ARL-TR-9497 • JULY 2022

DEVCOM ARMY RESEARCH LABORATORY

Continuous-Time Pipeline Level-Crossing Analog-to-Digital Converter and Continuous-Time Software-Reconfigurable Radio Architecture

by Patrick Jungwirth and W Michael Crowe

NOTICES

Disclaimers

The findings in this report are not to be construed as an official Department of the Army position unless so designated by other authorized documents.

Citation of manufacturer's or trade names does not constitute an official endorsement or approval of the use thereof.

Destroy this report when it is no longer needed. Do not return it to the originator.

ARL-TR-9497 ● JULY 2022

Continuous-Time Pipeline Level-Crossing Analog-to-Digital Converter and Continuous-Time Software-Reconfigurable Radio Architecture

Patrick Jungwirth
DEVCOM Army Research Laboratory

W Michael Crowe
DEVCOM Aviation and Missile Center

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. REPORT DATE *(DD-MM-YYYY)*	2. REPORT TYPE	3. DATES COVERED (From - To)
July 2022	Technical Report	1 February 2021–1 May 2022

4. TITLE AND SUBTITLE
Continuous-Time Pipeline Level-Crossing Analog-to-Digital Converter and Continuous-Time Software-Reconfigurable Radio Architecture

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

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

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
DEVCOM Army Research Laboratory
ATTN: FCDD-RLC-NC
Aberdeen Proving Ground, MD 21005

8. PERFORMING ORGANIZATION REPORT NUMBER
ARL-TR-9497

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
Approved for public release: distribution unlimited.

13. SUPPLEMENTARY NOTES

14. ABSTRACT
The field of signal processing consists of several areas, each of which has its own set of advantages and limitations. A new subfield of signal processing called continuous-time systems has some unique signal-processing characteristics. To better understand continuous-time systems, a review of analog signal processing and conventional digital signal processing (DSP) is presented first. Continuous-time systems behave similar to analog signal-processing systems with the benefits of discrete amplitude levels from DSP. Continuous-time systems also have the benefits of analog signal processing (power is "proportional" to slope of the input signal) without the limitations of conventional DSP: frequency aliasing, quantization error, and control system lag. Continuous-time systems with greater than 100 dB signal-to-noise ratio have been demonstrated; such systems are more energy efficient than DSP. In this technical report, a patent-pending continuous-time pipeline analog-to-digital converter (CT-ADC) is presented. The CT-ADC overcomes several design limitations present in conventional clocked ADCs. The CT-ADC topology can also be used in a conventional clocked pipeline ADC to reduce the complexity of error correction. A patent-pending continuous-time software-reconfigurable radio (CT-SRR) based on the CT-ADC is also described. The CT-SRR architecture overcomes dynamic range and intermodulation distortion problems present in conventional software-reconfigurable radios.

15. SUBJECT TERMS
continuous-time systems, sampling theory, analog, discrete time, digital, analog-to-digital conversion, software-reconfigurable radio, Network and Computational Sciences

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			Patrick Jungwirth
Unclassified	Unclassified	Unclassified	UU	68	19b. TELEPHONE NUMBER (Include area code) (410) 278-6174

Contents

List of Figures

List of Tables

1. Introduction

The field of signal processing consists of several areas, each of which has its own set of advantages and limitations. When planning the design of a signal-processing system, the choice of the signal-processing subfield is the foundation for the design and carries with it the assumptions, advantages, and limitations of that subfield. Analog signal processing and digital signal processing (DSP) are mature subfields that account for most current designs.

Continuous-time digital signal processing (CT-DSP) is an emerging subfield with some known advantages. For size, weight, and power (SWaP), continuous-time systems have demonstrated up to a 10 times reduction in power.[1] A continuous-time level-crossing analog-to-digital converter (ADC) application-specific integrated circuit (ASIC) has demonstrated greater than 100 dB signal-to-noise ratio (SNR).[2] A patent-pending continuous-time pipeline ADC has demonstrated greater than 10-dB improvement in intermodulation distortion[3] over a conventional software-defined radio architecture.

