ARL-TR-10353 ● May 2026

Aliased Sinc Bandlimited Interpolation and Digital Modulation

by Patrick Jungwirth and W. Michael Crowe

NOTICES

Disclaimers

ARL-TR-10353 ● MAY 2026

Aliased Sinc Bandlimited Interpolation and Digital Modulation

Patrick Jungwirth
DEVCOM Army Research Laboratory

W. Michael Crowe
DEVCOM Aviation and Missile Center

REPORT DOCUMENTATION PAGE

1. REPORT DATE	2. REPORT TYPE	3. DATES COVERED START DATE	3. DATES COVERED END DATE
May 2026	Technical Report	9/24/2024	03/02/2026

4. TITLE AND SUBTITLE
Aliased Sinc Bandlimited Interpolation and Digital Modulation

5a. CONTRACT NUMBER	5b. GRANT NUMBER	5c. PROGRAM ELEMENT NUMBER

5d. PROJECT NUMBER	5e. TASK NUMBER	5f. WORK UNIT NUMBER

6. AUTHOR(S)
Patrick Jungwirth and W. Michael Crowe

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)	8. PERFORMING ORGANIZATION REPORT NUMBER
DEVCOM Army Research Laboratory and DEVCOM Aviation and Missile Center ATTN: FCDD-RLA-NC ATTN: FCDD-AMT-MGP-EHS Aberdeen Proving Ground, MD 21005 Redstone Arsenal, AL 35805	ARL-TR-10353

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S)	11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
DISTRIBUTION STATEMENT A. Approved for public release: distribution unlimited.

13. SUPPLEMENTARY NOTES

14. ABSTRACT

Aliased sinc (asinc) interpolation and digital modulation is based on the bandlimited asinc interpolation kernel. The progressive asinc reconstruction (PAR) algorithm was originally developed to perform real-time reconstruction of continuous-time level-crossing samples. Applying the PAR algorithm to quadrature amplitude modulation (QAM) as time-domain pulse shaping results in a small, controlled amount of signal overshoot and a small, controlled amount of intersymbol interference. Numerical simulations indicate that for 16 QAM, asinc interpolation requires 24% less bandwidth and significantly improves sidelobes compared to Feher, half-cycle raised-cosine interpolation. PAR interpolation for digital modulation is currently patent pending.

Feher–QAM modulation replaces each QAM symbol with a half-cycle raised-cosine function. All symbol waveforms begin and end with zero slope. These zero-slope transitions help improve intersymbol interference and reduce timing jitter problems, but they also introduce a rectangular windowing effect, resulting in sidelobes. While the Feher–QAM sidelobes are significantly lower than those of rectangular time-domain pulse shaping, they are substantially higher than those resulting from the application of PAR. The new asinc interpolation offers a substantial improvement over Feher interpolation.

15. SUBJECT TERMS
Network, Cyber, and Computational Sciences; Electromagnetic Spectrum Sciences; data communications; modulation; continuous time digital signal processing; aliased sinc; progressive asinc reconstruction; quadrature amplitude modulation

16. SECURITY CLASSIFICATION OF: a. REPORT	b. ABSTRACT	C. THIS PAGE	17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	UU	56

19a. NAME OF RESPONSIBLE PERSON	19b. PHONE NUMBER *(Include area code)*
Patrick Jungwirth	(520) 691-6454

STANDARD FORM 298 (REV. 5/2020)
Prescribed by ANSI Std. Z39.18

Contents

List of Figures

Acknowledgments

We would like to acknowledge the continuous time–digital signal processing (CT-DSP) foundational research from Dr. Yannis Tsividis, Edwin Howard Armstrong Professor of Electrical Engineering, and colleagues at Columbia University. We would also like to thank the DEVCOM Aviation and Missile Center and ARL for the opportunity to research CT-DSP.

1. Aliased Sinc Bandlimited Interpolation and Digital Modulation Introduction

Modulation schemes like amplitude shift keying (ASK) and quadrature amplitude modulation (QAM) convert a symbol stream to a continuous-time (CT) output signal. Conventional modulation methods ignore the bandwidth issues caused by the rapid waveform change from symbol n to symbol $n + 1$. Interpolation or modulation methods, like Feher modulation, slowly change the output waveform over one full symbol time. Aliased sinc (asinc) bandlimited interpolation and digital modulation also uses a full symbol time for the output waveform to change from symbol n to symbol $n + 1$. It also provides a substantial improvement (reduction) in bandwidth. The new modulation method takes advantage of asinc signal reconstruction to create a smooth bandlimited waveform with a true low-pass filter frequency response.

Asinc interpolation was originally developed to interpolate level-crossing sampling (LCS) data.[1,2] Sections 1.1 and 1.2 present an introduction to asinc interpolation and how it is applied for digital modulation. In Section 2, an introduction to level-crossing sampling and CT-digital signal processing (CT-DSP) is presented.

Section 3 reviews the sampling theorem. The input data stream is sampled for asinc interpolation. A short review of QAM is found in Section 4. QAM is used to compare asinc interpolation to half-cycle raised-cosine interpolation. A review of Feher, half-cycle raised-cosine, interpolation is covered in Section 5. Section 6 focuses on asinc interpolation to implement aliased sinc bandlimited interpolation and digital modulation. Asinc interpolation performance is compared to half-cycle raised-cosine interpolation. For 16 QAM, asinc interpolation uses 24% less bandwidth than half-cycle raised-cosine interpolation.

1.1 Periodic Waveform Sampling and Reconstruction

As shown in Figure 1, the symbol stream can be represented as a sequence of sample values. The sample values are uniformly spaced impulse functions with the impulse weights set equal to the symbol values. The operation to convert these impulses to a continuous baseband signal is the same as Whittaker–Kotel'nikov–Shannon (WKS) reconstruction of a sampled signal. Any useful reconstruction operator must match the time and amplitude of each impulse and generate a bandlimited signal. WKS reconstruction satisfies these conditions by convolving the sampled signal with the cardinal sine (sinc) kernel. However, the sinc kernel does not converge quickly, necessitating many calculations for each reconstructed point.

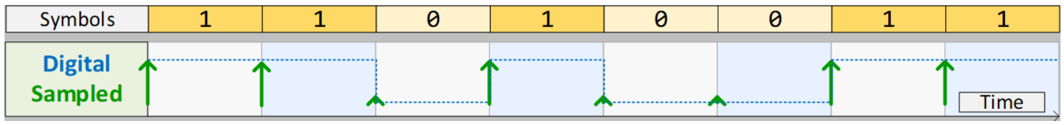


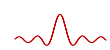
Figure 1. Sampled symbol stream.

For a uniformly sampled representation of a bandlimited, periodic waveform, the ideal reconstruction kernel is the asinc (or periodic sinc) kernel.[1,2] For a frame whose width is equal to the assumed period of the signal, T_p, the reconstruction of the center point consists of the convolution of the asinc kernel with the samples within that frame. Thus, the number of terms is finite.

The asinc kernel is defined by the sampling interval, T_s, and the period of the input waveform, T_p. The asinc kernel has an N-odd equation form and N-even form. Equation 1 gives the kernel where N is odd, which is the simpler and more common version. Equation 2 defines the kernel for the case where N is even. This form is a bit more complex and not as often discussed.

$$h(t) = \left(\frac{1}{N}\right) \frac{\sin(\pi\, t/T_s)}{\sin(\pi\, t/T_p)} \qquad (1)$$

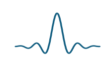
T_s = sampling time
T_p = periodic waveform (frame) time
$T_p = N\, T_s$



Asinc for N is odd

$$h(t) = \left(\frac{1}{N}\right) \cos(\pi\, t/T_p)\, \frac{\sin(\pi\, t/T_s)}{\sin(\pi\, t/T_p)} \qquad (2)$$

T_s = sampling time
T_p = periodic waveform (frame) time
$T_p = N\, T_s$



Asinc for N is even

Figure 2 compares the N-odd and N-even asinc graphs. The N-even graph in Figure 2 and Equation 2 have a unique property.[2] There is a point where the amplitude and slope are both zero at the same time. This unique property is taken advantage of to simplify the calculations for the asinc modulation method.

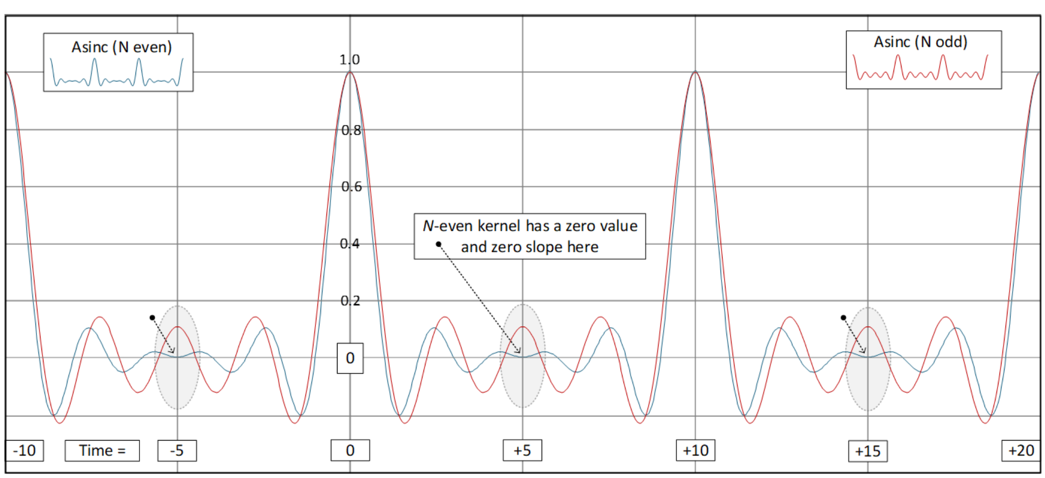
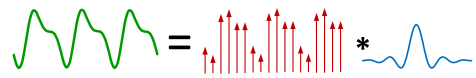


Figure 2. Even and odd asinc functions.

The asinc reconstruction of a periodic waveform is found in Equation 3. In the ideal case, the reconstructed waveform is exactly equivalent to the original input signal. A graphical example is shown in Figure 3. One period of the uniformly sampled version of the original waveform is shown in black. The convolution of the asinc kernel with one period of samples results in a fully reconstructed signal in green.

$$\hat{f}(t) = f_s(t) * h(t) = \sum_{n=0}^{N-1} f[n]\, h(t - nT_s) \qquad (3)$$



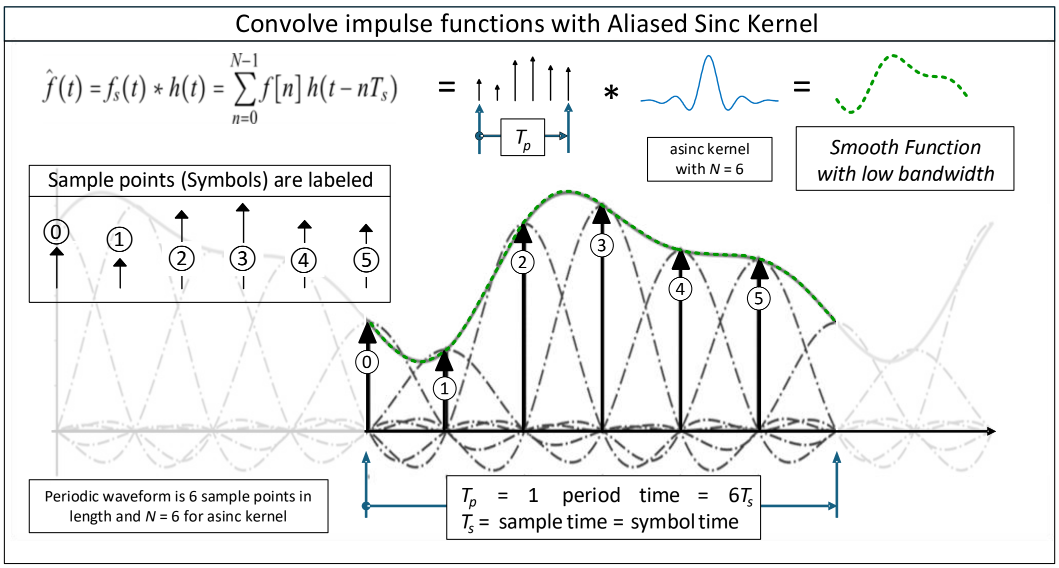


Figure 3. **Example ideal reconstruction of a periodic waveform from uniform samples.**

1.2 Asinc Bandlimited Interpolation and Digital Modulation

The asinc bandlimited interpolation and digital modulation method uses a modified version of the asinc interpolation example presented in Figure 3. It is based on the progressive asinc reconstruction (PAR) interpolation presented in Crowe and Jungwirth[2] and Section 6.4. PAR interpolation uses a slope-matching technique to provide for C^1 continuity (continuous in the first derivative) across the interpolation intervals.

Replacing the half-cycle raised cosine in Feher modulation by asinc interpolation results in a substantial reduction in bandwidth and improved sidelobe rejection. Simulations show a 24% improvement (reduction) in 99% power bandwidth points for asinc bandlimited interpolation and digital modulation[3] compared to half-cycle raised-cosine interpolation. The asinc modulation method is patent pending.

2. CT and CT-DSP Introduction

Asinc bandlimited interpolation and digital modulation adapted PAR interpolation for data modulation and communications applications. PAR interpolation was originally developed to interpolate level-crossing sampling waveforms for CT-DSP.

CT systems are continuous in time and can be either continuous or discrete in amplitude.[4–13] Figure 4[4,13] illustrates the continuous and discrete regions for analog and digital signal processing (DSP). CT-DSP is discrete in amplitude and continuous in time. The boundary regions have unique properties.[14] Frequency modulation and asynchronous delta sigma (ΔΣ) modulation both have properties of analog and asynchronous digital.

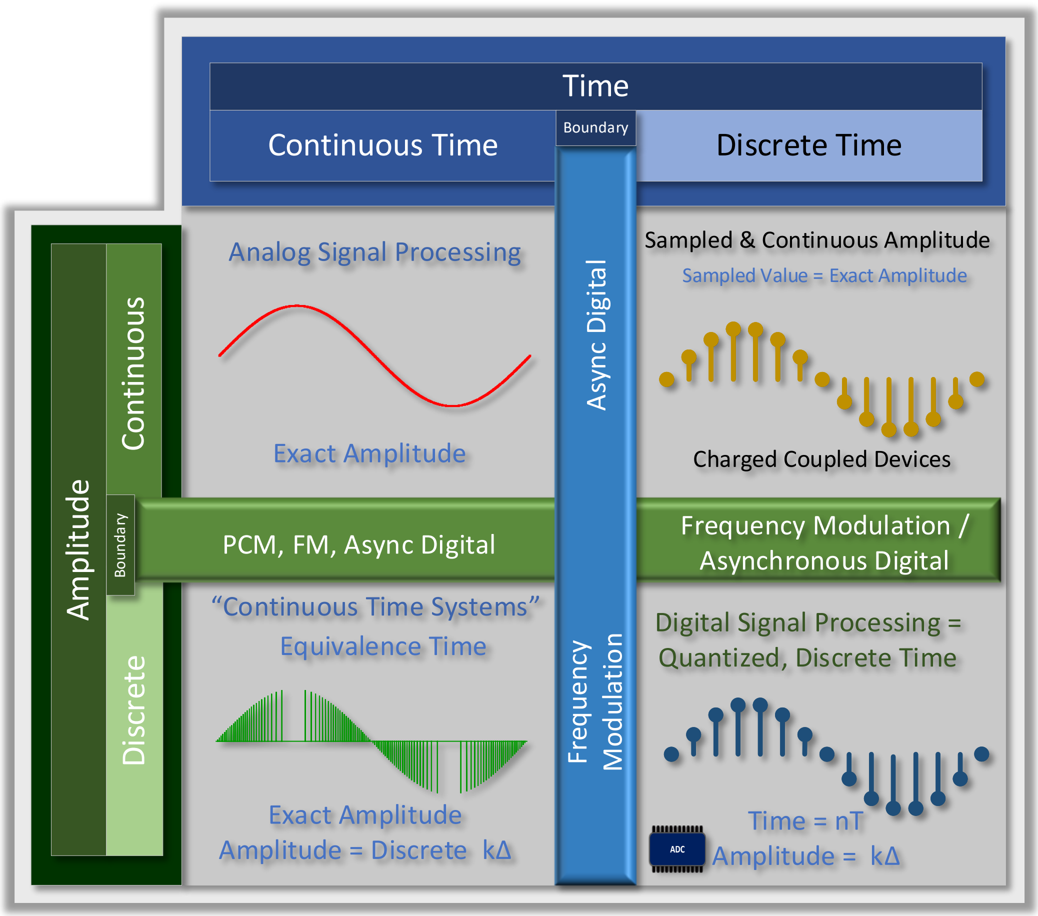


Figure 4. Signal processing regions.[4,13]

Figure 5 compares the properties of CT-DSP with conventional DSP. The main advantages of CT-DSP are no quantization error,[4] no frequency aliasing,[4] and signal activity dependent sampling (self-adapts to the input-signal's slope). The main disadvantages are the sampling process (signal capture) is nonlinear and the sample points are not uniformly spaced in time. Before any linear signal processing steps, CT-DSP must interpolate[12] the LCS values. The simplest interpolation algorithm is asynchronous zero-order hold (AZOH).[5] AZOH interpolation holds the last value constant until the next sample value is captured. For CT-DSP, the signal-to-noise ratio (SNR) is determined by the interpolation error. For conventional DSP, the signal-to-quantization noise ratio is determined by Bennett's equation, the oversampling ratio, and the low-pass frequency response of a zero-order hold.

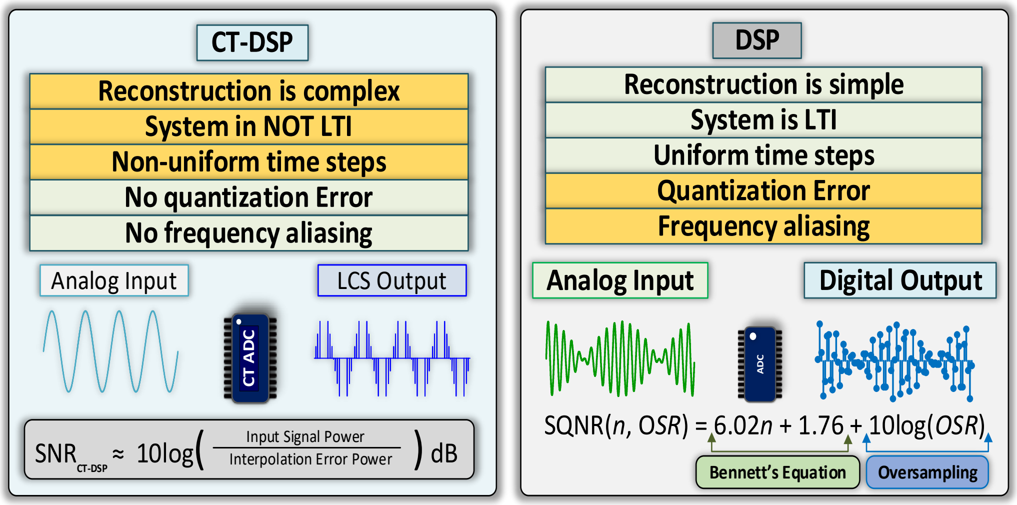


Figure 5. CT-DSP and conventional DSP comparison.

