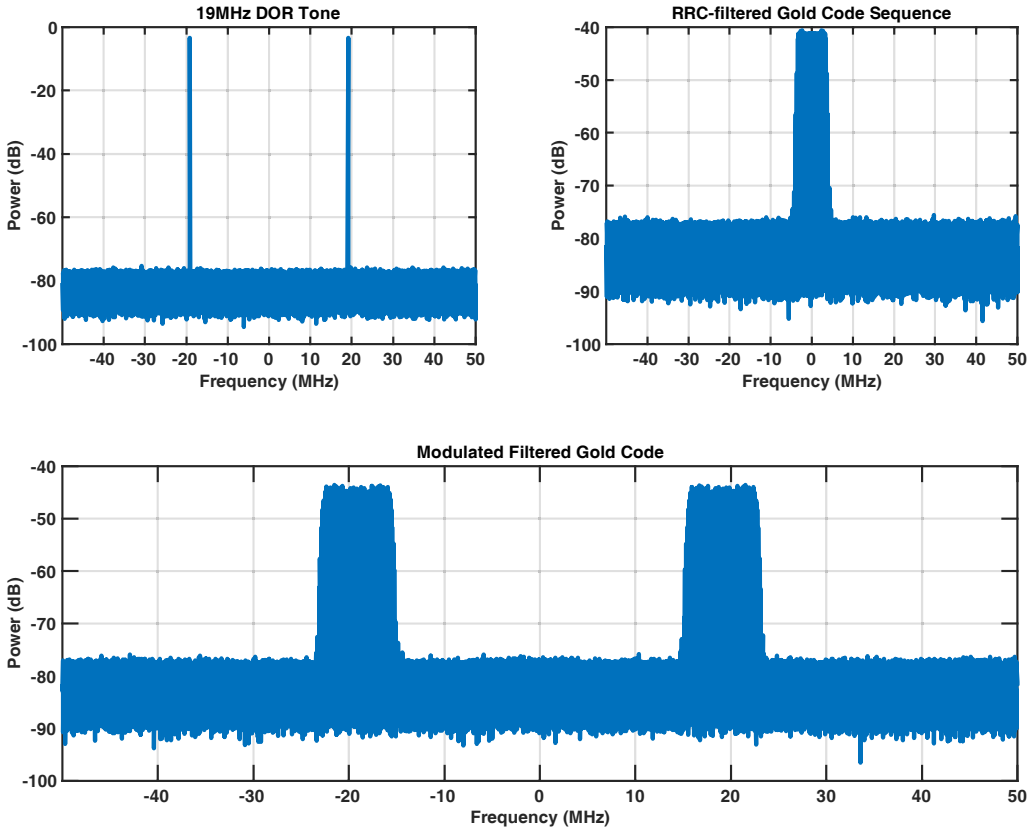

Figure 6. Top-left figure shows the frequency content of the 19 MHz Delta-DOR subcarrier (Step (A) in Figure 5). The top-right figure shows the frequency content of the interpolated Gold code (output of the RRC filter, Step (C) in Figure 5). The bottom figure shows the frequency content of the modulated output of the Gold code spread 19 MHz Delta-DOR subcarrier (Step (D) in Figure 5).

B. Fixed-point Simulation (Simulink) and HDL Code Generation

In terms of the implementation of the PN generator, the traditional 19 MHz subcarrier generation is assumed to be present in the radio hardware description language (HDL) code (such as Iris). However, the Gold code implementation as well as the RRC filter must be implemented in fixed point. This was done using Simulink and its built-in functions including Gold sequence generator and RRC transmit filter. Both of these blocks are compatible with Simulink's HDL coder which can generate Verilog/VHSIC Hardware Description Language (VHDL) code automatically from the Simulink block diagram. The Simulink fixed-point model is shown in Figure 7. The frequency response of the RRC filter is shown in Figure 8. The FPGA utilization for the PN Delta-DOR generation system (excluding DOR subcarrier modulation) is listed in Table 1. We note that the majority of required multipliers are due to the RRC transmit filter/interpolator.

Figure 9 shows the comparison of the floating-point model discussed in Section III.A and the fixed-point model developed in this section. The top-left plot shows the frequency-domain of the floating-point model developed in Section III.A, while the bottom-left plot shows the frequency-domain of the fixed-point model developed in this section. The difference between the outputs of the two models is shown in the frequency-domain of the

top-right plot. Finally, the bottom-right plot shows the difference in the model outputs superimposed on the output of the floating-point model (to show the relative difference in scales).

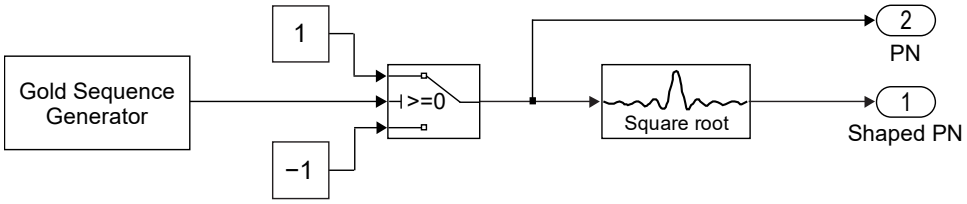

Figure 7. Simulink fixed-point model.

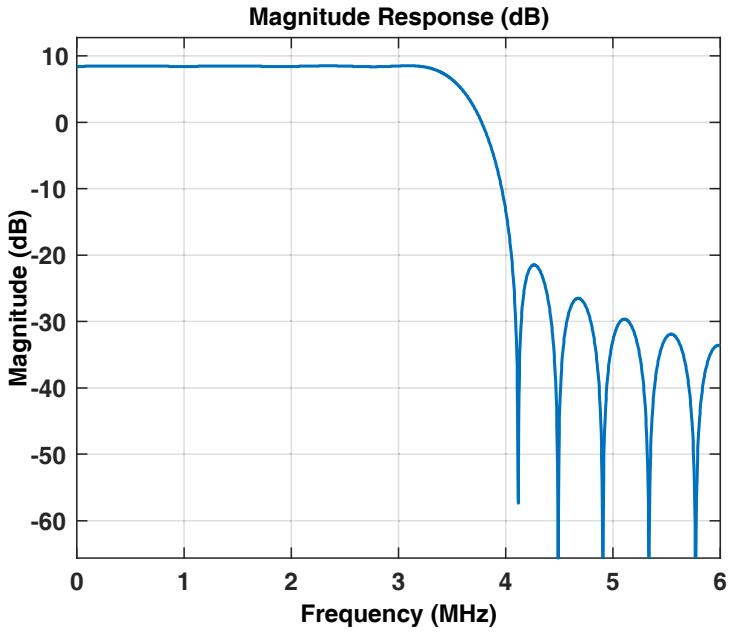

Figure 8. Frequency response for fixed-point RRC filter.

Table 1. Summary of FPGA utilization for PN Delta-DOR system.

Category	Count
Multipliers	17
Adders/Subtractors	16
Registers	19
Total 1-Bit Registers	67
Random Access Memory (RAM)	0
Multiplexers	19
I/O Bits	16
Static Shift Operators	0
Dynamic Shift Operators	0

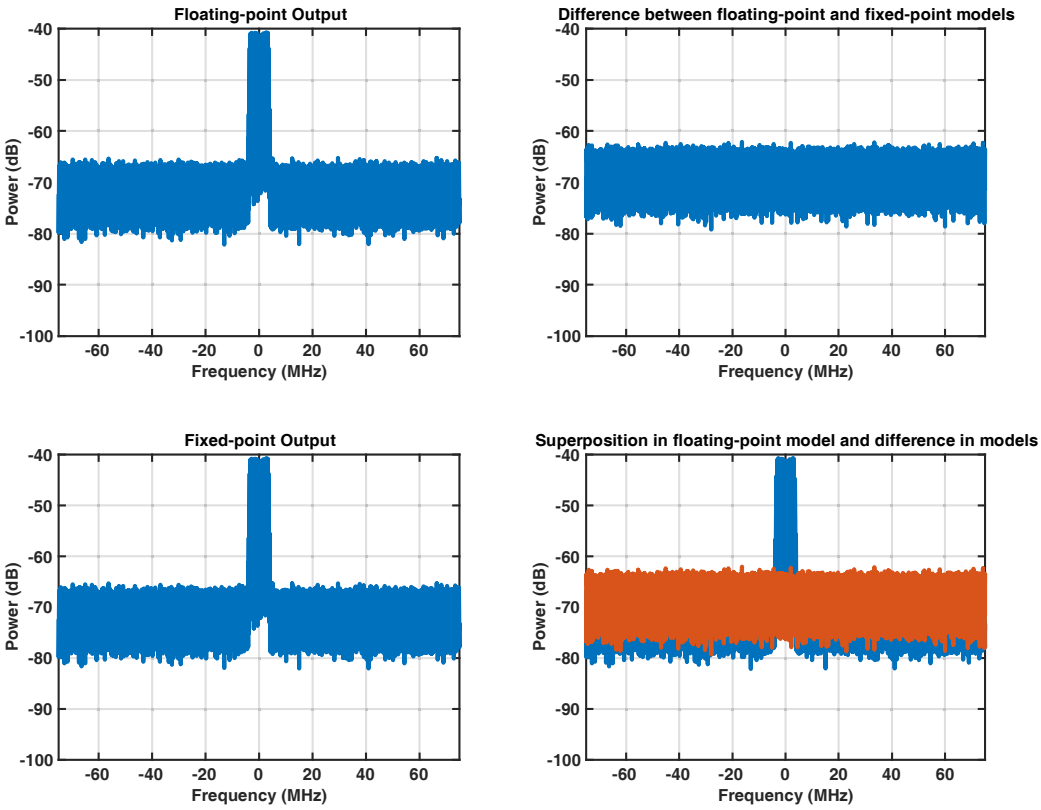

Figure 9. Comparison between the floating-point model and fixed-point model. Top-left is the frequency response of the floating-point model. The bottom-left is the frequency response of the fixed-point model. The top-right is the frequency response of the difference of the two models. The bottom-right is the difference superimposed on the floating-point model.

C. HDL Simulation (Vivado)

As new FPGA blocks were created using Simulink, the generated HDL was tested to verify that each new module meets the key requirements of the task using the Xilinx Integrated Synthesis Environment (ISE) simulation program Vivado. Whenever a new module or set of modules was generated, a separate test bench Verilog file was written to quickly test and

verify that the new block would output the proper signal that matched what was expected theoretically. This test bench file would make a call to the set of modules under test and save the key output signals to a text file. Finally, this text file output would be read by MATLAB scripts designed to analyze the signals of interest that are output by each new module.

In this way, each module could be tested in simulation on Vivado before being ported to the Iris prototype radio, ensuring that this new firmware would compile and provide the expected results on actual hardware. This proved to be an efficient verification method when the modulation index was optimized to make sure the optimal power was being supplied to the PN Delta-DOR tones and there was no saturation.

1. Modulation Index Optimization

A key goal of the HDL simulation is to confirm that no saturation (clipping) occurs at the data lines associated with the DOR tone generation blocks or subsequent HDL blocks associated with the Iris modulator. In addition to this, the power levels associated with the PN DOR signal must be set in a manner that is related to the carrier power. To do this, the HDL simulations were used to:

(1) Set the gains in the DOR generation path to guarantee no saturation.

(2) Maximize gains (under no saturation) so that the carrier power to PN Delta-DOR power is minimized.