2.1 Level Crossing Sampling (LCS)

As illustrated in Figure 6, LCS is the 1-D version of a contour map. A contour map shows where the elevation equals an exact height. A 1-D slice from the contour map is equivalent to an LCS graph. As shown in Figure 7, an LCS point occurs when the input signal exactly equals a discrete voltage level. There is no quantization error like conventional DSP; however, the times between sample points are not constant (nonuniform), whereas for conventional DSP, the time steps are constant. LCS signal capture is nonlinear, whereas DSP is inherently linear. The LCS points must be interpolated before any linear signal processing blocks. The simplest interpolation algorithm is AZOH, which simply holds the output constant until the next sample point.

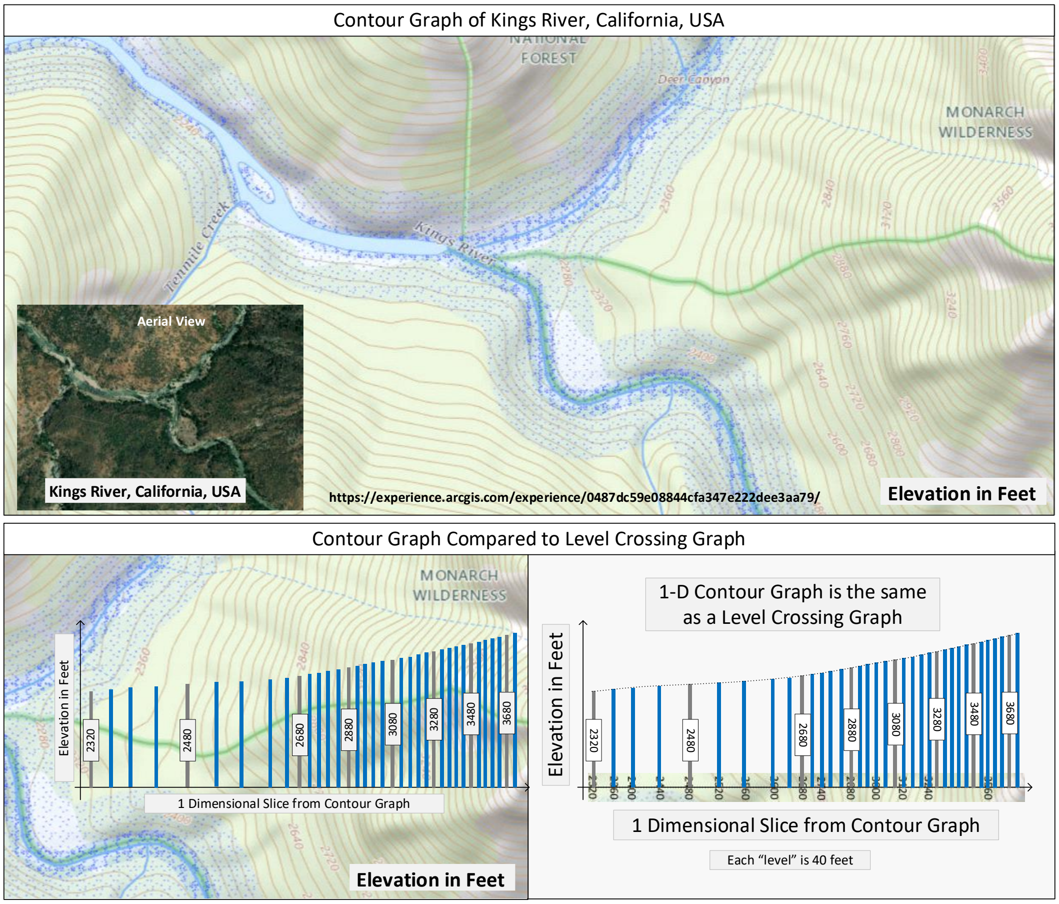


Figure 6. A contour graph is the 2-D equivalent of LCS. The contour lines show the lines of constant elevation. A slice of the contour map is the same as a level crossing signal.

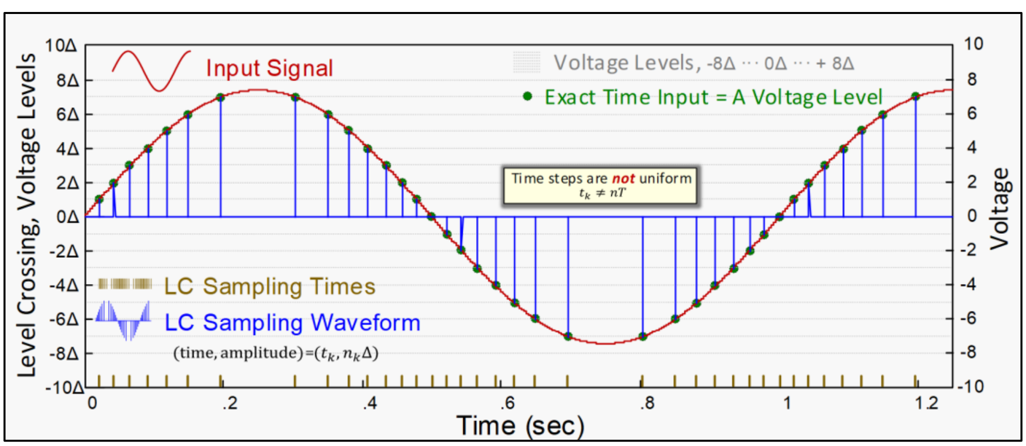


Figure 7. LCS example. An LCS occurs when the input signal exactly equals a discrete voltage level. A delta function is placed at each sample point to create an LCS signal.

Figure 8 illustrates AZOH interpolation with half-voltage level offset. The half-voltage level offset reduces the measurement error like half-voltage step offset reduces the quantization error for DSP. The waveform in Figure 8 is also called level crossing delta (LCD). Figure 9 compares the instantaneous sampling frequency for level crossing and discrete time. The compressive sensing property of LCS is shown Figures 10 and 11. Figure 10 compares a conventional 8-bit analog-to-digital converter (ADC) to a 32-voltage level, level-crossing ADC. Figure 11 compares an 8-bit conventional ADC to a 64-voltage level, level-crossing ADC. For CT-DSP, the instantaneous sampling frequency is proportional to the slope of the input signal. An electrocardiogram (ECG) signal has low- and high-slope regions. There are a few sample points in the low-slope regions. The high-slope regions are accurately captured with many sample points. Figure 11 shows LCS accurately captures the ECG waveform with only 38% of the number of discrete time samples.

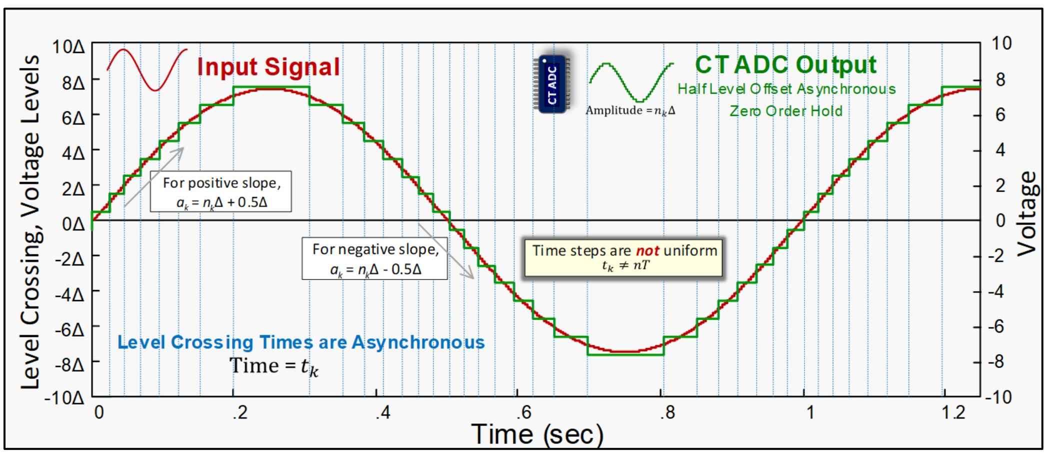


Figure 8. AZOH with half-voltage level offset. AZOH keeps the value constant until the next sample point is captured. For a positive slope, a half-voltage level is added, and for a negative slope, a half-voltage level is subtracted from the level-crossing value.

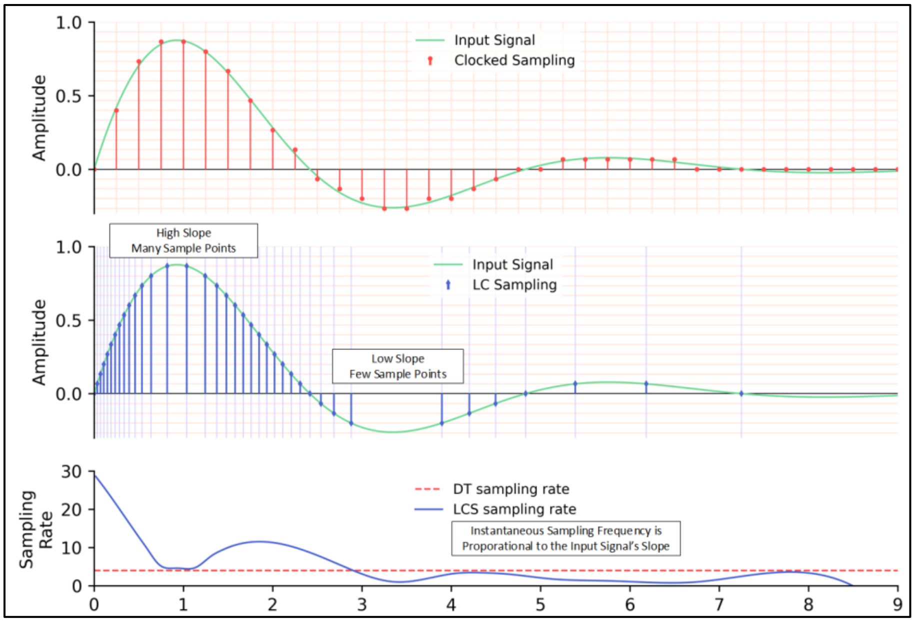


Figure 9. Conventional, discrete time sampling and LCS comparison. Discrete time has a fixed sampling time step and a fixed sampling frequency. For LCS, the instantaneous sampling frequency is proportional to the slope of the input signal.

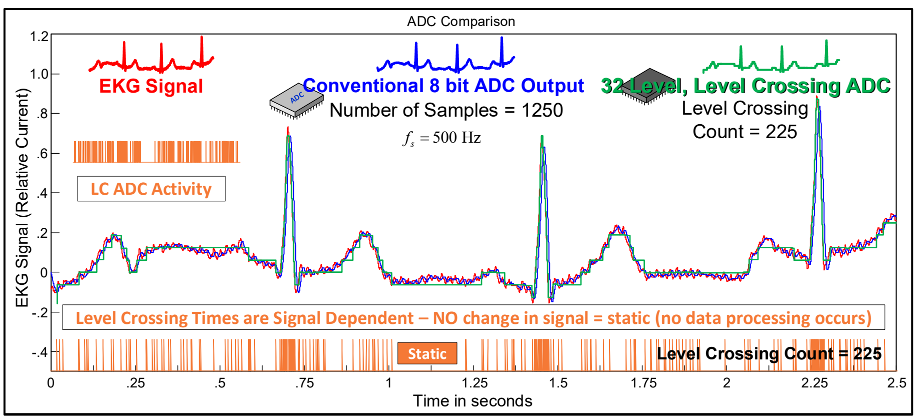


Figure 10. For ECG signals, Qaisar and Hussain showed a 3× reduction in the number of sample points while achieving a 97% accuracy in identifying heart arrythmias.[10]

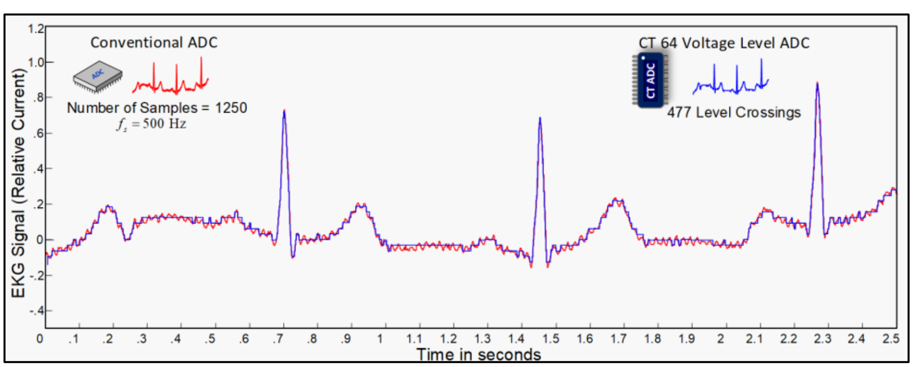


Figure 11. ECG signals for a conventional 8-bit ADC compared with a 64-voltage level, level-crossing ADC.

Figure 12 compares chirp waveforms for conventional discrete time and LCD. LCD self-adapts to the input signal's slope, accurately capturing the chirp waveform. For discrete time, a low-sampling frequency results in aliasing. Higher-sampling frequencies result in too many sample points at the low-frequency end of the chirp and not enough sample points at the high-frequency end of the chirp waveform.

Figure 13 compares the instantaneous sampling frequency for a chirp waveform captured by discrete time and LCS. As the chirp waveform's frequency changes, LCS captures a constant four sample points per half cycle. At the low-frequency end of the chirp waveform, the instantaneous sampling frequency is low. At the high-frequency end of the chirp waveform, the instantaneous sampling frequency is high, which accurately captures the chirp waveform. LCS's instantaneous sampling frequency self-adapts to the input signal's slope. For discrete time, the sampling frequency is constant. The constant sampling frequency captures too many sample points at the low-frequency end and not enough sample points at the high-frequency end. Discrete time results in a trade-off between high-frequency accuracy, oversampling ratio, and too many sample points at the low-frequency end of the chirp waveform.

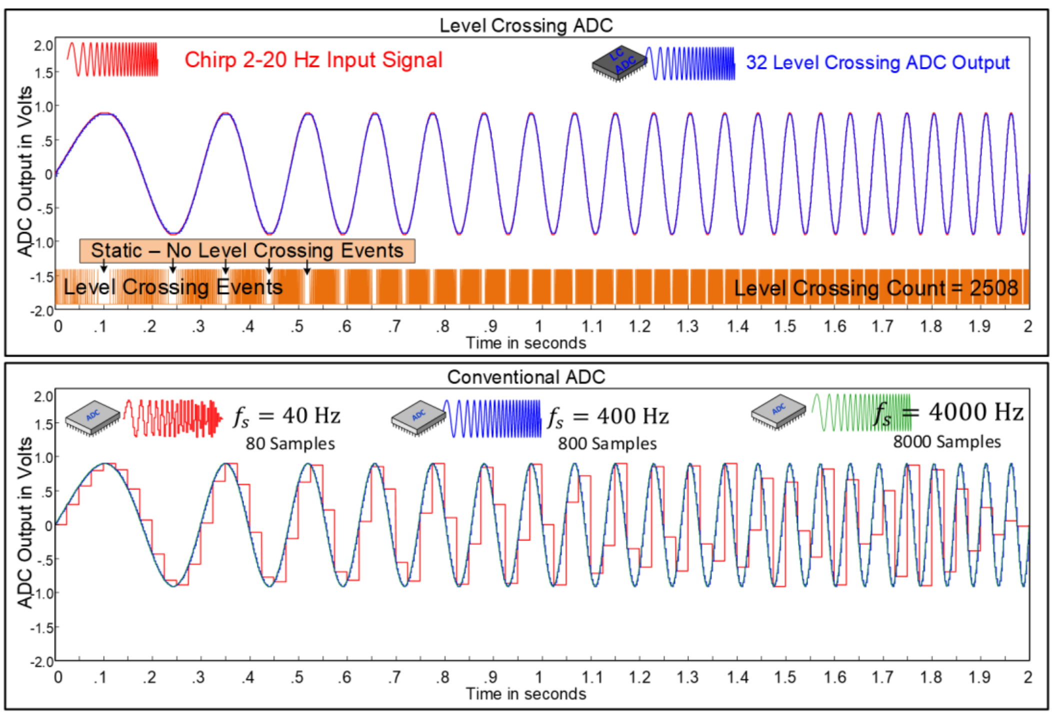


Figure 12. Chirp waveforms for a conventional ADC compared with a level-crossing ADC.

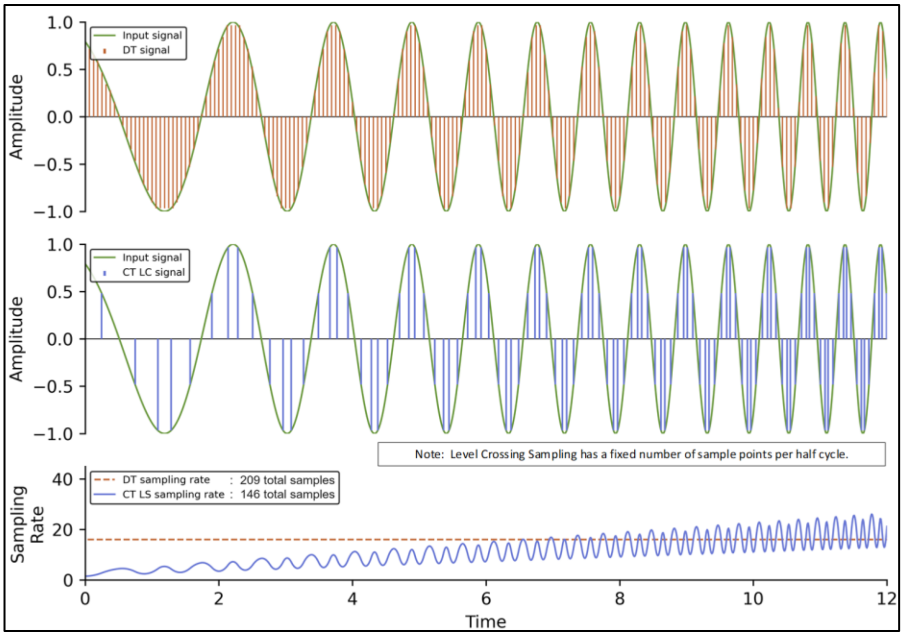


Figure 13. Discrete time sampling frequency and level-crossing sampling frequency comparison for a chirp waveform.

As illustrated in Figure 14, the slope of a sine wave is approximately linear near the zero crossing points. For a linear slope, LCS approaches Shannon sampling (constant time steps). The magnified view shows the linear region approaches Shannon sampling. Figure 15 compares a sine wave with a triangular wave. The triangular wave is not bandlimited. The triangular wave has linear slope regions and the time between sample points is constant. Linear slope waveforms are the invariant between Shannon sampling and LCS. We only use the triangular waveform to illustrate the invariant between Shannon sampling and LCS. All other cases discussed in this report are bandlimited.

In Figure 16, the time and amplitude axes for LCD are reversed; the x-axis = amplitude and y-axis = time. We find the y-axis is continuous and the x-axis is discrete. We also see the y-axis is multivalued. For discrete time, the x-axis is discrete and the y-axis is continuous. For digital, the y-axis is quantized, which makes it also discrete. Reversing the time and amplitude axes shows a similarity between CT and discrete time.

As illustrated in Figure 17, noise in CT-DSP systems affects the number of level crossings. For high SNR regions, there is only a small increase in the number of level crossings. Around SNR = 40 dB, there is an exponential increase in the number of level crossings. Near SNR = 30 dB, there is a break between the low-noise region where the level crossings from the input signal dominate to where the level crossings from noise start to dominate.

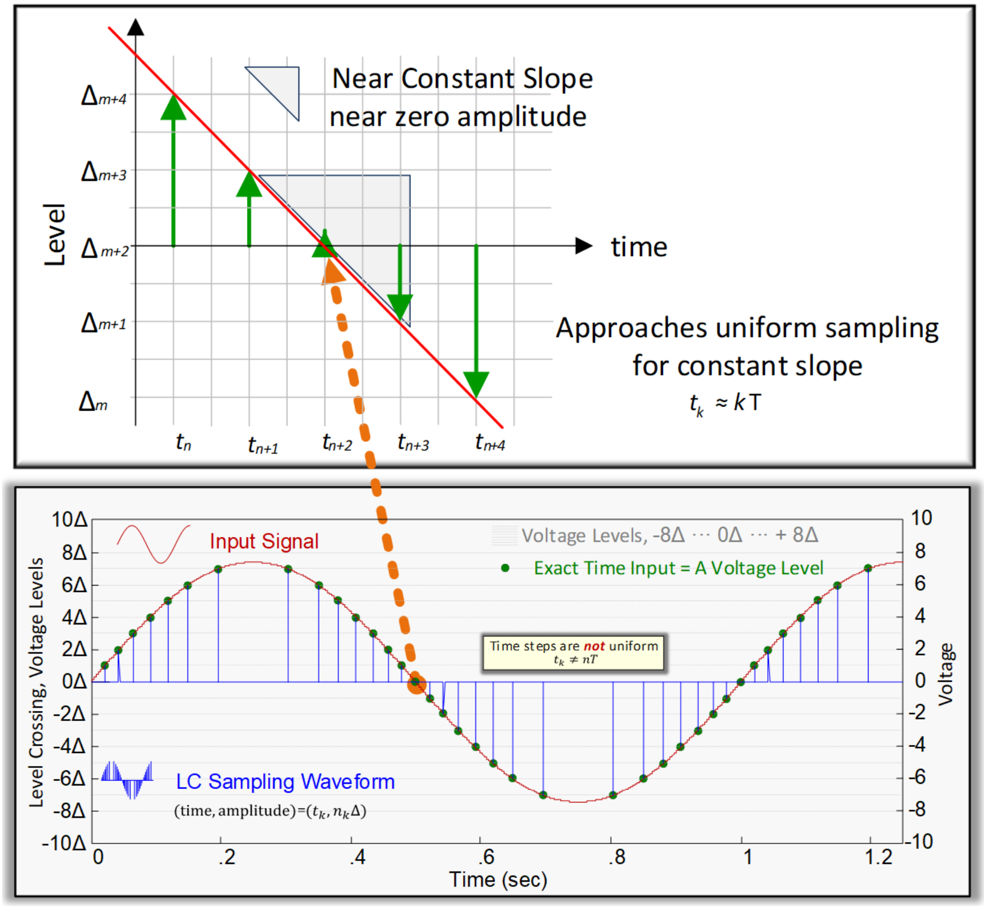


Figure 14. Constant slope region approaches Shannon sampling.

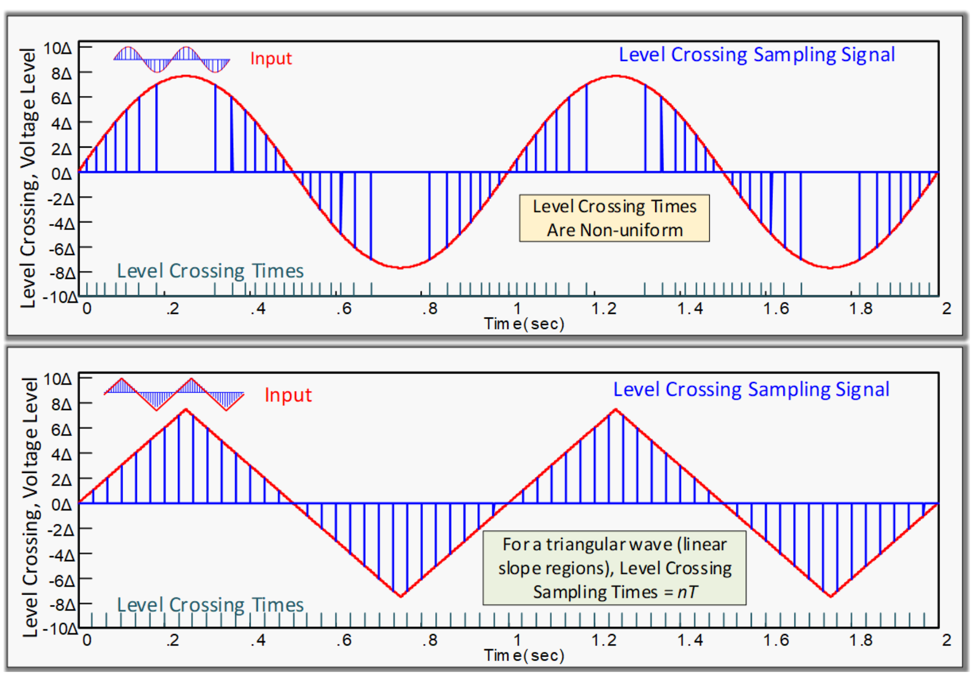


Figure 15. Constant slope waveforms have constant sampling times.

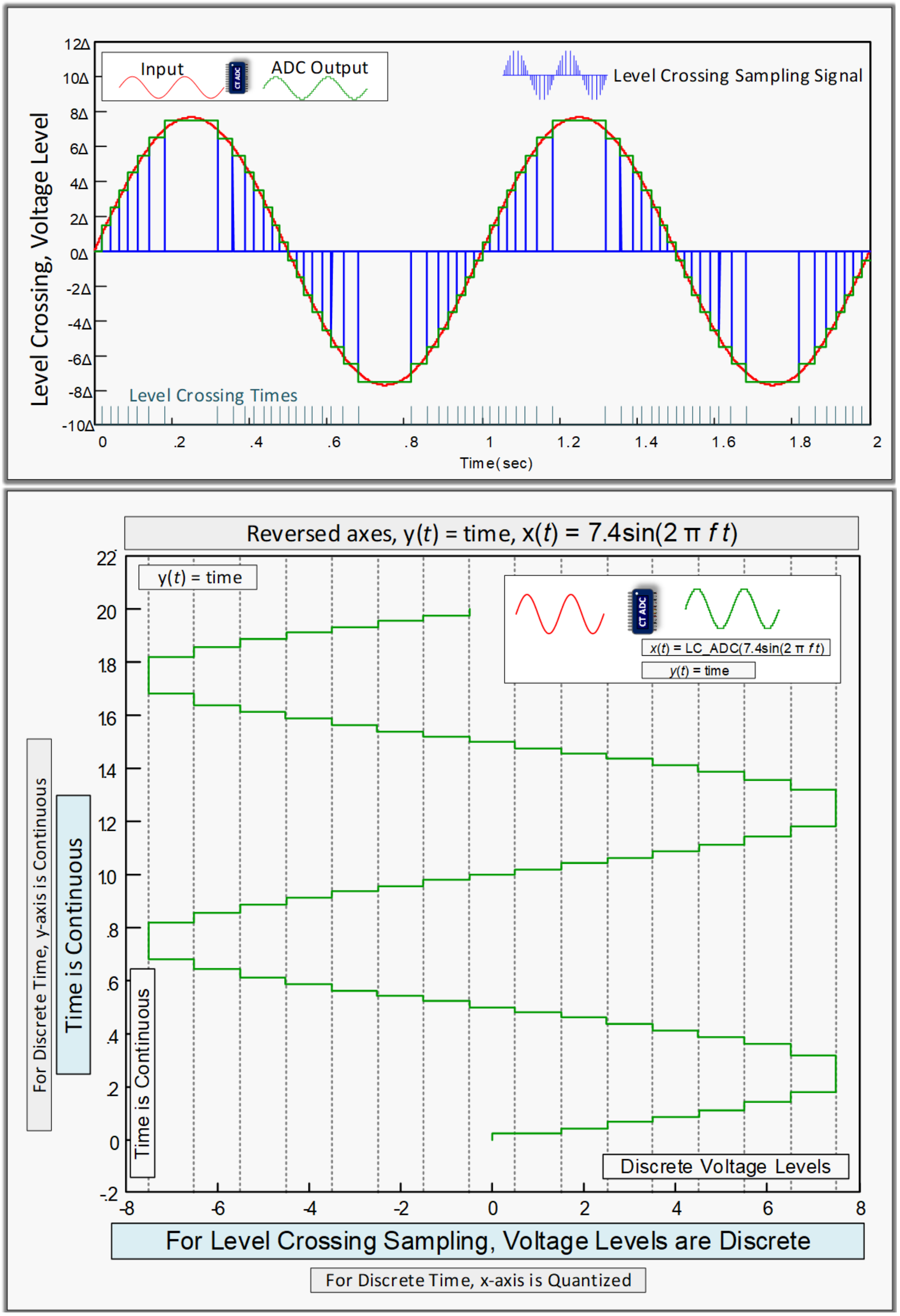


Figure 16. If the amplitude y-axis and time x-axis for CT-DSP are reversed, the reversed axes graph is similar to discrete time. The x-axis is discrete and y-axis is continuous. However, the reverse graph is multivalued.

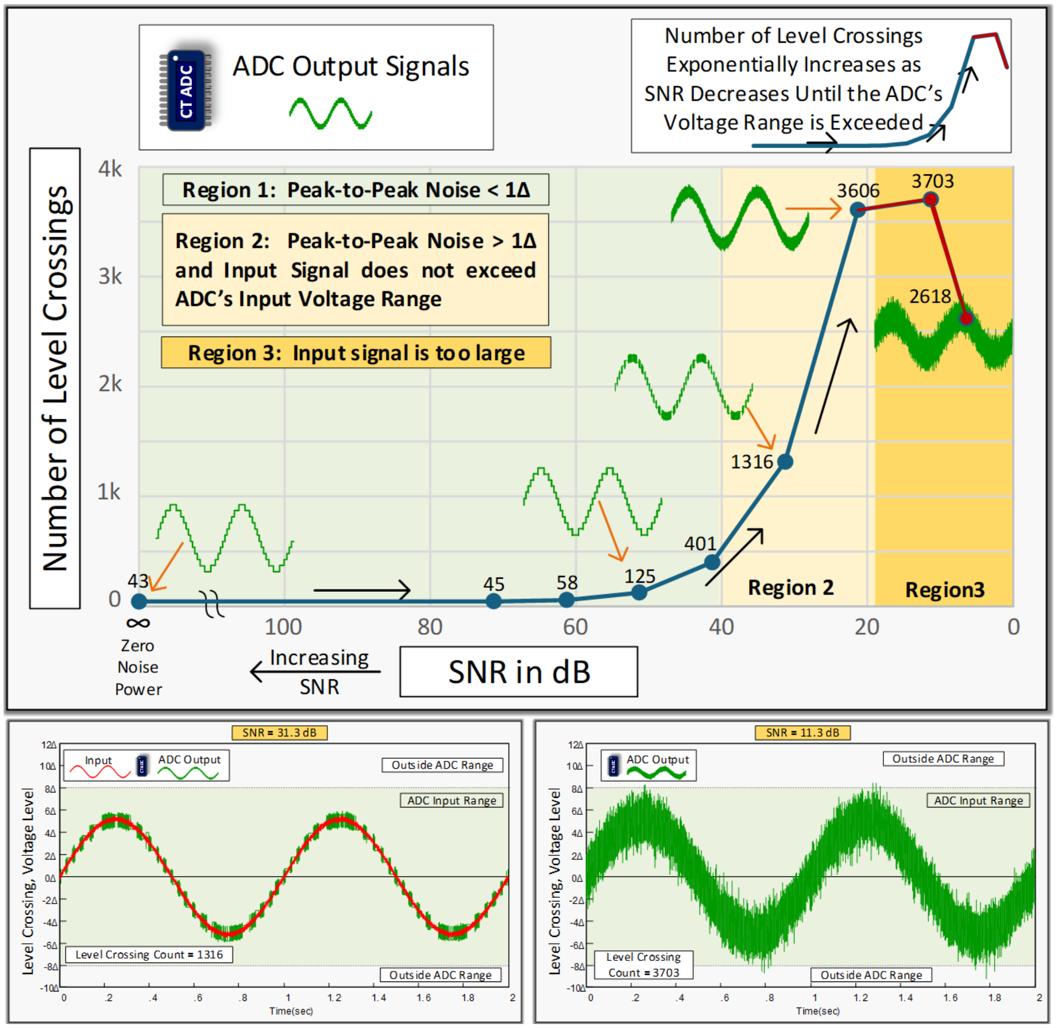


Figure 17. Gaussian white noise in CT-DSP systems.

2.2 LCS and PAR Interpolation

Section 2 presented an introduction to LCS and CT-DSP. PAR interpolation algorithm presented in Crowe and Jungwirth[1,2] was originally developed to interpolate LCS signals for CT-DSP. Section 6 only considers PAR interpolation for digital modulation applications using streaming data at a fixed clock frequency.

3. Sampling Theory Review

Asinc bandlimited interpolation and digital modulation uses PAR interpolation (asinc reconstruction with slope matching)[2] to reconstruct an interval inside a sliding frame. Sampling and reconstruction are reviewed to apply asinc reconstruction to a sliding frame.

3.1 Shannon Sampling Review

Figure 18 shows a block diagram for conventional Shannon sampling and signal reconstruction. Shannon sampling assumes all signals in the block diagram *exist for all time*. The input signal must be bandlimited, as shown in Equation 4. The ideal reconstruction filter for Shannon sampling is the ideal low-pass filter. The ideal low-pass filter has a sinc impulse response. The input signal is reconstructed by convolving a discrete time signal with the sinc function. Time-domain sampling is an amplitude modulation process with an infinite number of carrier frequencies. When the sampling theorem in Equation 4 is ***not*** satisfied, frequency aliasing occurs around the amplitude modulation carrier frequencies. The difficulty with the sinc convolution in Figure 18 is the very slow convergence of the sinc function. The slow convergence makes calculating the convolution difficult.

$$f_s > 2f_{bw}$$ The sampling frequency must be greater than twice the input signal's highest frequency. (4)

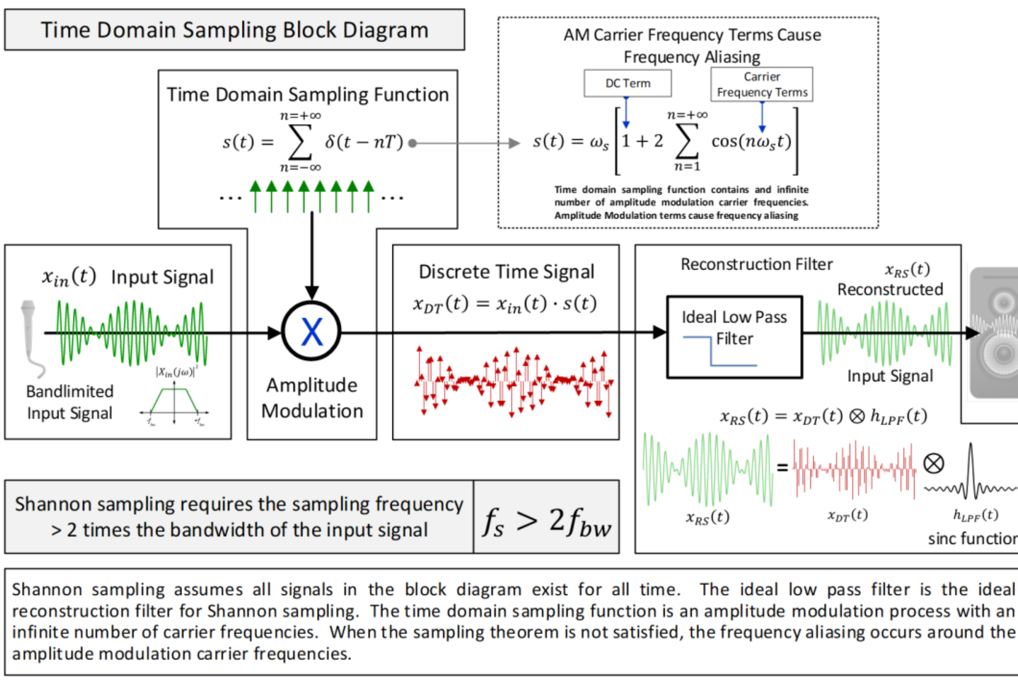


Figure 18. Shannon sampling block diagram.

Figure 19 illustrates the cosine form of the time-domain sampling function. As shown in Figure 18, multiplying the input signal by the sampling function results in sum and difference frequencies around the $cos(n\omega_s t)$ carrier frequencies where ω_s is the radian sampling frequency. Figure 20 illustrates that the sum and difference frequencies cause frequency aliasing when the sampling theorem in Equation 4 is not satisfied.

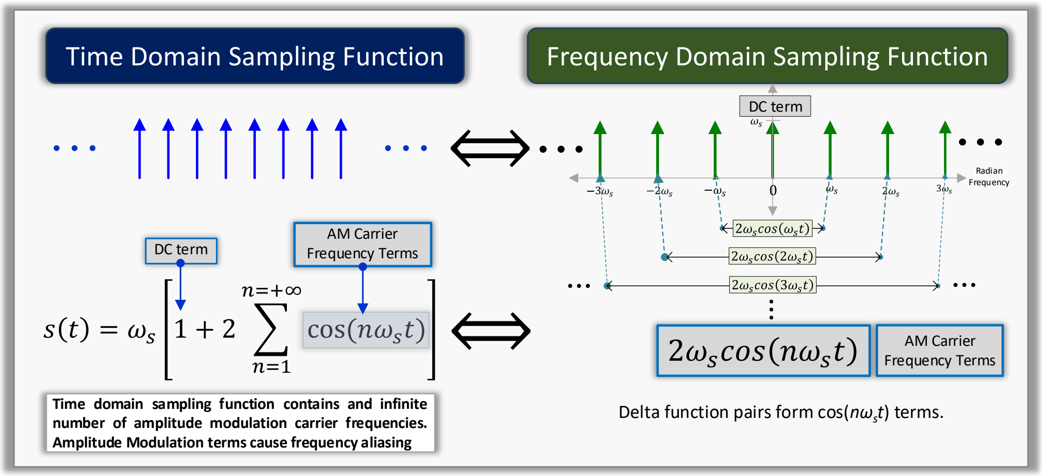


Figure 19. Time- and frequency-domain sampling functions.