Now, we note that it is easier to obtain higher power levels out of the classic DOR mode since a tone amplitude is easily bounded (i.e., the tone histogram has a sharp cut-off at the sinusoid amplitude). On the other hand, the RRC filter increases the dynamic range of the shaped PN Delta-DOR signal (i.e., the RRC filter creates a signal with a longer histogram tail). This is shown in Figure 10 for the classic Delta-DOR mode (specifically, the traditional 19 MHz Delta-DOR subcarrier). Observe that the histogram represents the probability distribution of a sinusoid $x = \sqrt{P}\cos(\theta)$ with random phase θ, where $P = 1/2$ for Figure 10:

$$f_x(x) = \frac{1}{\pi\sqrt{P}} \cdot \frac{1}{\sqrt{1 - \frac{x^2}{P}}} I_{|x| \le \sqrt{P}}(x) \tag{12}$$

where

$$I_{|x| \le \sqrt{P}}(x) = \begin{cases} 1, & |x| \le \sqrt{P} \\ 0, & \text{otherwise} \end{cases} \tag{13}$$

represents the indicator function for $|x| \le \sqrt{P}$. Clearly, there is a sharp drop-off in the probability at $|x| = \sqrt{P}$. On the other hand, Figure 11 shows the histogram of the shaped PN Delta-DOR. Observe that for this case, the histogram shows a longer tail. In fact, the complementary cumulative distribution function (CCDF) plotted in the top portion of Figure 11 shows that the probability that a sample exceeds a value of 1.984 is

approximately 0.4302 percent, which is low. This was done by optimizing the number of bits and the gains in the PN Delta-DOR generation path and visualizing the CCDF at each stage to make sure that the probability of exceeding the maximum realizable value at a stage is very low (less than 1 percent), as was done in Figure 11.

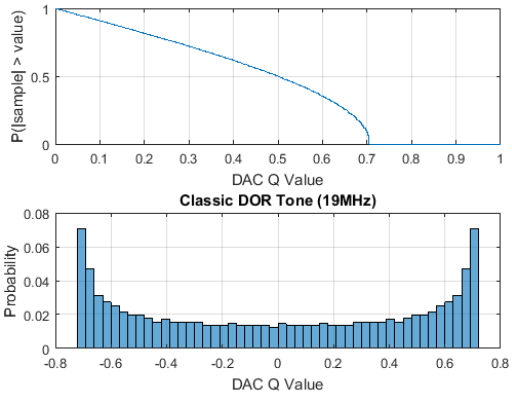

Figure 10. Histogram of classic DOR 19 MHz subcarrier.

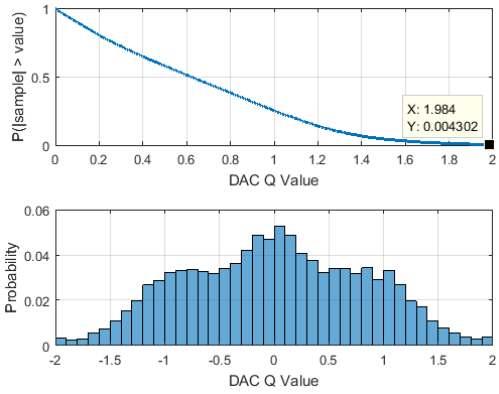

Figure 11. Histogram of shaped PN Delta-DOR signal.

Table 2 lists the power levels measured using a spectrum analyzer. It shows that the carrier power to PN Delta-DOR power is approximately –1.96 dB (this is with respect to both sides of the spectrum—i.e., it is the total power assigned to the PN Delta-DOR signal).

Table 2. The power levels measured using spectrum analyzer obtained through final implementation.

Carrier Power (P_c)	PN Delta-DOR Channel Power (P_d)	Total Transmit Power (P_T)	$\frac{P_c}{P_T}$	$\frac{P_d}{P_T}$	$\frac{P_c}{P_d}$
4.00 dBm	1.10 dBm	5.96 dBm	–1.96 dB	–4.86 dB	2.9 dB

Now, to obtain the modulation-index, which is a commonly used metric for referring to the carrier power relationship to the Delta-DOR signal power, we recall the following definitions from the classical expansion of a phase-modulated signal [1]:

$$\sqrt{P_T}\cos\left(2\pi f_c t + m\sin\left(2\pi f_d t\right)\right) \approx \sqrt{P_T}J_0(m)\cos\left(2\pi f_c t\right) - \sqrt{P_T}J_1(m)\sin\left(2\pi f_d t\right) \quad (14)$$

where P_T represents the total power, m represents the modulation index (measured in radians), f_c denotes the carrier frequency, and f_d denotes the Delta-DOR frequency. Finally, $J_0(z)$ and $J_1(z)$ represent the Bessel functions of the first kind. This equation applies to classical Delta-DOR signals (not a PN code Delta-DOR signal), but it allows us to relate the classical description of the modulation index (in radians) to the total carrier power and the PN channel power. From the above approximation, we obtain the following relationships for the carrier power to Delta-DOR signal power for the classic scenario (ignoring the 1 MHz subcarrier):

$$\frac{P_c}{P_T} = J_0(m)^2 \quad (15)$$

$$\frac{P_d}{P_T} = J_1(m)^2 \tag{16}$$

where P_c denotes the total power of the carrier and P_d denotes the total power of the classic Delta-DOR subcarrier. Thus, we obtain that the ratio of the carrier power to classic Delta-DOR subcarrier power is given by:

$$\frac{P_c}{P_d} = \frac{J_0(m)^2}{J_1(m)^2} \tag{17}$$

Now, it is possible to plot this equation for different values of m versus the decibel level of P_c / P_d , as shown in Figure 12. We observe that the value of m that corresponds to the $\frac{P_c}{P_d} = 2.9\,\text{dB},$ listed in Table 2, is approximately $m \approx 1.170$ rad = 67°. This is a relatively high modulation index (which assigns a relatively large amount of power to the Delta-DOR signal), and is deemed satisfactory. Again, the expansion above is only used to relate the relative powers of the carrier and the PN channel to the traditional modulation index value used in the classical Delta-DOR scenario.