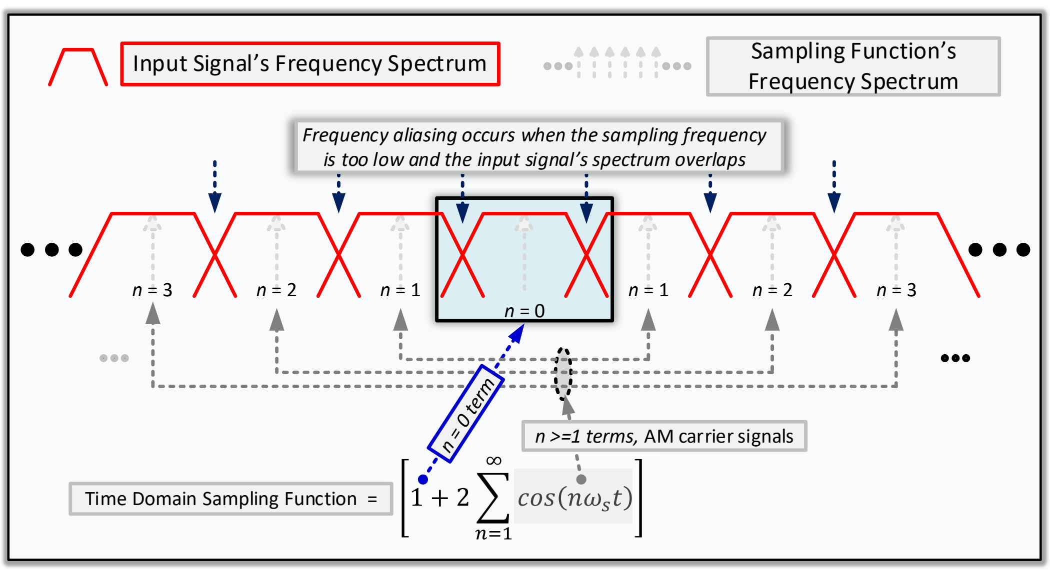


Figure 20. Frequency aliasing occurs around the $cos(n\omega_s t)$ carrier frequencies. The dashed delta functions are grouped in pairs for each $cos(n\omega_s t)$ carrier frequency. The input spectrum is replicated over the delta functions. When the replicated spectrums overlap, frequency aliasing results. Frequency aliasing cannot be removed.

3.2 Periodic Input Signal and Asinc Reconstruction

In Figure 21, the input signal is periodic, and the ideal reconstruction kernel is the asinc function. As illustrated in Figure 22, the asinc function is also periodic. Reconstruction over one period reconstructs the periodic input signal for all time. In Figure 18, sinc reconstruction has a very slow convergence and requires a multitude of terms for accurate reconstruction. The reconstruction window for the asinc is finite and it is much simpler to calculate than the sinc reconstruction in Figure 18. Asinc bandlimited interpolation and digital modulation takes advantage of the properties of the asinc function.

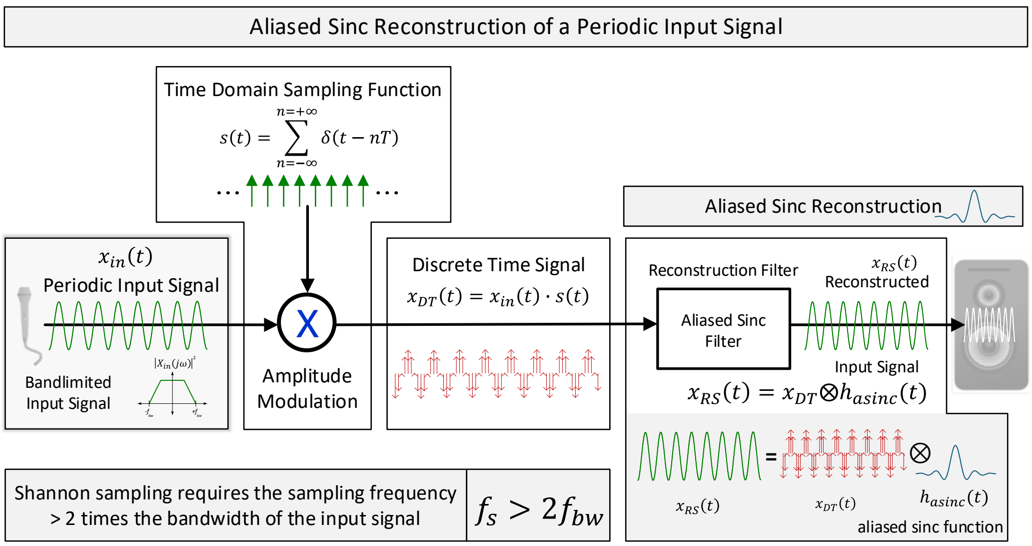


Figure 21. Block diagram for sampling a periodic function. The reconstruction kernel is the asinc function. The asinc function's period has a finite width, simplifying calculations.

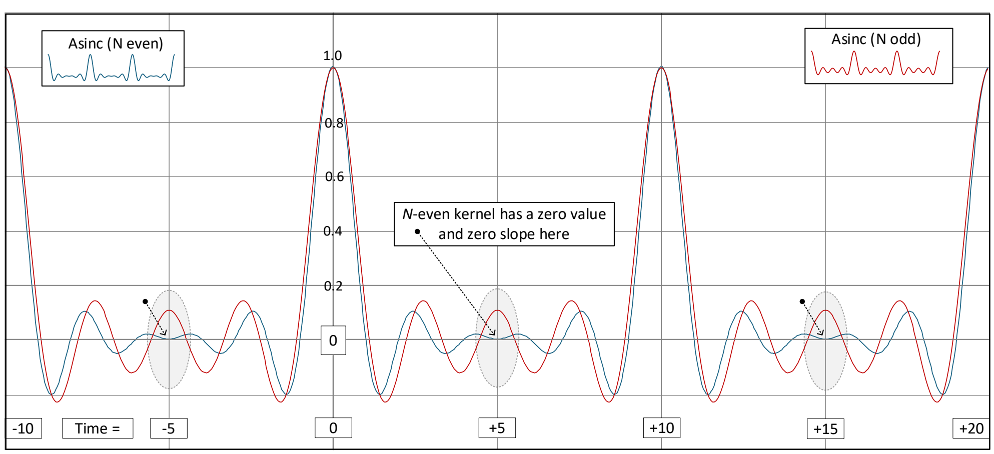


Figure 22. Comparison of even and odd asinc functions. The even asinc function has a zero slope and zero amplitude at the same point. This property of *N* even asinc function is used to simplify calculations for PAR interpolation.

4. Modulation Review

Section 4 reviews quadrature amplitude modulation (QAM).[15–20] Section 5 presents an overview of Feher QAM and Section 6 focuses on PAR interpolation for QAM.

Figure 23 presents some examples of conventional QAM for QAM (*a*, *b*) constellation diagram sizes of QAM 4 (2 × 2), QAM 16 (4 × 4), and QAM 64 (8 × 8). Figure 24 shows a block diagram for a QAM 16 (4 × 4) modulator. QAM combines amplitude modulation (in-phase, $I(t)$, channel)[21] and phase modulation (quadrature phase, $Q(t)$, channel).[22, 23] By converting $a(t)$ and $b(t)$ in Equation 5 into cosine and sine functions in Equations 6 and 7, the in-phase and quadrature phase form in Equation 8 is found.

Due to the symmetry of the QAM 4 constellation diagram, the coordinates have a constant radius from the origin. QAM 4 has a constant amplitude value and only the phase changes. QAM 16 has four values for the in-phase axis and four values for the quadrature phase axis. QAM 64 has eight values for in-phase and eight values for quadrature phase.

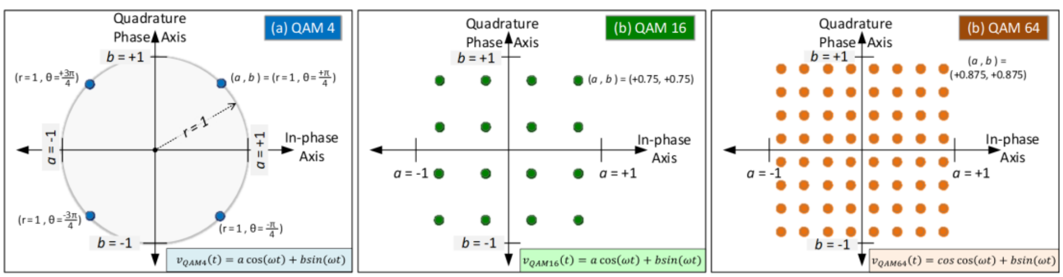


Figure 23. QAM constellation diagrams.

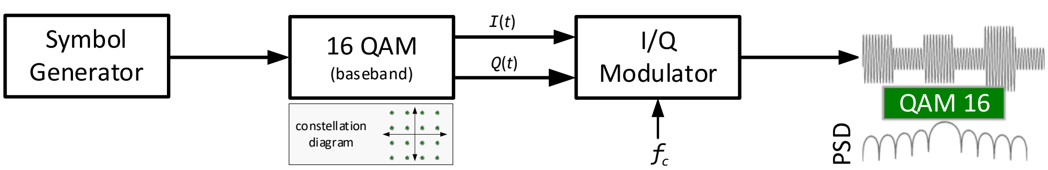


Figure 24. Simple QAM modulator block diagram.

$$v_{QAM(n)}(t) = a(t)cos(\omega t) + b(t)sin(\omega t) \quad \text{where } \omega = \text{carrier frequency} \qquad (5)$$

$$v_{QAM(n)}(t) = \sqrt{a^2+b^2}\left[\frac{a\cos(\omega t)}{\sqrt{a^2+b^2}} + \frac{b\sin(\omega t)}{\sqrt{a^2+b^2}}\right] \quad \text{where } a \text{ and } b \text{ are the QAM coefficients} \qquad (6)$$

$$w_{QAM(n)}(t) = cos(\theta)cos(\omega t) - sin(\theta)sin(\omega t) \quad cos(\theta) = \frac{a}{\sqrt{a^2+b^2}} \quad sin(\theta) = \frac{b}{\sqrt{a^2+b^2}} \qquad (7)$$

$$w_{QAM(n)}(t) = I(t)cos(\omega t) - Q(t)sin(\omega t) \quad \text{where } I(t) = \text{in-phase channel, and } Q(t) = \text{quadrature channel} \qquad (8)$$

5. Feher or Half-Cycle Raised-Cosine Interpolation

Feher or half-cycle raised-cosine interpolation (also called modulation) replaces symbol changes with a half-cycle raised cosine.[24–30] To minimize the bandwidth, the half-cycle raised-cosine segment uses the full symbol time. Feher modulation has some limitations. A controlled amount of intersymbol interference and signal overshoot can be used to reduce bandwidth. Feher modulation does not allow for a controlled amount of intersymbol interference, nor does it allow a controlled amount of signal overshoot. Feher modulation is not bandlimited.

5.1 Serial Binary Half-Cycle Raised-Cosine Interpolation

Figure 25 illustrates the mapping from symbol transitions to Feher interpolation. There are three mapping cases for a two-symbol system, falling edge symbol transition, rising edge symbol transition, and no transition (symbol value stays the same). Figure 26 shows a mapping from binary serial data to half-cycle raised-cosine interpolation. Feher modulation results in a smooth curve with all symbol transition points occurring at zero slope points. Some limitations of half-cycle raised-cosine modulation are presented in Figure 27. A line segment and a half-cycle raised-cosine waveform are not bandlimited. The half-cycle raised-cosine results in a larger slope than a linear approximation.

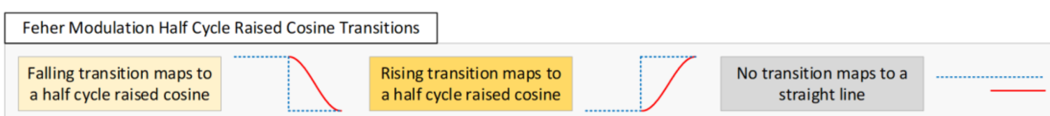


Figure 25. Half-cycle raised-cosine symbol mapping.

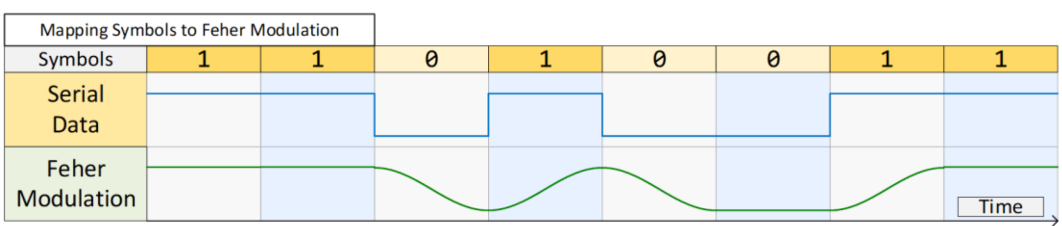


Figure 26. Binary serial data mapping to Feher, half-cycle raised-cosine interpolation.

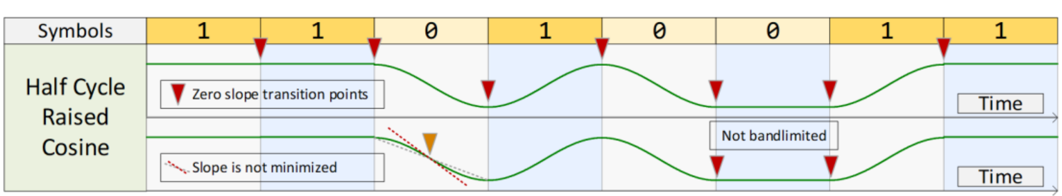


Figure 27. Some limitations of half-cycle raised-cosine interpolation.

5.2 Feher 16 QAM Interpolation

Feher, half-cycle raised-cosine interpolation block diagram for 16 QAM is presented in Figure 28. Feher modulation is applied to $I(t)$, and $Q(t)$ channels separately. Feher QAM modulation offers a significant bandwidth improvement over standard QAM; however, it still exhibits significant sidelobes in its power spectral density graph. A block diagram for Feher QAM modulation is found in Figure 29. A mapping from the I-Channel data stream to Feher 16 QAM modulation is shown in Figure 30. Half-cycle raised-cosine waveforms connect the symbol starting points together forming a smooth curve.

Figure 31 shows some performance limitations for half-cycle raised-cosine modulation. Half-cycle raised-cosine modulation is not bandlimited. The slope is not minimized. Feher modulation results in harmonic energy occurring at multiples of the symbol frequency. Overshoot is not allowed and there is no controlled intersymbol interference. Figure 31 also shows the $I(t)$ channel (or $Q(t)$ waveform is the same) for 16 QAM Feher modulation. When a half-cycle raised-cosine segment to the left of a transition point does not match the half-cycle raised-cosine segment to the right, the output waveform is skewed at the transition point (same as symbol point).

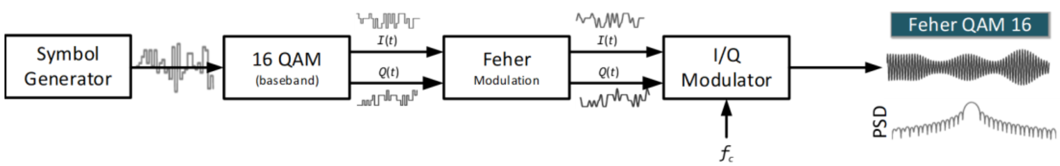


Figure 28. Feher QAM modulator block diagram.

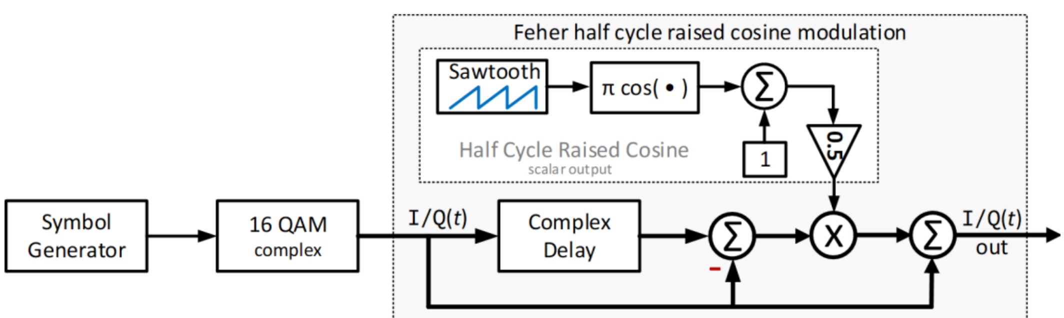


Figure 29. Feher modulation block diagram.

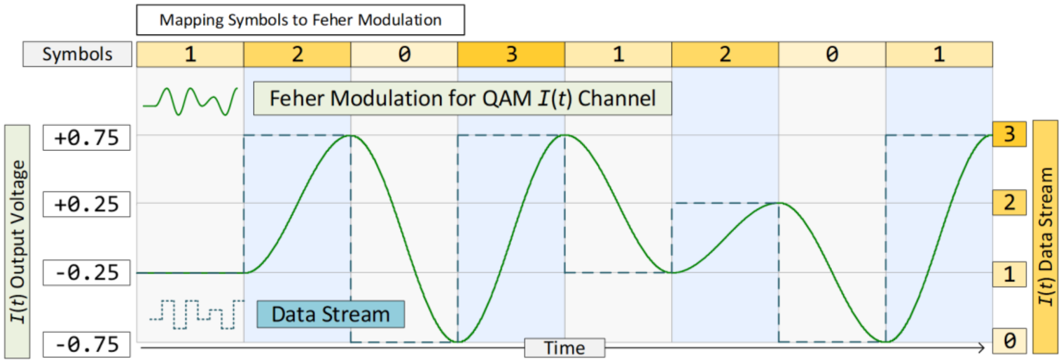


Figure 30. Feher interpolation for 16 QAM $I(t)$ channel.

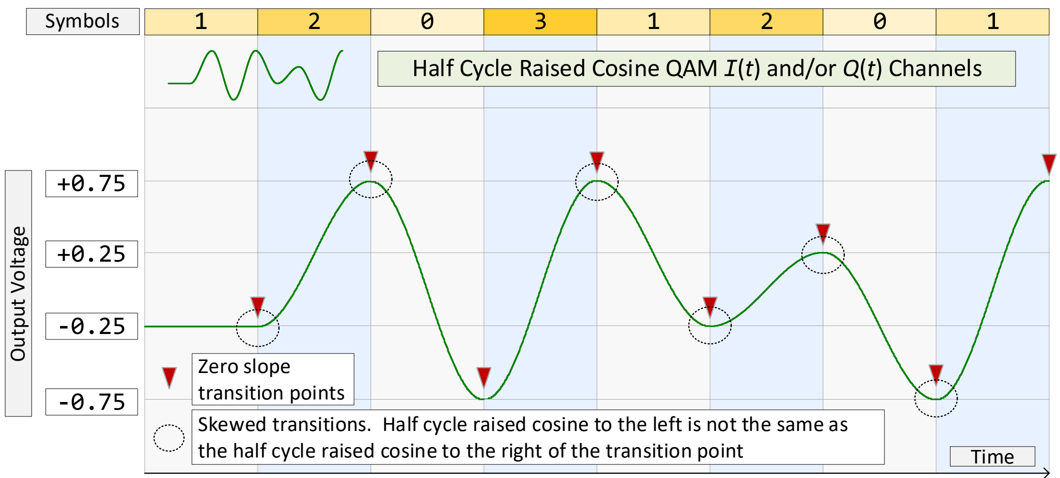


Figure 31. Some limitations of Feher, half-cycle raised-cosine modulation.

Figure 32 shows a diagram that illustrates how the sampling function and convolution can be used to create Feher modulation. Figure 26 illustrates serial binary half-cycle raised-cosine interpolation. In Figure 32, a serial data stream consisting of +1 and –1 symbols is sampled, resulting in a sequence of impulse functions labeled as +1↑ and –1↑. The impulse functions are located at the start of each symbol. The impulse functions are spaced one symbol time apart. The state change row shows three cases: 1) the symbol stream does not change, 2) has a rising edge, or a 3) falling edge. The half-cycle raised-cosine kernel is convolved with the state change's impulse functions giving the results shown in the result row. The result row shows a discontinuous waveform. By adding in the initial amplitude from each serial data point, a smooth Feher interpolated waveform is generated as shown. Figure 32 illustrates how a sequence of sample points can be convolved with a half-cycle raised-cosine function to create a Feher (half-cycle raised-cosine) interpolation waveform.