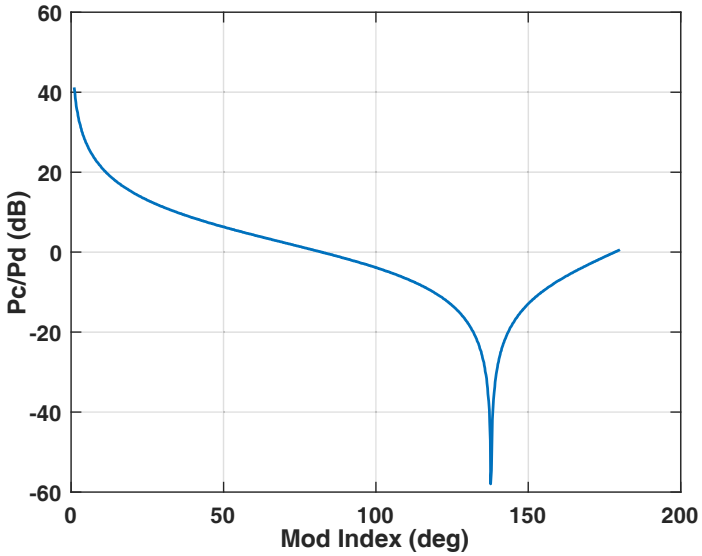

Figure 12. Carrier power over classic Delta-DOR power against modulation index m. Observe that a value of 2.9 dB corresponds to a modulation index of 67°.

The difficulty in improving the modulation index is the fact that the current configuration was obtained while making sure that the signal is not saturated. The alternative would be to decrease the carrier power so that the ratio of carrier to PN Delta-DOR signal power is improved, but this would also decrease the total transmit power if the power amplifier is not re-biased. We note that there are additional ways to improve the output power of the PN Delta-DOR signal without modifying the carrier power. This involves modification of the RRC pulse-shaping filter in order to increase the roll-off factor β. To illustrate the fact that large roll-off factors of the RRC allow us to reduce the dynamic range of the shaped PN Delta-DOR signal, we illustrate the histogram of the shaped PN Delta-DOR signal

generated with a larger roll-off factor. The histogram tail is much shorter than that shown in Figure 11 with the smaller β. Again, such a choice of roll-off factor would cause the spectrum to contain wider transition bands.

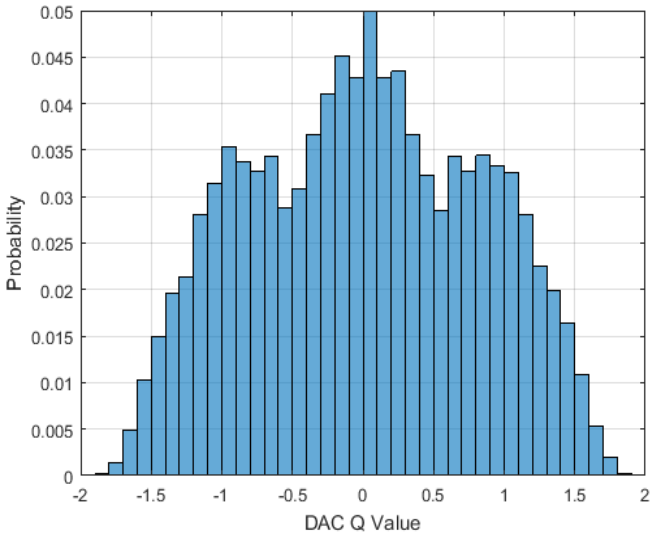

Figure 13. Histogram of output shaped PN Delta-DOR signal with a larger β.

The next section discusses undesired spectrum shaping caused by the analog circuitry of the prototype radio and a method devised to overcome it.

D. Iris Prototype SN104 Implementation: The Need for (Pre-) Equalization

Once the design was verified through the HDL simulator, it was implemented onto the Iris prototype unit. However, the output exhibited frequency shaping as shown in Figure 14.

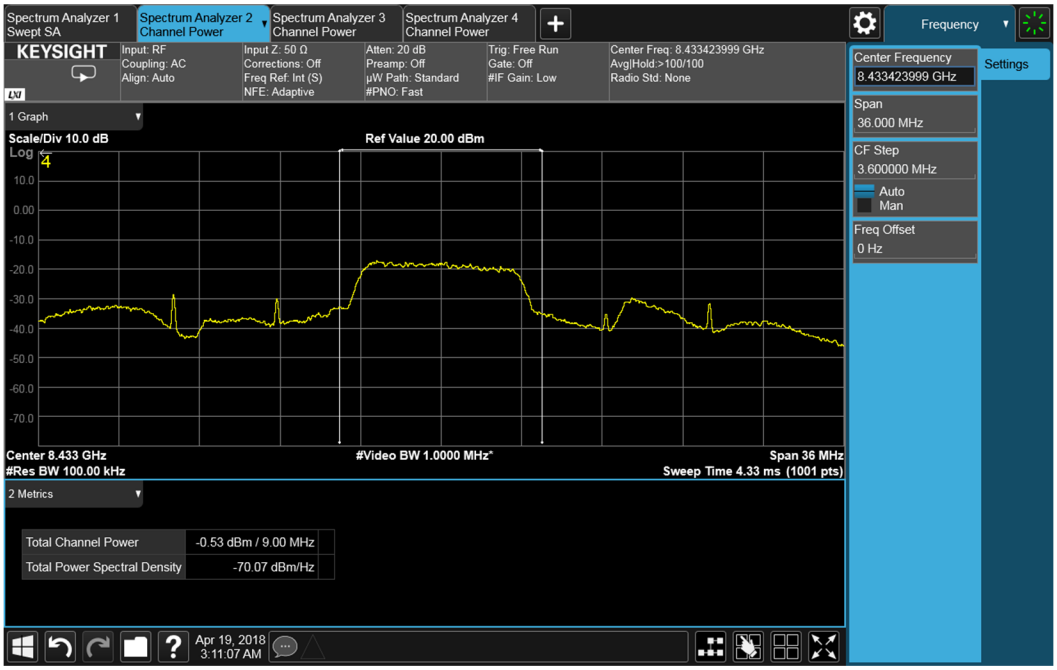

Figure 14. Zoomed-in view of the uncompensated PN Delta-DOR output.