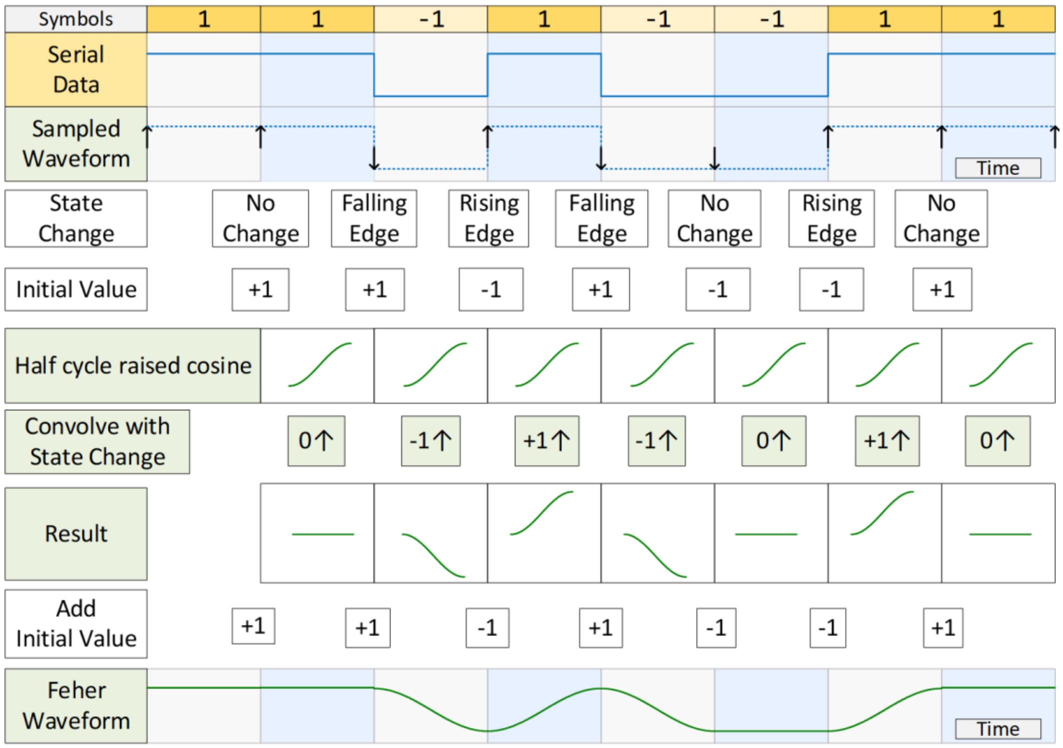


Figure 32. State change convolution to generate Feher interpolation for serial binary data.

6. Asinc Bandlimited Interpolation

Asinc bandlimited interpolation uses a sliding asinc interpolation frame with slope matching (PAR algorithm in Crowe and Jungwirth[1,2]) to convert a symbol stream into a smooth, low-bandwidth waveform. The asinc interpolation includes slope matching to provide improved (less) bandwidth and improved (reduced) sidelobe power spectral density artifacts compared to Feher half-cycle raised-cosine interpolation. For 16 QAM, PAR interpolation shows an approximately 24% improvement in the 99% power bandwidth points compared to Feher half-cycle raised-cosine interpolation and improved sidelobe rejection.

The genesis of this modulation technique lies in the ideal reconstruction of uniformly sampled periodic signals. This ideal reconstruction is accomplished by convolution of the samples with the asinc function.[1,2] A sequence of QAM symbols, for either in-phase or quadrature phase, can be represented as a uniformly sampled signal; however, when processing signals in real time, it is not possible to know if any particular segment of the signal is periodic.

Modulation schemes like ASK and QAM encode a digital signal by defining a set of keys (or symbol values) representing the range of digital values. These keys are uniquely mapped to voltage (or current, power, flux, etc.) levels. Thus, a time sequence of keys can be represented as a sequence of uniformly spaced impulses.

The operation to convert these impulses to a continuous baseband signal is the same as reconstruction of a discrete time signal.

To be useful, any reconstruction operator must match the time and amplitude of each impulse and generate a bandlimited signal. WKS reconstruction satisfies these conditions by convolving with the sinc kernel in Figure 23. However, the sinc kernel does not converge quickly, necessitating many calculations at each reconstructed point.

When processing a signal in real time, only a limited segment of the signal is processed at any particular time. Applying asinc convolution to such a limited segment might introduce inaccuracies. For example, any disparity of amplitude of slope between the beginning sample and the ending sample would cause unwanted artifacts. To overcome these limitations, this technique adopts a scheme that reconstructs the interval between an adjacent pair of samples using an enveloping frame consisting of surrounding samples. After one interval has been reconstructed, the frame is shifted by one sample (one symbol), and the process is repeated, reconstructing the next interval.

The asinc function in Figure 33 has two different forms, depending on the evenness of the number of samples contained in a frame. For a sequence containing an even number of samples, the *N*-even function is used. For a sequence of an odd number of samples, the *N*-odd form is used. This technique can make use of either form of the asinc function.

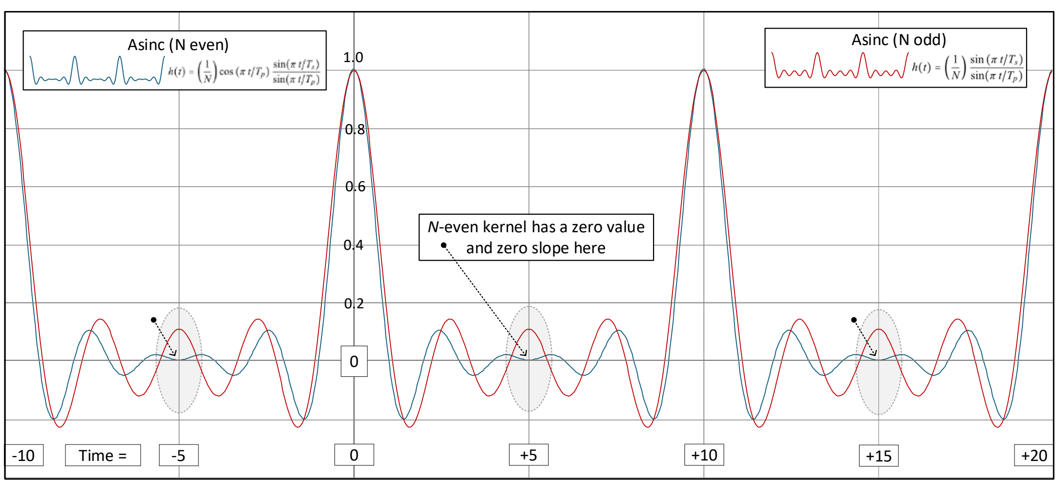


Figure 33. Asinc kernel graphs.

The examples given here describe a technique that uses frames, which are always composed of six samples. Previous numerical results indicated that a frame size of 6 was nearly as accurate as a frame size of 10 or 12 samples. To accurately estimate slopes, at least one sample must be known following the interval. For samples within a frame, indexed as {⓪, ①, ②, ③, ④, ⑤}, the interval being reconstructed is between ③ and ④. Sample ⑤ provides accuracy in slope estimation. Samples ⓪ and ① provide kernels whose amplitudes can be adjusted for slope correction. And sample ② exists to force the kernel to be *N*-even, enabling the use of the simpler slope correction calculation.

In Figure 33, the *N*-even asinc kernel has a zero value and a zero-slope sidelobe at the same position in time. This property greatly simplifies the slope correction calculations. The *N*-odd aliased kernel case does *not* have this useful property. The slope correction calculations for the *N*-odd aliased kernel require matrix algebra with six or more equations for a solution. This level of complexity makes the *N*-odd aliased kernel impractical for hard real-time streaming applications; however, it is still a valid solution. The *N*-odd case can provide additional flexibility when working with nonlinear and/or complex communication channels (for example, arctic flutter, multipath, dispersive, and ionosphere disturbances). The additional equations can provide more opportunities for tuning and adjustment at the cost of a much higher computation cost.

While *N*-even and *N*-odd asinc function waveforms can be used for this purpose, we found that slope matching can be greatly simplified by using the *N*-even asinc function waveform. In the case of $N_{even} = 6$, the frame positions may be identified and referenced as g[0] = ⓪, g[1] = ①, g[2] = ②, g[3] = ③, g[4] = ④, and g[5] = ⑤, and the selected sample interval points are frame positions g[3] and g[4].

In Figure 34, we found for the even case, the amplitudes of the values in the frame corresponding to frame positions g[1] and g[0] can be adjusted to simplify the calculations to adjust the slopes at positions g[3] and g[4], respectively. More particularly, we found that the amplitude of the value in the frame corresponding to frame position g[1] can be adjusted to ğ[1] to correct the slope difference Δs[3] between segments beginning and ending at frame position g[3]. The amplitude of the value in the frame corresponding to frame position g[0] is adjusted to ğ[0] to correct the slope difference Δs[4] between segments beginning and ending at frame position g[4]. After one interval has been reconstructed, the frame is shifted by one sample (1 symbol), and the process is repeated, reconstructing the next interval.

The slope matching advantageously provides improved (less) bandwidth and improved (reduced) sidelobe power spectral density artifacts compared to Feher half-cycle raised-cosine interpolation. For 16 QAM, we found approximately 24% improvement in the 99% power points compared to Feher half-cycle raised-cosine interpolation and improved sidelobe rejection.

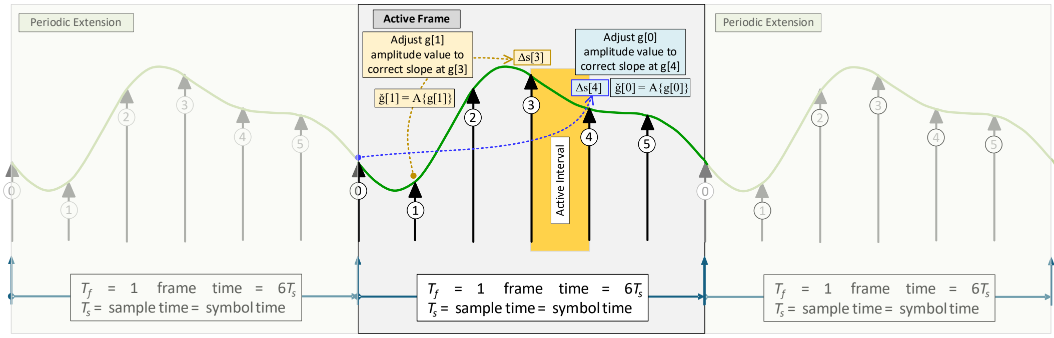


Figure 34. Adjust asinc kernels for slope matching.

The description focuses on ASK and QAM modulation systems. The PAR interpolation can easily be adapted and applied to any compatible modulation scheme or any communications system. A non-exhaustive list includes on/off keying (OOK), ASK, phase shift keying (PSK), frequency shift keying (FSK), pulse amplitude, pulse frequency, pulse position, pulse duty, QAM, discrete pulses, Morse code, spread spectrum, or more advanced combinations of amplitude/phase/frequency/pulse/time modulation. The modulation system may also be used for pulse applications like radar, sonar, autocorrelation applications, and others.

The progressive asinc reconstruction method has the property that the symbol value always equals the output waveform at each symbol's start time. By simply sampling at the symbol times, the exact symbol values can be recovered.

To demodulate the asinc modulation, there are several possible methods. Prior art sampling is a relatively simple approach.[15–20] More complex methods can be used. We have found that half-cycle raised-cosine modulation is a good approximation. Half-cycle raised-cosine demodulation techniques may be used as an approximation.[25–29] Prior art tracking demodulation methods can also be applied to create a demodulator.[15–20]

6.1 Asinc Reconstruction without Slope Matching

In this section, PAR interpolation without slope matching is presented. Modulation schemes like ASK and QAM encode a digital signal by defining a set of keys (or symbol values) representing the range of digital values. These keys are uniquely mapped to voltage (or current, power, flux, etc.) levels. Thus, a time sequence of keys can be represented as a sequence of uniformly spaced impulses. The operation to convert these impulses to a continuous baseband signal is the same as reconstruction of a digital signal.

To be useful, any reconstruction operator must match the time and amplitude of each impulse and generate a bandlimited signal. WKS reconstruction satisfies these conditions by convolving with the sinc kernel (Figure 18). However, the sinc kernel does not converge quickly, necessitating many calculations for each reconstructed point.

For a uniformly sampled representation of a bandlimited, periodic waveform in Figure 21, the ideal reconstruction kernel is the asinc (periodic sinc) kernel in Figure 33. For a frame whose width, $T_f = T_p$, is equal to the assumed period of the signal, T_p, the reconstruction of the center point consists of the convolution of the asinc kernel, with the samples within that frame. Thus, the number of terms is finite. We will call the assumed period of the signal, the frame time, T_f. The asinc kernel is defined by the sample spacing (or symbol time), T_s, and the frame time, T_f (which is equal to the assumed period of the input signal, T_p). There are two forms of the asinc function, depending on the evenness of N in Figure 33. Equation 9 gives the form for the case where N is odd, which is the simpler and the more common version.

$$h(t) = \left(\frac{1}{N}\right)\frac{\sin(\pi\, t/T_s)}{\sin(\pi\, t/T_p)} \qquad (9)$$

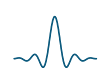
T_s = sample time (or symbol time)
T_p = 1 period = T_f = frame time = $N\,T_s$
N = number of sample points

N is odd

Equation 10 defines the kernel for the case where N is even in Figure 33. This form is a bit more complex, and not as often discussed. The peak of these kernels occurs at $t = 0$, and is replicated every period, T_p, where $t = qT_p$, for integer q (frame time, T_f, is assumed be equal to T_p). At these peak locations, the above expressions are indeterminate; however, using L'Hôpital's rule, it can be shown that the amplitude at these points approaches 1, and the slope approaches 0. These indeterminate points must be accounted for, explicitly, when performing numerical simulations. In Figure 33, the N-even asinc is characterized by a zero-amplitude point and zero-slope sidelobe. This property is used to create the PAR interpolation[1,2] algorithm.

$$h(t) = \left(\frac{1}{N}\right)\cos(\pi\, t/T_p)\,\frac{\sin(\pi\, t/T_s)}{\sin(\pi\, t/T_p)} \qquad (10)$$

T_s = sample time (or symbol time)
T_p = 1 period = T_f = frame time = $N\,T_s$
N = number of sample points



N is even

As pointed out in Equations 9 and 10, the frame time, T_f, is assumed to equal the period of the waveform, T_p, which equals N times the sample time (or symbol time), $T_f = T_p = N\,T_s$, where, N is the number of sample points in the frame.

In Figure 21, we show a block diagram for ideal reconstruction of a periodic input signal. The only difference between Figures 18 and 21 is the input signal is periodic and the ideal reconstruction kernel is the asinc function. Reconstruction of a periodic signal over one period reconstructs the input signal for all time. Instead of the slow convergence of the sinc function, we can take advantage of the finite length of one period of the asinc function for interpolation. The present modulation method takes advantage of this property.

Figure 35 provides a flow diagram for PAR without slope matching. PAR (which includes slope matching) will be discussed later. A symbol generator creates a stream of impulse functions. A moving frame of width $N_{even} = 6$ symbols (samples) is used to calculate the reconstructed waveform over the interval (consisting of symbols $g[3]$ = ③ and $g[4]$ = ④ in Figure 36). The symbol stream may be synchronous, or asynchronous.

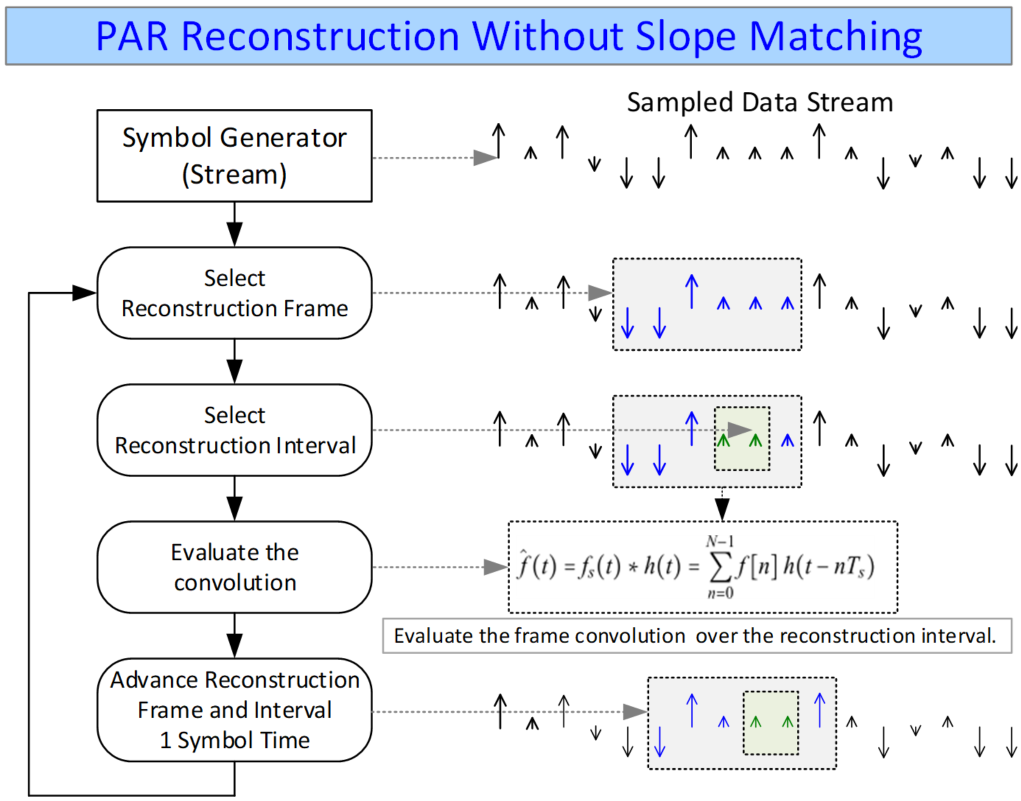


Figure 35. Flow diagram for PAR without slope matching.

Figure 36 presents a block diagram for PAR reconstruction without slope matching. The calculations are performed in the sliding frame. Each frame corresponds to a width of $N_{even} = 6$ symbols from the input stream. A symbol generator is converted to 16 QAM $I(t)$ channel data. A sliding frame may be composed of several sequential latches. The clocked/handshake-driven latches form the symbol frame composed of sample points {$g[0]$ = ⓪, $g[1]$ = ①, $g[2]$ = ②, $g[3]$ = ③, $g[4]$ = ④, and $g[5]$ = ⑤} where {⓪, ①, ②, ③, ④, and ⑤} are the respective symbol values. The reconstruction interval covers sample points (symbols) $g[3]$ = ③ and $g[4]$ = ④. The output frame vector is {$g[0]$, $g[1]$, $g[2]$, $g[3]$, $g[4]$ and $g[5]$}. The frame width is

N_{even} = 6, which provides useful properties to simplify the slope matching calculations in Section 6.2. The frame processing includes the following steps:

1) Get N_{even} = 6 symbols (amplitude values) g[0], g[1], ⋯ g[5] from the input stream.
2) Calculate scaled g[n] · asinc( ) functions.
3) Evaluate the frame convolution in Equation 11 over the g[3] and g[4] interval.
4) Reconstruct the output waveform over the g[3] and g[4] interval.
5) Advance the frame by 1 symbol time using clock/handshake/enable signals.
6) Goto step 1).