The frequency shaping indicates the presence of a low-pass filter (LPF) after the digital-to-analog converter (DAC). This determination is made because the HDL simulation indicated a flat spectrum for the shaped PN sequence output whereas the implementation on the Iris unit yielded a shaped output. The next section discusses the design and implementation of a pre-equalization/pre-distortion filter before the DAC in an attempt to flatten the spectrum at the output of the SN104 unit. It was determined that the analog Chebyshev low-pass baseband filter that is used on the Iris exciter prior to the in-phase/quadrature-phase (I/Q) mixer is causing the frequency shaping, as indicated in the flowchart in Figure 15.

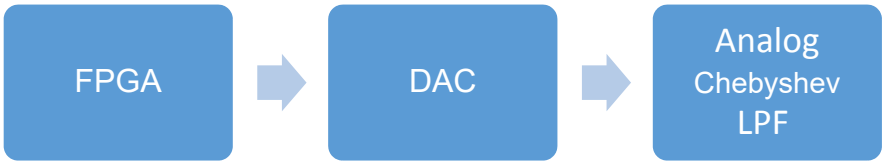

Figure 15. Iris radio flowchart. The output of the digital board is synthesized as an analog signal via a DAC and later is filtered by a Chebyshev LPF.

E. Equalization/Pre-distortion Filter Design and Implementation

The LPF response of the analog filter begins to decay around 20 MHz from the center frequency. This was reasonable for Iris telecom because the bandwidths required for the current generation of Iris are generally very low (much lower than 20 MHz). In addition, when the classic Delta-DOR mode was added to Iris, a single traditional 19 MHz tone was transmitted at the high frequency. This 19 MHz tone *did* experience shaping by the filter. However, due to its low bandwidth, it only experienced an attenuation that was compensated for by choosing the appropriate modulation index. For these reasons, there was no need to compensate for the effect of this analog LPF. In future generations of Iris, a wider filter will be used instead of this one in order to allow transmissions with higher data rates.

Now, the S-parameters of the filter were obtained, and an equivalent digital LPF was synthesized. The design of the pre-distortion/pre-equalization filter, was determined by the desire for the cascade filter (our pre-distortion filter cascaded with the analog Chebyshev filter) to have a linear phase response over the passband:

$$H(j\Omega) = F_1(j\Omega) \cdot F_2(j\Omega) \approx e^{-jD\Omega} \tag{18}$$

where D is a desired group delay, $F_1(j\Omega)$ represents the pre-distortion filter frequency response, and $F_2(j\Omega)$ represents the analog Chebyshev filter frequency response. Thus, we obtain:

$$F_1(j\Omega) \approx \frac{e^{-jD\Omega}}{F_2(j\Omega)} \tag{19}$$

A value of $D = 70$ samples were chosen for the design, and the pre-distortion filter was found to be an finite-impulse-response (FIR) filter of length 103 that was designed using the least-squares criterion. Figure 16 shows the cascade filter, $H(j\Omega)$, response. We observe that the cascade filter response is very flat up to a frequency of 0.6π rad. Finally, we implemented this design through the Simulink HDL toolbox as shown in Figure 17, where the coefficients were stored in two's complement format with 15 fractional bits and 1 sign bit. This filter was implemented at the modulator module before the final filters prior to the DAC. Observe that there would now be a ramp-up in the PN Delta-DOR signal power out of the DAC converter as a function of increasing frequency. This is due to the compensation filter. Figure 18 shows the zoomed-in view of the PN Delta-DOR signal with compensation for the Chebyshev filter (i.e., using the pre-distortion filter) as obtained by the same spectrum analyzer. We observe a clear flattening of the spectrum once the pre-distortion filter is utilized, which suggests proper compensation for the shaping performed by the Chebyshev filter.

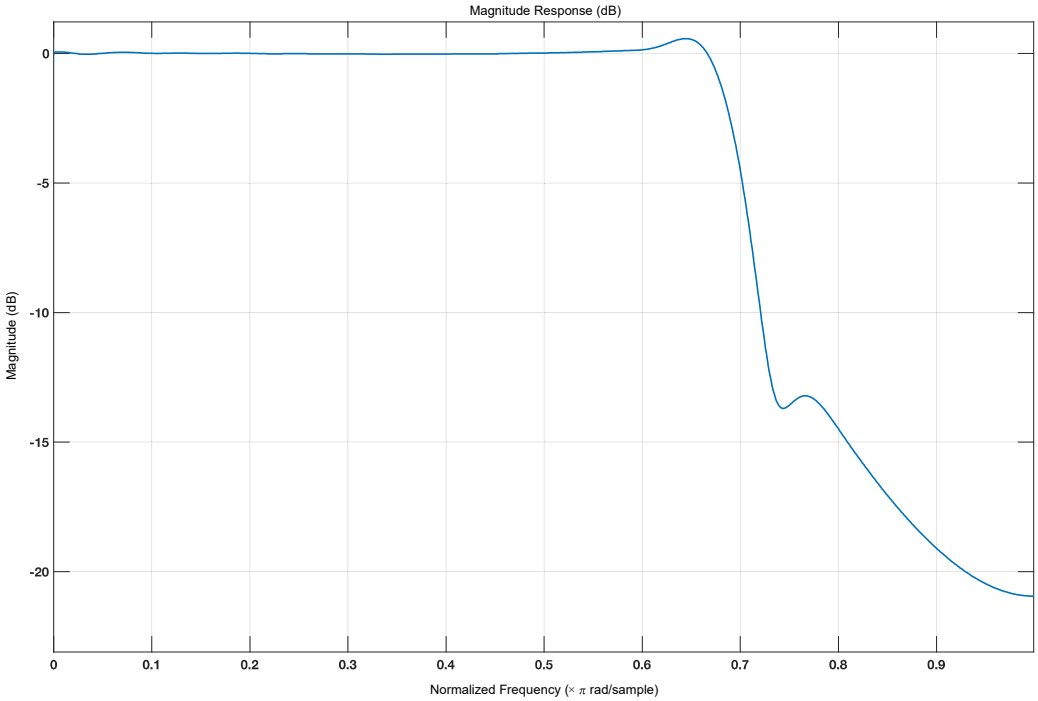

Figure 16. Cascade filter response.

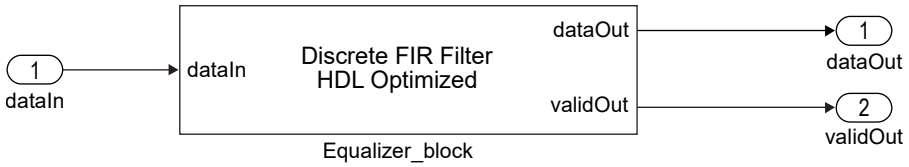

Figure 17. Simulink implementation of pre-distortion filter.

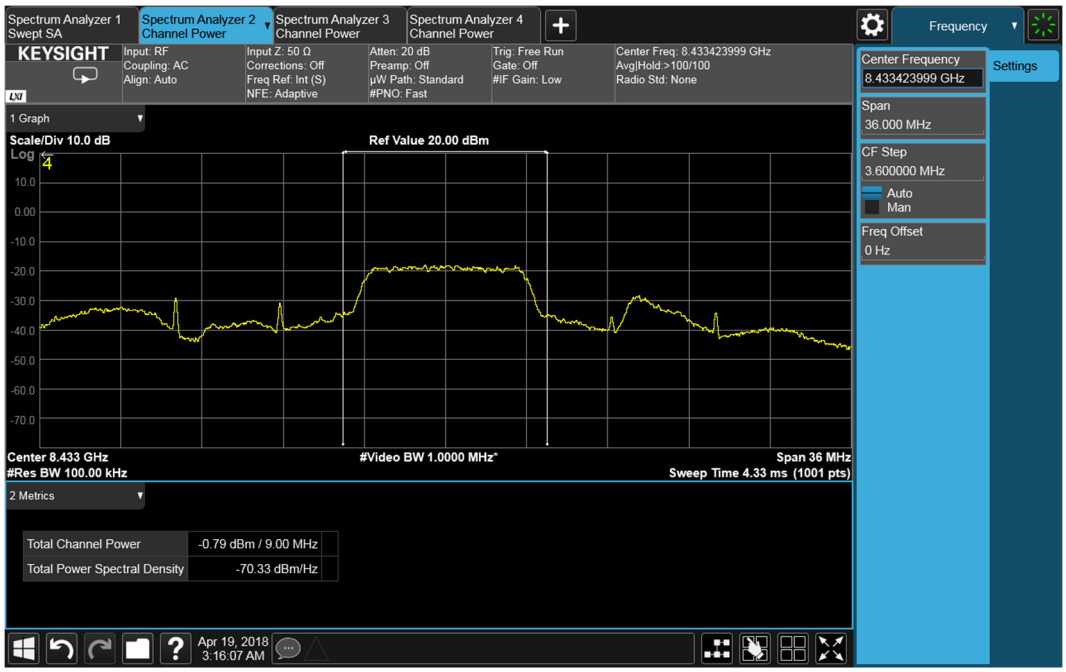

Figure 18. Zoomed-in view of the compensated (with pre-distortion filter) PN Delta-DOR output.

F. Time Domain Verification of Iris Output

Results shown in Figure 18 allow for the verification of the output spectrum from the Iris radio. However, the verification of the actual code structure (i.e., the demodulation of the output signal of the Iris radio) is still necessary.

To accomplish this, we used a Keysight spectrum analyzer to capture I/Q samples of the X-band output of the Iris prototype unit. The constellation of the received symbols, obtained without performing any frequency recovery or sampling time recovery is shown in Figure 19. In this figure, a constant carrier frequency is estimated and the data are frequency shifted accordingly, the carrier removed, and the constellation de-rotated so that the PN code resides mainly on the in-phase component of the result.

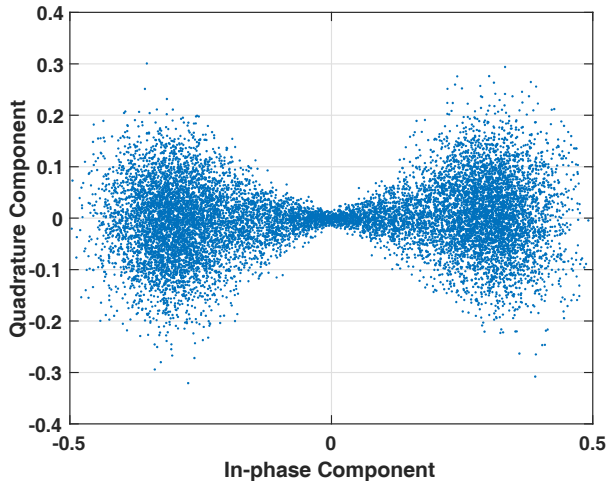

Figure 19. Received symbols obtained without performing time or frequency recovery. Only coarse carrier phase acquisition and telemetry extraction is performed.