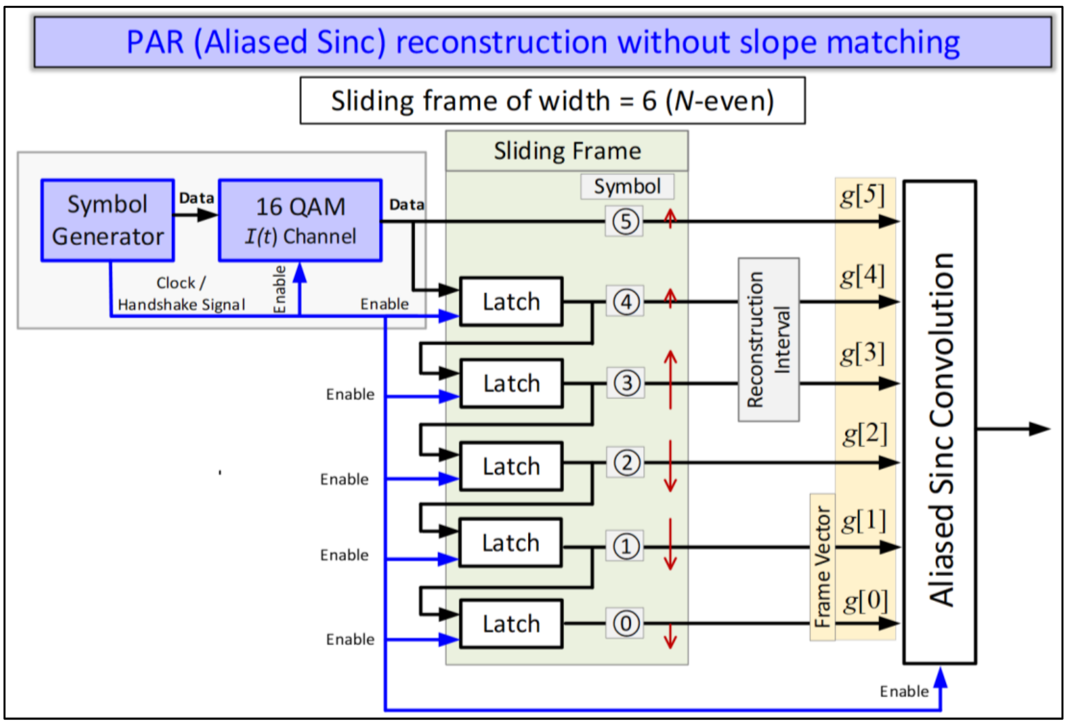


Figure 36. Block diagram for PAR interpolation without slope matching.

Figure 37 shows frame symbols (sample points) as {⓪, ①, ②, ③, ④, and ⑤}. For the ideal periodic case, the reconstructed waveform is exactly equivalent to the original input. The N_{even} = 6 convolution kernel is from Equation 11 and convolution reconstructs the ③–④ interval following the algorithm in Figure 35. The convolution of the asinc kernel with one frame of samples (data symbols are shown as ↑) reconstructs the ③–④ interval in the frame. The reconstruction error in Figure 38 needs to be improved. By adding slope matching between the previous interval and the current interval, the reconstruction error is significantly improved, as described in Section 6.2. The slope matching provides C^1 continuity for the ends of the ③–④ reconstruction interval in the frame.

$$\hat{f}(t) = f_s(t) * h(t) = \sum_{n=0}^{N-1} f[n]\, h(t - nT_s)$$

Convolution equation
$f_s(t)$ = symbol sequence (11)
$h(t)$ = aliased sinc kernel

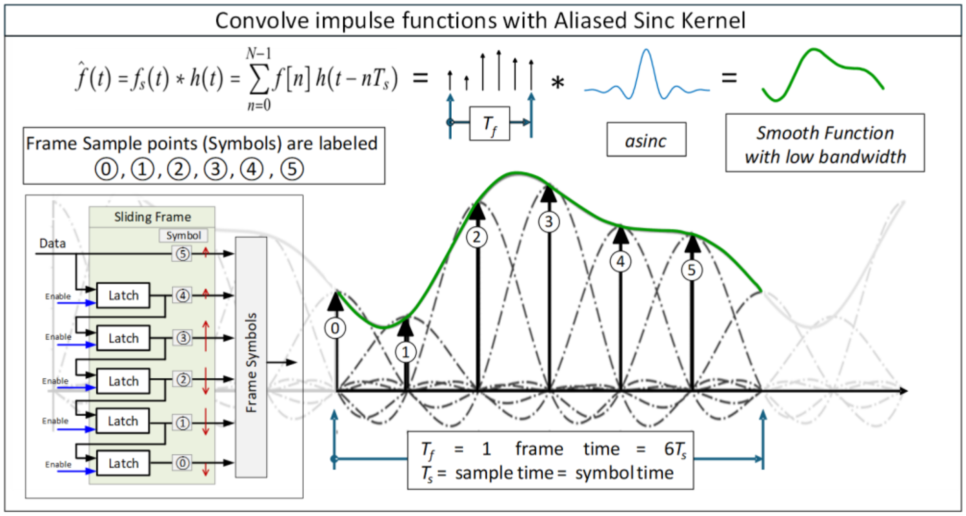


Figure 37. PAR reconstruction without slope matching.

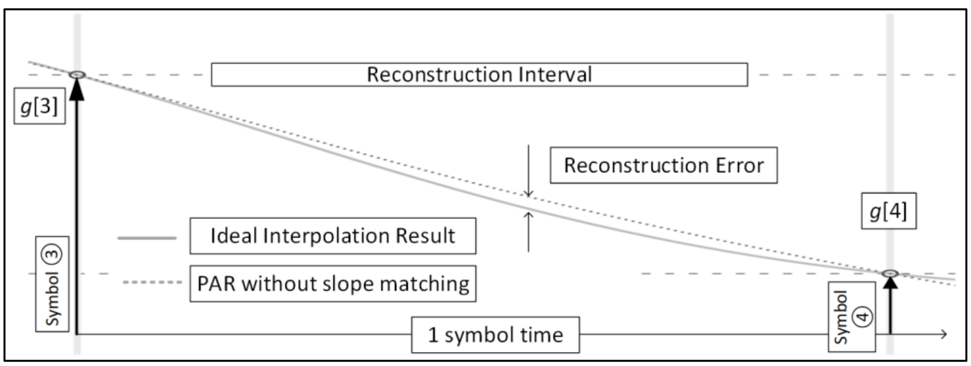


Figure 38. Reconstruction error for PAR interpolation without slope matching.

6.2 Asinc Reconstruction with Slope Matching Introduction

Asinc interpolation and digital modulation takes advantage of the PAR interpolation algorithm to provide better (lower) bandwidth and better (lower-amplitude) sidelobe performance compared to Feher half-cycle raised-cosine interpolation. Figures 39–42 illustrate the PAR algorithm, which includes slope matching across ③–④ reconstruction intervals. Figure 39 presents a high-level flow diagram for the PAR algorithm. Figure 40 focuses on the sliding frame for PAR interpolation. Figure 41 presents a detailed flow diagram for PAR interpolation. Figure 42 compares interpolation errors.

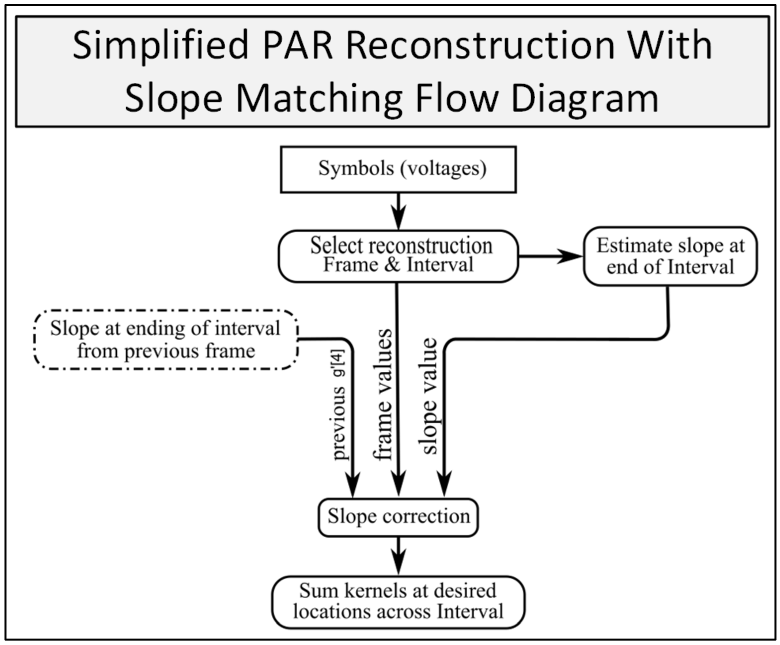


Figure 39. Simplified PAR reconstruction algorithm.

Figure 39 presents a high-level flow diagram for PAR interpolation methodology with slope matching. Here, we apply a slope matching algorithm for the reconstructed signal segments before outputting the reconstructed output signal segment and advancing the frame. We used a reconstruction frame in Figure 40 of width $N_{\text{even}} = 6$ symbols. The slope is estimated. The slope correction block uses inputs from previous frame (slope from previous interval), frame values, and slope estimate from the current interval. The sum kernels block is based on the slope corrections in the frame.

Figure 40 provides a detailed description of the moving frame and reconstruction interval for three reconstruction intervals Frame[$n-1$], Frame[n], and Frame[$n+1$]. The Frame[$n-1$], Frame[n], and Frame[$n+1$] symbols are given by {$g[0]$ = symbol ⓪, $g[1]$ = symbol ①, $g[2]$ = symbol ②, $g[3]$ = symbol ③, $g[4]$ = symbol ④, and $g[5]$ = symbol ⑤}. The moving frame is used to calculate the waveform connecting interval symbols ③ and ④ together at time step n. The example frames illustrated in Figure 40 are six symbols wide. The N_{even} interpolation kernel requires the length to be an even number. The slope matching algorithm takes advantage of the properties of the N_{even} class of asinc interpolation kernels. The $N_{\text{even}} = 6$ case provides excellent interpolation performance at a reasonable computation cost. In Figure 40, the reconstructed interval is the segment ③–④ for all frames shown: Frame[$n-1$], Frame[n], and Frame[$n+1$].

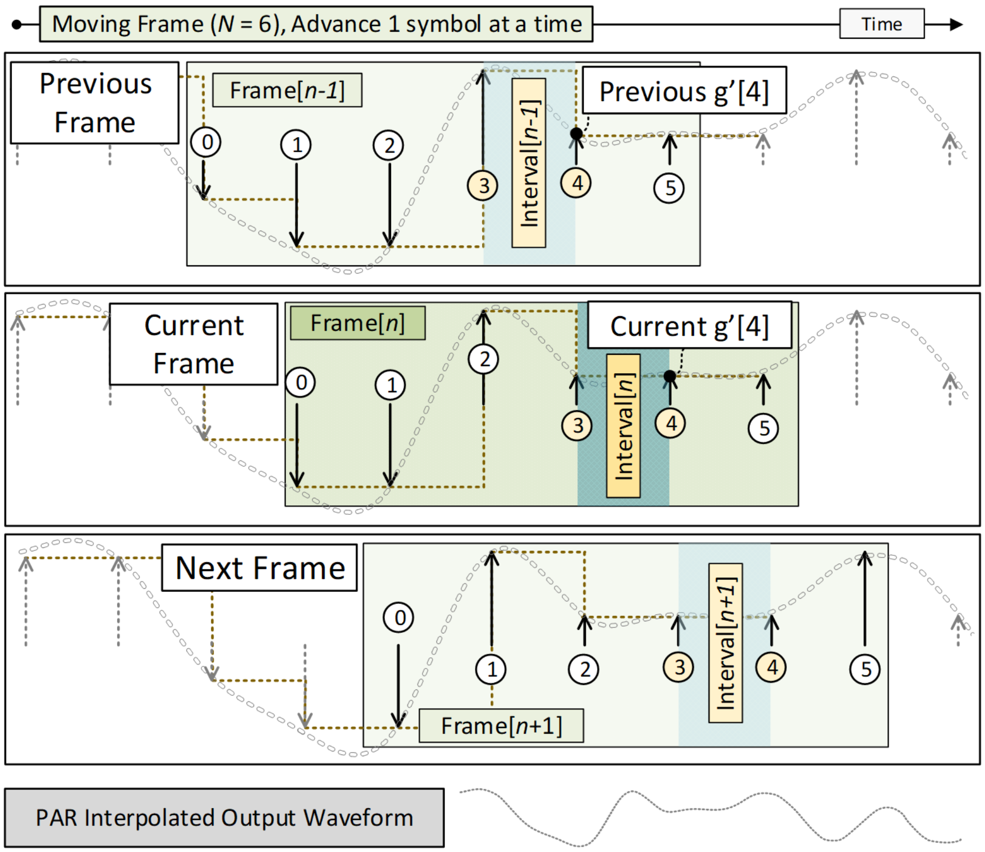


Figure 40. Moving frame used for PAR interpolation.

Figure 41 presents a detailed PAR flow diagram. A frame and interval are selected based on the input symbol stream. The initial frame consists of sample points (symbols) {g[0], g[1], ⋯ g[5]}. The initial interval consists of symbols g[3] and g[4]. The slope is estimated from the frame values shown in Figure 41: {*g*[0] = ⓪, *g*[1] = ①, *g*[2] = ②, *g*[3] = ③, *g*[4] = ④, and *g*[5] = ⑤}. The adjusted slope block computes adjustment values for *g*[0] and *g*[1] for interval ③–④. The final frame values contain the adjusted values for *g*[0] and *g*[1] and the rest of the frame values. Final frame block completes a "convolution" calculated for interval ③–④ using the final frame values.

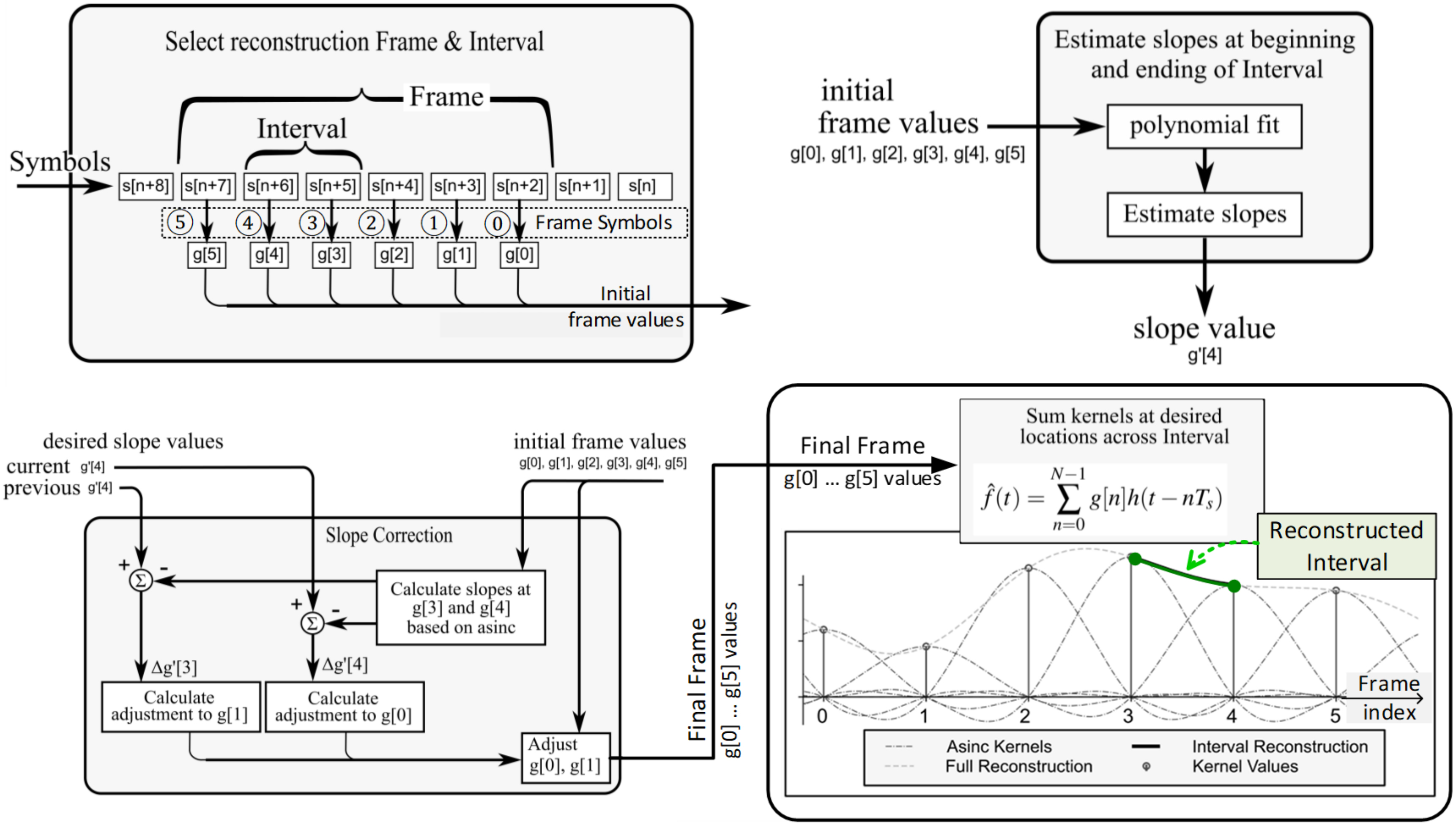


Figure 41. Detailed diagram for PAR interpolation for asinc bandlimited interpolation and digital modulation.

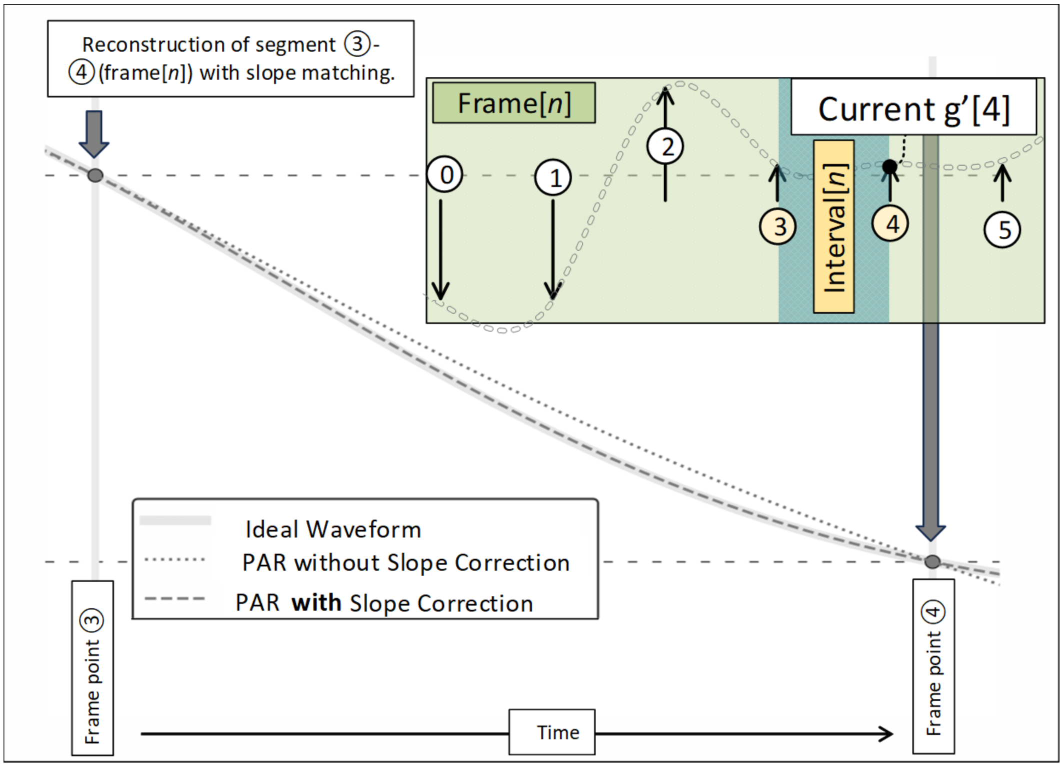


Figure 42. Interpolation error for PAR with slope matching compared with PAR interpolation without slope matching.

Figure 42 compares the interpolation error for PAR with slope matching to PAR without slope matching. PAR with slope matching provides much better interpolation performance.

6.3 Detailed Description of PAR Interpolation

Each step of the PAR procedure in Figure 41 reconstructs the interval between two successive symbols. To perform the calculations, a frame encompassing the interval is selected. For example, in Figure 40, the reconstruction interval between ③ and ④ is reconstructed using the values for sample points (symbols) ⓪–⑤. For reconstruction of the next interval, the frame is moved ahead one symbol time, thus acting like a sliding frame, as shown in Figure 40.

In Figures 40 and 41, the sample (or symbol) points at the endpoints of the reconstruction interval are ③–④(*n*), are accurate in both time and amplitude. The asinc kernels at these locations benefit from this accuracy and require no adjustment. The amplitudes of the remaining kernels, known as *contributing kernels*, at frame(*n*) points g[0] = ⓪, g[1] = ①, g[2] = ②, and g[5] = ⑤ are equal to the symbol values contained in the sliding frame in Figure 40.

The initial values for the contributing kernel amplitudes are directly taken from the symbol stream within the sliding frame. This is illustrated in Figures 40 and 41, where the "Contributing Estimate" at points ⓪, ①, ②, and ⑤ are the respective symbol values (sample points) within the moving frame. In Figure 37, the uniform grid locations are numbered, starting from the left, as ⓪ through ⑤. The reconstruction convolution places an asinc kernel at each of these symbol locations, with an amplitude equal to the sample amplitudes (or symbol values) shown as arrows.

For PAR interpolation with slope matching, the reconstruction algorithm is illustrated in Figure 41. Notice that symbol values at ③ and ④ lie at the endpoints of the reconstruction interval. The asinc function is periodic and infinite in extent. So, the finite sum of these six kernels for ⓪ through ⑤ in Figure 40 and Equation 12 defines the reconstruction for all time.

$$\hat{f}(t) = f_s(t) * h(t) = \sum_{n=0}^{5} f[n]\, h(t - nT_s) \qquad (12)$$

PAR convolution equation for $N_{\text{even}} = 6$ asinc kernel where $h(t)$ is the asinc kernel and $f_s(t)$ is the impulse sequence derived from the symbol stream

With all the symbols defined, as $g[n]$ in Equation 13 and the N_{even} asinc, $h(t)$ from Equation 10, the initial reconstruction (stage 1) can now be expressed as a convolution in Equation 13 following Equations 11 and 12.

$$\hat{g}(t) = \sum_{n=0}^{N-1} g[n]\, h(t - nT_s) \qquad (13)$$

Convolution kernel equation with g[0] = ⓪, g[1] = ①, ⋯ g[5] = ⑤ where ⓪, ①, ⋯ , ⑤ are the symbols in the moving frame in Figure 40 and PAR flow diagram in Figure 41.

Visually, the reconstruction in Figure 37 seems to be a fairly good match to the ideal reconstruction function; however, further improvement is still needed, as shown in Figure 38.

There are numerous possible approaches to improving the accuracy of these reconstructions, but one has proven both efficient and accurate. In Figure 39, given accurate estimates of the output waveform's slope at the endpoints of the reconstruction interval ③-④, the amplitudes of the contributing kernels, ⓪, ①, ②, and ⑤ in the frame (Figure 40), can be adjusted such that the reconstructed slopes match these target slopes at these points (Figure 41). This also ensures C^1 continuity (continuous in the first derivative). This means that the tangent line to the curve does not have any abrupt changes in direction at any point along the curve. In practical terms, this ensures that the curve has a smooth transition without sharp corners or cusps. Recall that the goal of this step of the procedure is to accurately reconstruct the interval between symbols ③ and ④; inaccuracies at the other points in the frame have no effect on the final reconstruction.

Because of the unique properties of the N_{even} asinc in Figure 33 and Equation 14, the procedure for correcting the slope values at the edges of the reconstruction interval ③-④ only requires adjusting the amplitude of two of the contributing kernels in Figure 41.

$$h(t) = \left(\frac{1}{N}\right)\cos\left(\pi\, t/T_p\right)\frac{\sin(\pi\, t/T_s)}{\sin(\pi\, t/T_p)} \qquad \begin{array}{l} N_{\text{even}} \text{ asinc function with } N = 6 \text{ samples} \\ T_s = \text{symbol time, and} \\ T_p = T_f = \text{frame time} = NT_s \end{array} \tag{14}$$

Figure 40 illustrates the moving frame. The frame consists of six symbols shown as impulse functions. The convolution in Equation 13 with the N_{even} asinc in Equation 14 is shown for symbol values ⓪, ①, ②, ③, ④, and ⑤ in Frame[n - 1], Frame[n], and Frame[n + 1] in Figure 40. The slope matching method described in Figure 41 shows how the contributing kernels ⓪, ①, ②, and ⑤ are adjusted to provide slope matching for segment ③-④ as the frame slides one symbol at a time. The slope matching algorithm matches the slope at point ③(Frame[n]) = slope at ④(Frame[n-1]). This slope matching provides C^1 continuity for the output waveform. The updated values for $g[0]$, $g[1]$, $g[2]$, and $g[5]$ in Equation 13 are described in Figure 43.

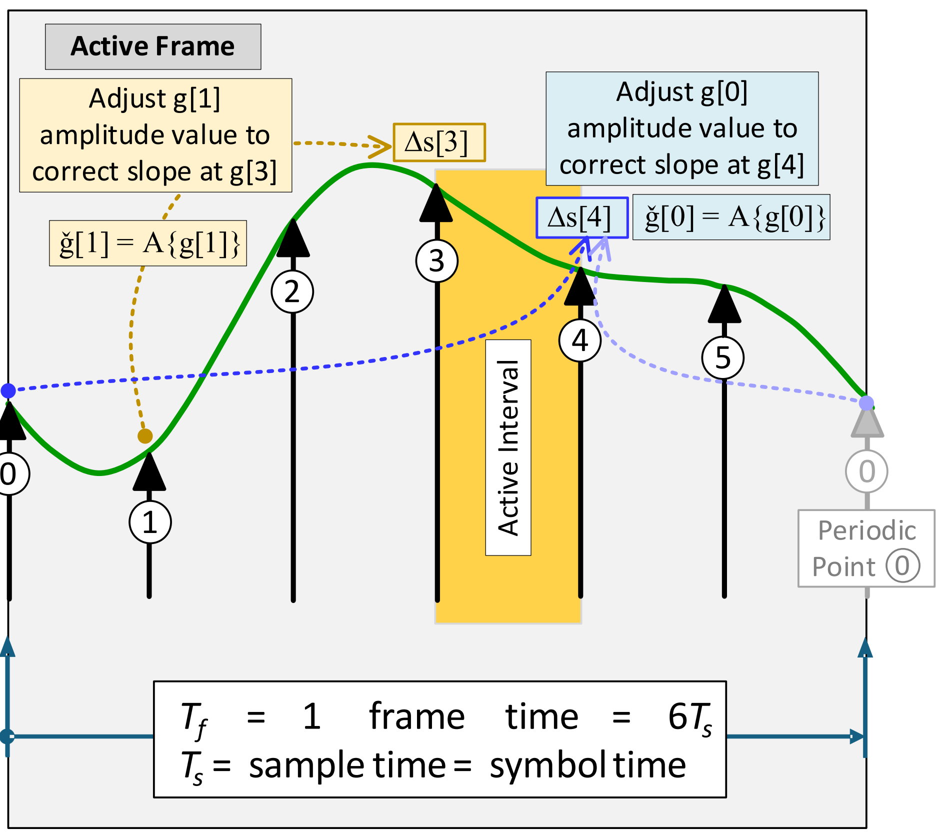


Figure 43. Correcting slope at point g[3] by adjusting amplitude at g[1]. Correcting slope at point g[4] by adjusting amplitude at g[0]. Note g[0] is periodic and repeats at point ⓪.

In Figure 40, consider kernel ①, located at the distance $((N/2) - 1)T_s = 2T_s$ from symbol ③ and $(N/2)T_s = 3T_s$ from symbol ④. Modifying the amplitude of kernel ① in Figure 43 will adjust the slope at symbol ③ without affecting the slope or amplitude at symbol ④. This allows the adjustment of slope at one edge of the reconstruction interval, without affecting the amplitude or slope at the other edge. Similarly, modifying the amplitude of kernel ⓪ can independently adjust the slope at symbol ④, without affecting amplitude or slope at symbol ③. The relationship between the amplitude of these contributing kernels and their influence on the slope at the interval edges is given by the relationship of that kernel amplitude to its slope at that particular edge.

The slope correction calculation makes use of the expression for the slope of the N_{even} asinc function at its zero crossings in Equation 14. The slope of an asinc kernel at its m^{th} zero crossing, which occurs at the distance of $r_m = mT_s$ from its peak, is given in Equation 15. Similar to Equation 14, this formula is indeterminate at positions, $m = qN$, for integer q; however, the limit of the slope at these points can be shown to approach 0, using L'Hôpital's rule.

$$h'(r_m) = \begin{cases} 0\,, & m = qN\,,\, q \in \mathbb{Z} \\ 0\,, & m = qN + \frac{N}{2}\,,\, q \in \mathbb{Z} \\ \frac{\pi}{T_p}\left(\frac{\sin(\pi m/N)\cos(\pi m/N)}{\sin^2(\pi m/N)}\right)(-1)^m\,, & \text{otherwise} \end{cases} \qquad \text{Slope of asinc } N\text{-}_{\text{even}} \text{ kernel} \qquad (15)$$

Obviously, the third expression in Equation 15 can be simplified mathematically; however, using the standard C math libraries for calculation, the results are more accurate if the above expression is used as it is.

For a particular asinc function, characterized by N and T_s, the relationship between its amplitude and its slope at a particular m^{th} zero crossing is linear, given by a fixed coefficient. Thus, calculating the reconstructed slope at any particular symbol value is given by the sum of a handful of multiplications.

The reconstructed slope at any symbol location is the sum of slopes of all kernels at that location. Using Equation 13, and summing over all initial estimates of kernel amplitudes, the initial reconstructed slope at symbol ③ can be expressed in Equation 16.

$$s[3] = g'(t)\Big|_{t=3T_s} = \sum_{n=0}^{N-1} g[n]\frac{\pi}{T_p}\left(\frac{\cos((3-n)\pi/N)}{\sin((3-n)\pi/N)}\right)(-1)^{(3-n)} \qquad \text{Initial reconstructed slope at symbol ③ in Figure 43} \qquad (16)$$

Incremental adjustment of reconstructed slope at one edge of the interval is accomplished by modifying the amplitude of the associated contributing kernel, as described in Equation 16. In Figure 43, the relationship between the kernel amplitude at point ① and its contribution to slope at point ③ is given by Equation 17. In this case, $m = 3 - 1 = 2$.

$s_1[3] = g[1]\frac{\pi}{T_p}\left(\frac{\cos(2\pi/N))}{\sin(2\pi/N))}\right)$	Kernel amplitude at symbol ① and its contribution to slope at symbol ③	(17)

Given these relationships, the adjustment of the reconstructed slope at symbol ③ can be expressed as a modification to the amplitude of kernel ①. The required change in slope is the difference between the target slope, $s_{tgt}[3]$, and the initial reconstructed slope, $s_1[3]$, at this point in Equation 18.

$\Delta_s[3] = s_{tgt}[3] - s_1[3]$	The required change in slope is the difference between the target slope, $s_{tgt}[3]$, and the initial reconstructed slope, $s_1[3]$, at this symbol	(18)

With all other kernels held constant, an adjustment of kernel ①'s amplitude will cause a change in slope at kernel ③. The resulting modification to the reconstructed slope at kernel ③ is equal to that change. Thus, the desired correction in reconstructed slope can be expressed in terms of a modification to the amplitude of kernel ①. Defining $\breve{g}[1]$ as the desired modified amplitude of kernel ①, this relationship can be expressed as Equation 19.

$\Delta_s[3] = (\breve{g}[1] - g[1])\frac{\pi}{T_p}\left(\frac{\cos(2\pi/N)}{\sin(2\pi/N)}\right)$	$\breve{g}[1]$ as the desired modified amplitude of kernel ①	(19)

Using this, the corrected amplitude value at point ① can be calculated as below in Equation 20. This corrects the slope at symbol ③ without changing the slope or amplitude at symbol ④.

$\breve{g}[1] = g[1] + \Delta_s[3]\frac{T_p}{\pi}\left(\frac{\sin(2\pi/N)}{\cos(2\pi/N)}\right)$	This corrects the slope at symbol ③ without changing the slope or amplitude at symbol ④ in Figures 34 and 43.	(20)

Similarly, the slope at kernel ④ can be modified by adjusting the amplitude of symbol ⓪, without altering the waveform at symbol ③. The initial reconstructed slope at symbol ④ is given in Equation 21.

$s[4] = g'(t)\Big\|_{t=4T_s} = \sum_{n=0}^{N-1} g[n]\frac{\pi}{T_p}\left(\frac{\cos((4-n)\pi/N)}{\sin((4-n)\pi/N)}\right)(-1)^{(4-n)}$	The initial reconstructed slope at symbol ④ in Figures 34 and 43.	(21)

The contribution of symbol ⓪ to the reconstructed slope at symbol ④ is given in Equation 22.

$s_0[4] = g[0]\frac{\pi}{T_p}\left(\frac{\cos(4\pi/N)}{\sin(4\pi/N)}\right)$	The contribution of symbol ⓪ to the reconstructed slope at symbol ④	(22)

Once again, the desired change in slope is the difference between the target slope and the initial reconstructed slope in Equation 23.

$\Delta_s[4] = s_{tgt}[4] - s[4]$	The desired change in slope is the difference between the target slope and the initial reconstructed slope	(23)

This change in reconstructed slope can be expressed in terms of the contribution of kernel ⓪. Defining $\breve{g}[0]$ as the desired corrected amplitude of kernel ⓪, the relationship can be expressed in Equation 24.

$\Delta_s[4] = (\breve{g}[0] - g[0])\dfrac{\pi}{T_p}\left(\dfrac{\cos(4\pi/N)}{\sin(4\pi/N)}\right)$	$\breve{g}[0]$ as the desired corrected amplitude of kernel ⓪	(24)

Using Equation 24, the corrected amplitude value at symbol ⓪ can be calculated as below in Equation 25. This corrects the slope at symbol ④ without changing the slope or amplitude at symbol ③.

$\breve{g}[0] = g[0] + \Delta_s[4]\dfrac{T_p}{\pi}\left(\dfrac{\sin(4\pi/N)}{\cos(4\pi/N)}\right)$	This corrects the slope at symbol ④ without changing the slope or amplitude at symbol ③ as shown in Figure 43.	(25)

Figure 42 displays the slope-corrected reconstruction compared to the initial reconstruction (non-slope corrected) in Figures 37 and 38. Slope correction significantly reduces the reconstruction error.

6.4 PAR Interpolation and 16 QAM Simulations

Figure 44 compares time-domain waveforms for Feher to PAR interpolation for 16 QAM. PAR interpolation shows a small, controlled amount of overshoot and a small, controlled amount of intersymbol interference. Feher interpolation is not bandlimited and the zero slope start and endpoints for half-cycle raised-cosine interpolation do not allow for any signal overshoot nor any limited amount of intersymbol interference. The overshoot for PAR interpolation is limited to about one-half of a QAM voltage level.

Figure 45 compares QAM 16 $I(t)$ ($Q(t)$ is the same) channel power spectral density for simulations of half-cycle raised-cosine interpolation and PAR interpolation. PAR interpolation provides 24% lower bandwidth, near zero sidelobes, and a near true low-pass filter response. Figure 45 shows PAR interpolation is significantly better than half-cycle raised-cosine interpolation.

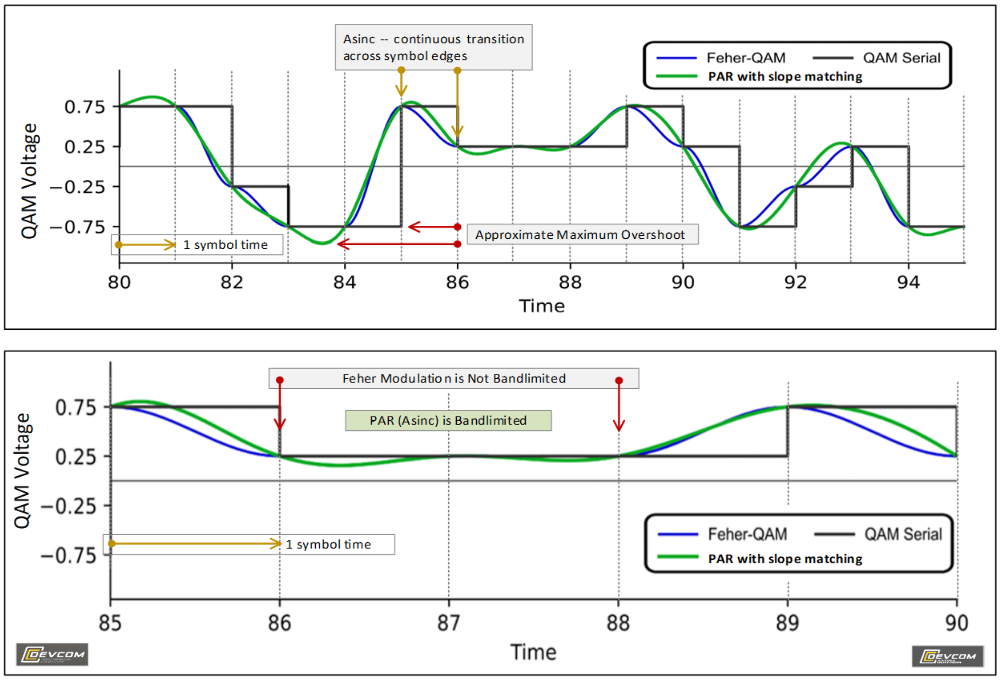


Figure 44. Graphs compare QAM 16 time-domain waveforms for half-cycle raised-cosine interpolation and PAR (asinc) interpolation. PAR interpolation is bandlimited and provides a smoother curve between symbol points. At each symbol start time, both Feher and PAR interpolation equal the symbol value.