Performing hard thresholding on the result and correlating against the true Gold code, we obtain Figure 20. Observe that the maximum correlation is less than $2^N - 1$ (the maximum correlation value, which is the code length). This implies that bit errors are made during the hard-thresholding process. In addition, the correlation strength decreases with time. This effect indicates slow frequency and timing drifts in the telemetry that must be compensated for since more errors are made as time progresses, which is a sign of imperfect frequency and timing compensation at the ground station. The first and last peaks are lower than the others because they contain only partial (truncated) sequences.

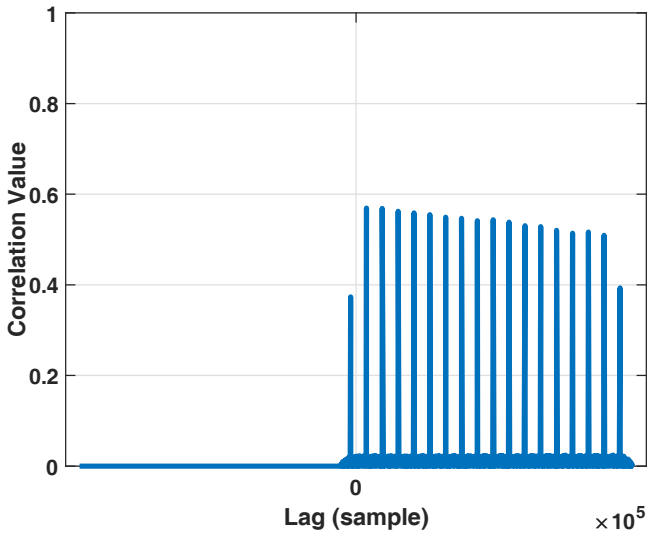

Figure 20. Normalized correlation of uncompensated received samples against true Gold code. The non-maximal correlation values indicate that bit errors were made in the hard-thresholding operation. The reason for this is that the frequency and timing compensation performed at the ground station for this figure was not adequate to remove the frequency shifts and symbols sample time slips when obtaining soft chips.

To counteract the timing and frequency drift effects, we implemented the basic radio blocks shown in Figure 21 in order to correct for any frequency offsets, timing offsets, and finally demodulation:

Figure 21. Software-receiver architecture for demodulation of recorded Iris output.

This receive chain was implemented in MATLAB using tools from the communication toolbox. The phase estimates from the phase-locked loop (PLL) as well as the soft symbols are shown in Figure 22. We observe that the soft symbols are generally far away from the threshold (zero) and thus are not likely to yield bit errors once hard decisions are made.

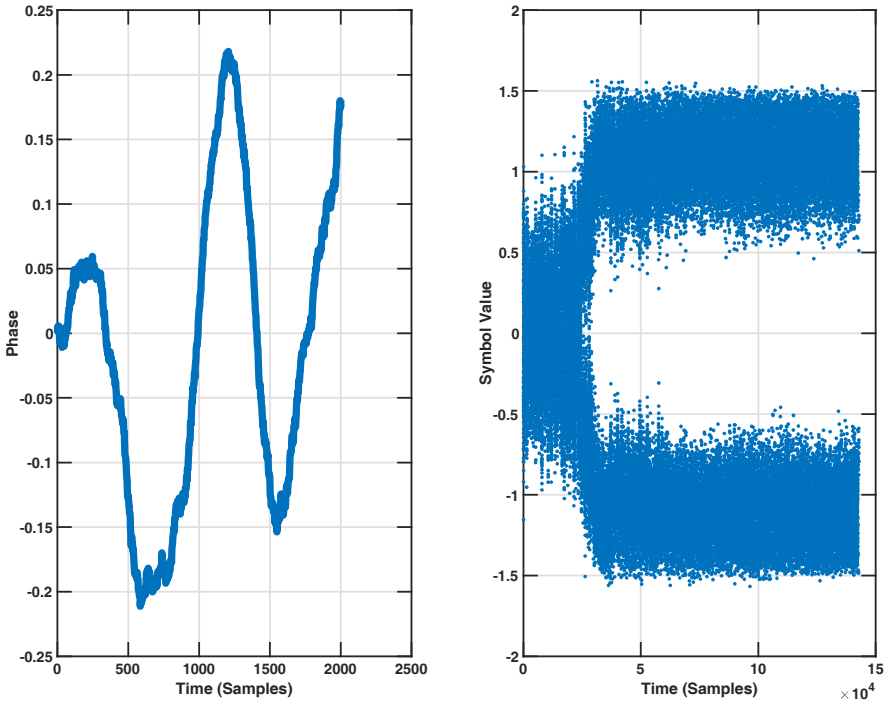

Figure 22. Recovered phase and soft-symbols from recorded data.

The correlation of the hard decision bits from the soft-symbol output shown in Figure 22 against the true Gold code is shown in Figure 23. We observe that the maximum correlation value for all the peaks except the truncated ones is exactly $2^N - 1$ (the length of the code) and thus the signal was correctly demodulated to confirm the presence of the valid code. We stress, however, that our verification procedure for the Iris implementation is not intended to be a reference Deep Space Network (DSN) implementation, which is likely to consider many other effects in the Delta-DOR calculations.

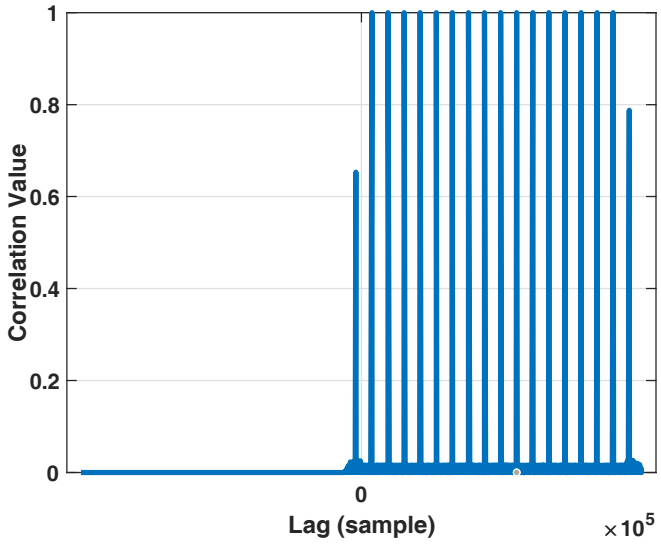

Figure 23. Normalized correlation of time- and frequency-compensated received samples against true Gold code.