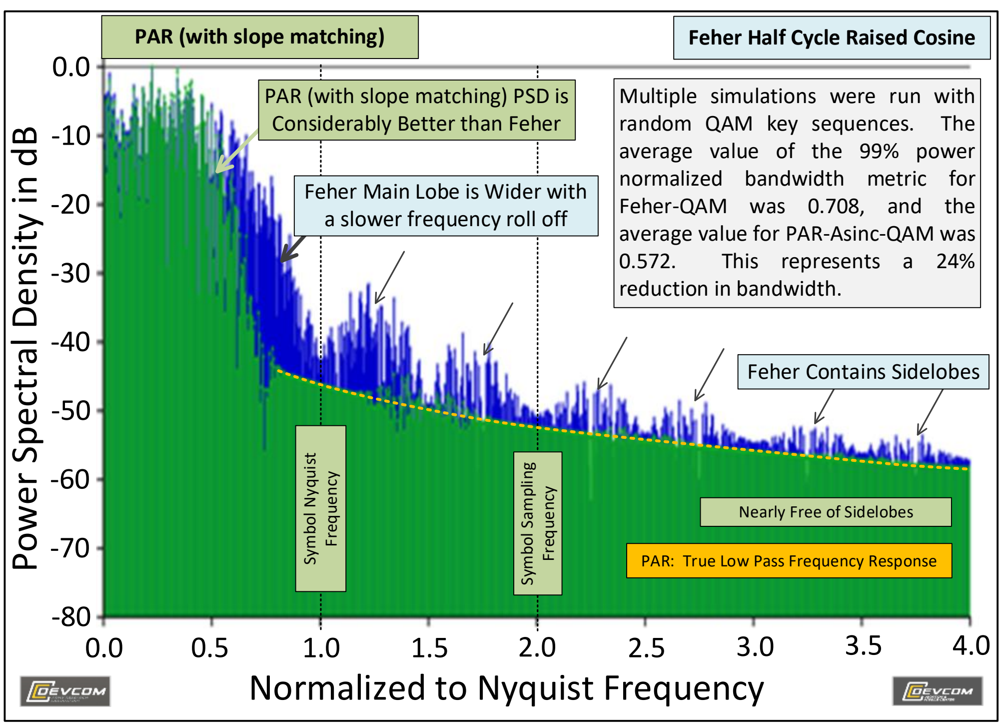


Figure 45. QAM 16 *I*(*t*) (*Q*(*t*) is the same) channel power spectral density for half-cycle raised-cosine interpolation compared to PAR interpolation. PAR interpolation provides 24% improvement (reduction) in bandwidth and significantly improved sidelobes. PAR interpolation also offers a near true low-pass filter response outside the channel bandwidth.

6.5 Demodulation

A conventional sampling receiver[15–20] can be used to demodulate standard QAM, Feher QAM, and PAR interpolated QAM. Various prior art demodulation synchronization techniques can be used for carrier recovery.[15–20] By sampling at the start of each symbol, the amplitude (voltage, power, current, flux, etc.) is the same value as the symbol. Figure 46 shows a sampling demodulator. Receiver provides a modulated waveform. Complex oscillator and complex mixer convert the receiver's output signal to baseband. Low-pass filters create in-phase, $I(t)$, and quadrature phase, $Q(t)$, signals. The sample and hold blocks are sampled by impulse train block. The blocks provide the $a(t)$ and $b(t)$ QAM values in Equation 8. Merge block combines the $a(t)$ and $b(t)$ values to create the output symbol stream.

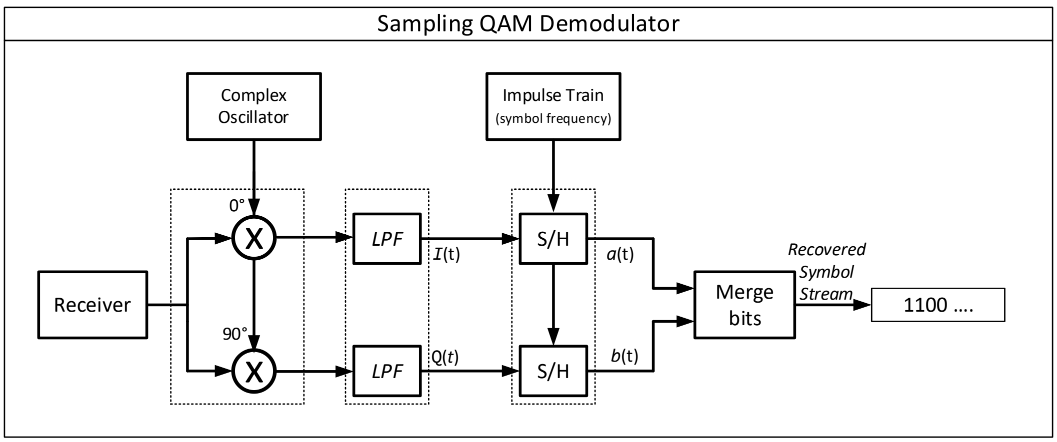


Figure 46. A conventional sampling receiver.

Figure 47 illustrates a more complex demodulation prior art case for QAM 16. QAM 16 $I(t)$ generates four symbol streams. The current symbol N is being decoded. The previous symbols N-5, N-4, ⋯ N-1 are known. The symbol streams represent the four possible symbol streams for the $I(t)$ (or $Q(t)$) channel in QAM 16 in Figure 47(a). For standard QAM, the four symbol streams are compared to the receive stream in Figure 47(b). The symbol with the highest comparison metric is selected as the received symbol. For Feher QAM 16 in Figure 47(c), the four symbol streams are compared to the receive stream. The symbol with the highest comparison metric is selected as the received symbol.

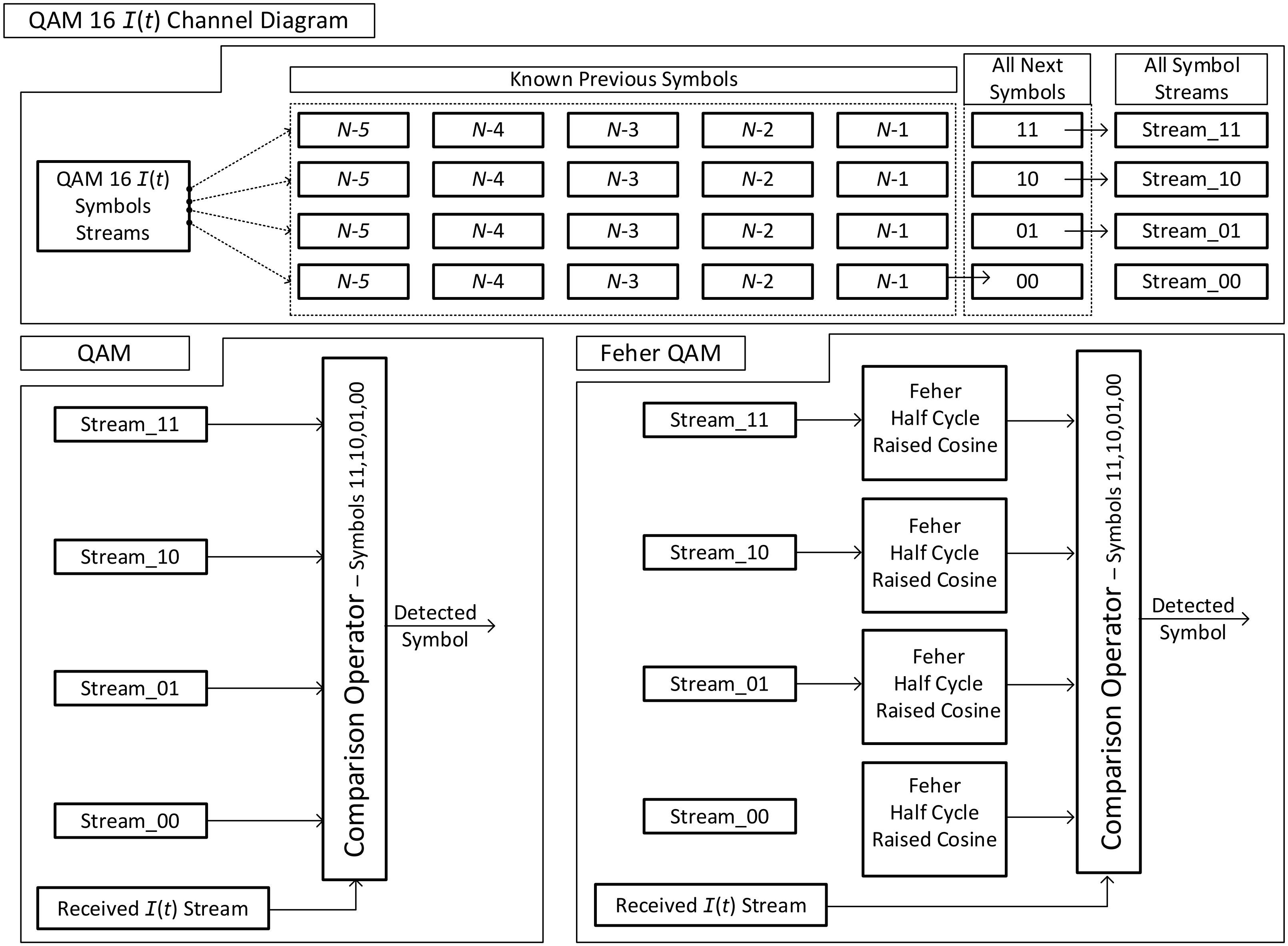


Figure 47. Symbol stream comparison demodulation method.

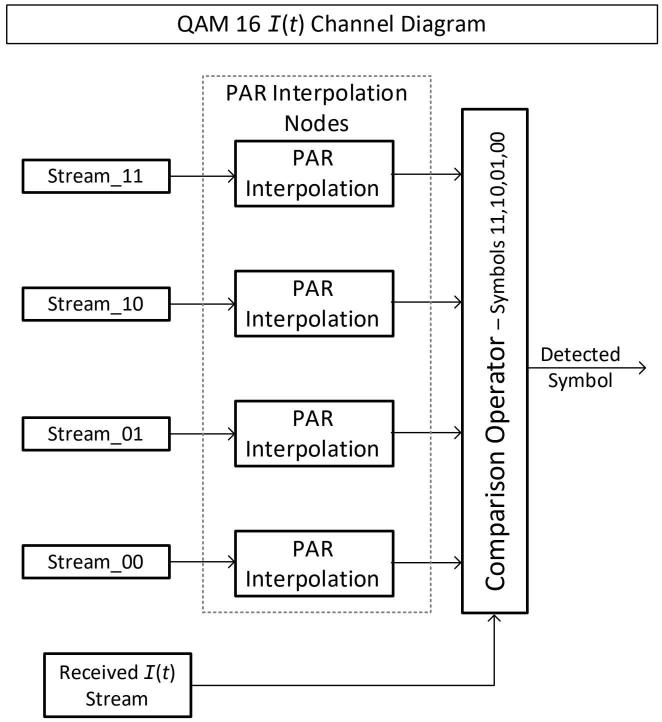


Figure 48. PAR interpolation symbol stream comparison demodulation method.

Figure 48 adapts Figure 47 to PAR interpolation. The Feher blocks are replaced with PAR interpolation blocks. Symbol streams are interpolated by PAR interpolation blocks. The PAR interpolation streams are compared to the input stream. The symbol stream with the highest comparison metric is the detected symbol.

7. Conclusion

As illustrated in Figure 45, asinc interpolation and PAR interpolation[1,2] offers a substantial improvement in bandwidth and sidelobe rejection over half-cycle raised-cosine modulation. A comparison of the spectra of Feher-QAM and PAR-QAM modulation schemes is shown in Figure 45. In this example, the 99% power normalized bandwidth metrics are as follows: Feher-QAM is 0.71 and PAR-QAM is 0.52. Multiple simulations were run with random QAM symbol sequences as in the four-level case example. The average value of the 99% power normalized bandwidth metric for Feher-QAM was 0.708, and the average value for PAR-QAM was 0.572. This represents a 24% reduction in bandwidth.

The asinc interpolation could be very beneficial for any digital communications, radio communications, wireless internet, cable modems, radar, sonar, and optical communications links.

8. References

1. Crowe WM, Jungwirth PW. Continuous time systems disruptive signal processing and accurate real time signal reconstruction. In: Blowers M, Bharadwaj R, Wysocki BT, editors. Proc SPIE 13058, Disruptive Technologies in Information Sciences VIII, 1305815; 2024 Apr 21–25; National Harbor, MD. https://doi.org/10.1117/12.3014164

2. Crowe WM, Jungwirth PW. Introduction to the progressive asinc reconstruction (PAR) technique for accurate real-time reconstruction of level-crossing (LC) signals. DEVCOM Aviation and Missile Center (US); 2026 Jan. Report No.: TR-FCDD-AMT-25-13. DTIC Accession No: AD1347247.

3. Crowe WM, Jungwirth PW, inventors. Aliased sinc bandlimited interpolation and digital modulation. United States patent application no. 19/554,382. Submitted 2026 Mar.

4. Tsividis Y. Continuous-time digital signal processing. Electron Lett. 2003;39(21):1551.

5. Li Y, Shepard K, Tsividis Y. Continuous-time digital signal processors. 11th IEEE International Symposium on Asynchronous Circuits and Systems; 2005 Apr; New York, NY. p. 138–143. https://doi.org/10.1109/ASYNC.2005.15

6. Tsividis Y. Event-driven data acquisition and digital signal processing – a tutorial. IEEE Trans Circuits Syst II Express Briefs. 2010;57(8).

7. Vezyrtzis C, Tsividis Y. Processing of signals using level-crossing sampling [presentation]. ISCAS 2009. Proceedings of IEEE International Symposium on Circuits and Systems; 2009 May 24–27; Taipei, Taiwan. IEEE; c2009. p. 2293–2296. https://doi.org/10.1109/ISCAS.2009.5118257

8. Kurchuk M. Signal encoding and digital signal processing in continuous time [dissertation]. Columbia University; 2011.

9. Aeschlimann F, Allier E, Fesquet L, Renaudin M. Asynchronous FIR filters: towards a new digital processing chain. In: Proceedings of the 10th International Symposium on Asynchronous Circuits and Systems (ASYNC'04); 2004 Apr 19–23; Crete, Greece. IEEE; c2004. p. 198–206.

10. Qaisar S, Hussain S. Arrhythmia diagnosis by using level-crossing ECG sampling and sub-bands features extraction for mobile healthcare. Sensors. 2020;20(8):2252.

11. Zhao Z et al. CT-DSP based controller for high-frequency DC-DC converters. APEC '07. Twenty-Second Annual IEEE Applied Power Electronics Conference and Exposition; 2007 Feb 25–Mar 1; Anaheim, CA. IEEE; c2007. p. 882–886.

12. Jungwirth PW, Crowe WM. Continuous time digital signal processing and signal reconstruction. 2023 IEEE 13th Annual Computing and Communication Workshop and Conference (CCWC); 2023 Mar 8–11; Las Vegas, NV. IEEE; c2023. p. 1205–1211. https://doi.org/10.1109/CCWC57344.2023.10099213

13. Jungwirth PW, Crowe WM. Continuous-time pipeline level crossing analog-to-digital converter and continuous-time software-reconfigurable radio architecture. DEVCOM Army Research Laboratory (US); 2022 July. Report No.: ARL-TR-9497.

14. Jungwirth PW. Dual zero crossing analog-to-digital conversion [dissertation]. University of Alabama in Huntsville; 2005 Aug.

15. Sari H, Moridi S. New phase and frequency detectors for carrier recovery in PSK and QAM systems. IEEE Trans Commun. 1988;36(9):1035–1043. https://doi.org/10.1109/26.7515

16. Kurakake T, Nakamura N, Oyamada K. A blind 1024-QAM demodulator for cable television. International Zurich Seminar on Communications; 2004 Feb 18–20; Zurich, Switzerland. IEEE; c2004. p. 136–139. https://doi.org/10.1109/IZS.2004.1287407

17. Lyon D. Envelope-derived timing recovery in QAM and SQAM systems. IEEE Trans Commun. 1975;23(11):1327–1331. https://doi.org/10.1109/TCOM.1975.1092720

18. Baccetti B, Bellini S, Filiberti G, Tartara G. Full digital adaptive equalization in 64-QAM radio systems. IEEE J Sel Areas Commun. 1987;5(3):466–475. https://doi.org/1109/JSAC.1987.1146547

19. Issman E, Webb WT. Carrier recovery for 16 level QAM in mobile radio. IEEE Colloquium on Multi-Level Modulation Techniques and Point-to-Point and Mobile Radio; 1990 Mar 19; London, England. p. 9/1–9/8.

20. Franks L. Carrier and bit synchronization in data communication – a tutorial review. IEEE Trans Commun. 1980;28(8):1107–1121. https://doi.org/10.1109/TCOM.1980.1094775

21. Roder H. Amplitude, phase, and frequency modulation. Proceedings of the Institute of Radio Engineers. 1931;19(12):2145–2176. https://doi.org/10.1109/JRPROC.1931.222283

22. Armstrong EH. A method of reducing disturbances in radio signaling by a system of frequency modulation. Proceedings of the IRE. 1936;24(5):689–740. https://doi.org/10.1109/JRPROC.1939.228061

23. Crosby MG. Communication by phase modulation. Proceedings of the IRE. 1936;27(2):126–136. https://doi.org/10.1109/JRPROC.1939.228061

24. Feher K, inventor; Feher K, assignee. Filter. United States patent US4339724A. 1982 July 13.

25. Simon MK, Yan T-Y. Performance evaluation and interpretation of unfiltered Feher-patented quadrature-phase-shift keying (FQPSK). National Aeronautics and Space Administration (US); 1999 May 15. TMO Progress Report No.: 42-137. p. 1–29.

26. Simon M, Divsalar D. Further results on a reduced-complexity, highly power/bandwidth-efficient coded Feher-patented quadrature-phase-shift-keying system with iterative decoding. National Aeronautics and Space Administration (US); 2001 Aug 15. IPN Progress Report No.: 42-146.

27. Lili G, Zhou B, Sun Z, Gao F. Symmetric raised cosine keying modulation and performance analysis. Proceedings of the 2011 IEEE International Conference on Computer Science and Network Technology (ICCSNT 2011); 2011 Dec 24–26; Harbin, China. IEEE; c2011. p. 88–91. https://doi.org/10.1109/ICCSNT.2011.6181915

28. Simon MK, Arabshahi P, Srinivasan M. Trellis-coded quadrature-phase-shift keying (QPSK) with variable overlapped raised-cosine pulse shaping. National Aeronautics and Space Administration (US); 1999 Feb 15. TMO Progress Report No.: 42-136.

29. Austin MC, Chang MU. Quadrature overlapped raised-cosine modulation. IEEE Trans Commun. 1981;29(3):236–249. https://doi.org/10.1109/TCOM.1981.1094986

30. Jungwirth PW. Feher modulation 16 QAM. 34th ARRL and TAPR Digital Communications Conference (DCC); 2015 Oct 9–11; Chicago, IL.

List of Symbols, Abbreviations, and Acronyms

1-/2-D	One-/Two-Dimensional
ADC	Analog-to-Digital Converter
ARL	Army Research Laboratory
asinc	Aliased Sinc
ASK	Amplitude Shift Keying
ΔΣ	Asynchronous Delta Sigma
AZOH	Asynchronous Zero-Order Hold
CT	Continuous Time
CT-DSP	Continuous Time-Digital Signal Processing
DEVCOM	U.S. Army Combat Capabilities Development Command
DSP	Digital Signal Processing
ECG	Electrocardiogram
FSK	Frequency Shift Keying
LCD	Level-Crossing Delta
LCS	Level-Crossing Sampling
OOK	On/Off Keying
PAR	Progressive Asinc Reconstruction
PSK	Phase Shift Keying
QAM	Quadrature Amplitude Modulation
sinc	Cardinal Sine
SNR	Signal-to-Noise Ratio
WKS	Whittaker–Kotel’nikov–Shannon

1 (PDF)	DEFENSE TECHNICAL INFORMATION CTR DTIC OCA
1 (PDF)	DEVCOM ARL FCDD RLB CB TECH LIB