IV. Ka-band Design Overview

Recall that the bandwidth relationship to the chip-rate and roll-off factor is given by:

$$BW = R_c(1+\beta) \tag{20}$$

where R_c denotes the chip-rate of the PN sequence and β denotes the roll-off factor of the RRC filter. Rearranging the equation, we obtain:

$$R_c = \frac{BW}{1+\beta} \leq BW \tag{21}$$

where the inequality is due to $\beta \geq 0$. Although the development of a complete Ka-band design requires knowledge of the FPGA frequency in order to determine the roll-off factor and samples per symbols (K) via the equation below, we may obtain an upper bound on the code length.

$$BW = \frac{F_{\text{FPGA}}}{K}(1+\beta) \tag{22}$$

The driving requirement behind the code length is the fact that the code period must be at least 1 ms. Considering the upper bound on the code rate obtained above (code rate is upper bounded by the desired bandwidth), the code length would be related to the code period via the following equality:

$$2^N - 1 = T_c \cdot R_c \leq T_c \cdot BW \tag{23}$$

where T_c denotes the code period in seconds, and $2^N - 1$ denotes the code length in chips. Re-arranging the equation, we obtain:

$$T_c = \frac{2^N - 1}{R_c} \geq \frac{2^N - 1}{BW} \tag{24}$$

Using the constraint that the code period must be at least 1 ms, we can enforce the lower bound on the code-period to be at least 1 ms:

$$T_c \geq \frac{2^N - 1}{BW} > 10^{-3} s \tag{25}$$

where the bandwidth for the Ka-band design is required to be 32 MHz. We can now obtain a sufficient condition on the code length by solving the above inequality for N:

$$N > \log_2(10^{-3} \cdot BW + 1) = \log_2(10^{-3} \cdot 32 \cdot 10^6 + 1) = 14.966 \tag{26}$$

where $N \in \mathbb{Z}$, since it represents the number of bits in the LFSR. Thus, a sufficient condition on N is:

$$N \geq 15 \tag{27}$$

which implies that the code length is

$$2^N - 1 \geq 32767 \tag{28}$$

which has a period of at least $T_c = \frac{2^N - 1}{R_c} \geq \frac{2^N - 1}{BW} \approx 1.024$ ms. It is possible to determine a primitive polynomial pair that satisfy the theorem listed in Section II.C for an appropriate choice of N. What remains is the determination of the pair (K, β) as was done in Section II.A. However, this determination requires the knowledge of the exact FPGA frequency on which the design will be implemented.

V. Future Work and Conclusions

The following tasks will be considered now that the PN Delta-DOR mode has been successfully demonstrated on an Iris radio prototype. First, we plan on implementing the X-band PN Delta-DOR on the Universal Space Transponder (UST). Active discussion is in the work with the UST team in order to implement the PN Delta-DOR mode on the UST prototype radio. One characteristic of the UST that may cause the design to differ from that of Iris is the FPGA frequency.

Second, we hope to infuse the X-band PN Delta-DOR mode onto one or two of the Iris EM-1 radios flight software. This will provide an opportunity to demonstrate the PN Delta-DOR mode in flight in the near future. A proposal is being prepared to consider infusing the PN Delta-DOR mode into the two EM-1missions: Lunar Flashlight and NEAScout. We hope that such infusion will allow us to perform more rigorous testing of the PN Delta-DOR subsystem, including possibly through the use of the Block V receiver.

Finally, we plan on extending the X-band design into the Ka-band. Requirements are being prepared for the Ka-band version of the PN Delta-DOR mode. Next, a suitable Gold code and an appropriate RRC filter will be tailored to the Iris and/or UST proposed Ka-band designs. These codes and RRC filters can be obtained by following the design methodology outlined in Section IV.

Acknowledgments

We would like to graciously thank Faramaz Davarian (member of the Communication Technologies and Standards Program office) for providing funding for this effort. Faramaz also provided many valuable suggestions through discussions that helped improve this work and article. We also would like to thank the Iris team, including Brandon Burgett, Sarah Holmes, and Lauren McNally for making an Iris prototype available for testing and evaluations.

References

[1] Delta-DOR—Technical Characteristics and Performance, CCSDS Green Book 500.1-G-1, May 2013.

[2] M. M. Kobayashi, T. Dobreva, E. Satorius, S. Holmes, A. Yarlagadda, F. Aguirre, M. Chase, K. Angkasa, B. Burgett, and L. McNally, "The Iris Deep-Space Transponder for the SLS EM-1 Secondary Payloads," Submitted for publication.

[3] M. M. Kobayashi, "Iris Deep-Space Transponder for SLS EM-1 CubeSat Missions," in *2017 Small Satellite Conference*, Logan, Utah, pp.1–7, August 2017.

[4] B. P. Lathi, Z. Ding, *Modern Digital and Analog Communication Systems*, 4th edition, Oxford University Press, p. 674, January 23, 2009.

[5] R. Gold, "Optimal binary sequences for spread spectrum multiplexing (Corresp.)," in *IEEE Transactions on Information Theory*, vol. 13, no. 4, pp. 619–621, October 1967.

[6] New Wave Instruments, Linear Feedback Shift Registers, http://www.newwaveinstruments.com/resources/articles/m_sequence_linear_feedback_shift_register_lfsr.htm, Retrieved May 10, 2018.