CT-DSP has the benefits of analog signal processing (power is "proportional" to the slope of the input signal) without the limitations of conventional digital signal processing: frequency aliasing, quantization error, and control system lag. CT-DSP technology challenges include unclocked systems, asynchronous design techniques, nonuniform sampling, and still being an experimental technology area. With the development of asynchronous designs for field-programmable gate arrays (FPGAs),[4–7] applications for CT-DSP are becoming practical for systems engineering.

All models have built-in assumptions and limitations. In the words of G Box, "All models are wrong, but some are useful."[8] A compact disc[9–11] has a 96-dB SNR. What are the assumptions used to calculate the SNR? We tend to learn the central concepts of a domain and then move on; per J von Neumann, "In mathematics, you don't understand things. You just get used to them."[12] The importance of some discoveries was not well understood at the time. In 1850, Michael Faraday made a very astute observation about the future of electricity to William Gladstone, Chancellor of the Exchequer (Britain's minister of finance): "Why, sir, there is every probability that you will soon be able to tax it!"[13]

This technical report focuses on continuous-time analog-to-digital conversion and a software-reconfigurable radio architecture. To get past the conventional DSP mindset, a review of analog signal processing and conventional DSP is presented. The Introduction provides a foundation for the advantages of continuous-time systems for analog-to-digital conversion and software-reconfigurable radio architectures.

2. Signal Processing Introduction

Most analog signal processing is based on the computational model of a linear time-invariant (LTI) system. Discrete time is a computational model based on LTI systems and ideal Shannon sampling. The impulse response of a LTI system does not change with time. Standard linear signal processing techniques are used for both analog- and discrete-time signal processing.

Continuous time (CT)* is shorthand for continuous time and "exact" quantized amplitude. CT is a computational model based on analog signal processing, linear time systems, and equivalence time (when the input signal exactly equals a discrete voltage level). Continuous-time systems are not linear time invariant; they are linear time variant. Linear system theory can be applied to continuous-time systems; however, the impulse response is time varying.

To explain CT signal processing, a review of DSP is helpful. The theory and system design techniques for DSP are mature and have proven to be successful for a wide range of problems. Unfortunately, it is easy to take the mindset that DSP can solve all signal processing problems. There are limitations to DT systems, and some of them can be mitigated by using CT systems. DSP systems sample at fixed times and quantize the signal's amplitude. The quantization produces a noise floor that is dictated by the amplitude resolution in bits. Most DSP techniques rely on the theoretical underpinnings of the discrete Fourier transform (DFT), which assumes that *all* signals are periodic. This limits the signal of interest to a set of discrete frequency components, and Shannon sampling aliases any part of the signal that is above the Nyquist frequency (half of the sampling frequency). One needs to understand the concepts of time limited and band limited. A time-limited signal's Fourier series consists of harmonics at $f_n = nf_0$, where the fundamental frequency is $f_0 = \frac{1}{T_w}$, and T_w is the time width of the window. All Fourier series signals are periodic. No real-world signal is bandlimited. There are *almost* band-limited signals that are modeled as bandlimited. For DSP, an almost bandlimited signal can result in a higher noise floor. Because CT-DSP only samples signals at the equivalence times (exact time the input signal equals a discrete voltage level), the samples do not have quantization error nor frequency aliasing.

Figure 1 illustrates signal processing architectures.[14] Signal processing architectures cover continuous to discrete *time* and continuous to discrete *amplitude*. The boundaries between continuous and discrete for time and amplitude offer unique properties. The boundary regions are where signal classes overlap.[15]

* Continuous time, as defined by Tsividis,[14] is written here as "CT"; "continuous time" (spelled fully) refers to any time value on a number line.

For example, frequency modulation (FM) is both CT and asynchronous digital. For high sampling frequencies, FM and a binary waveform look the same. Pulse amplitude modulation can be continuous or discrete. Pulse amplitude modulation fits in analog signal-processing class (continuous time and continuous amplitude) and continuous-time signal-processing class. The boundary regions can be used to create useful approximations. For example, Li et al.[16] used a tapped delay line with a very large oversampling factor to synthesize finite impulse response filters for continuous-time systems.

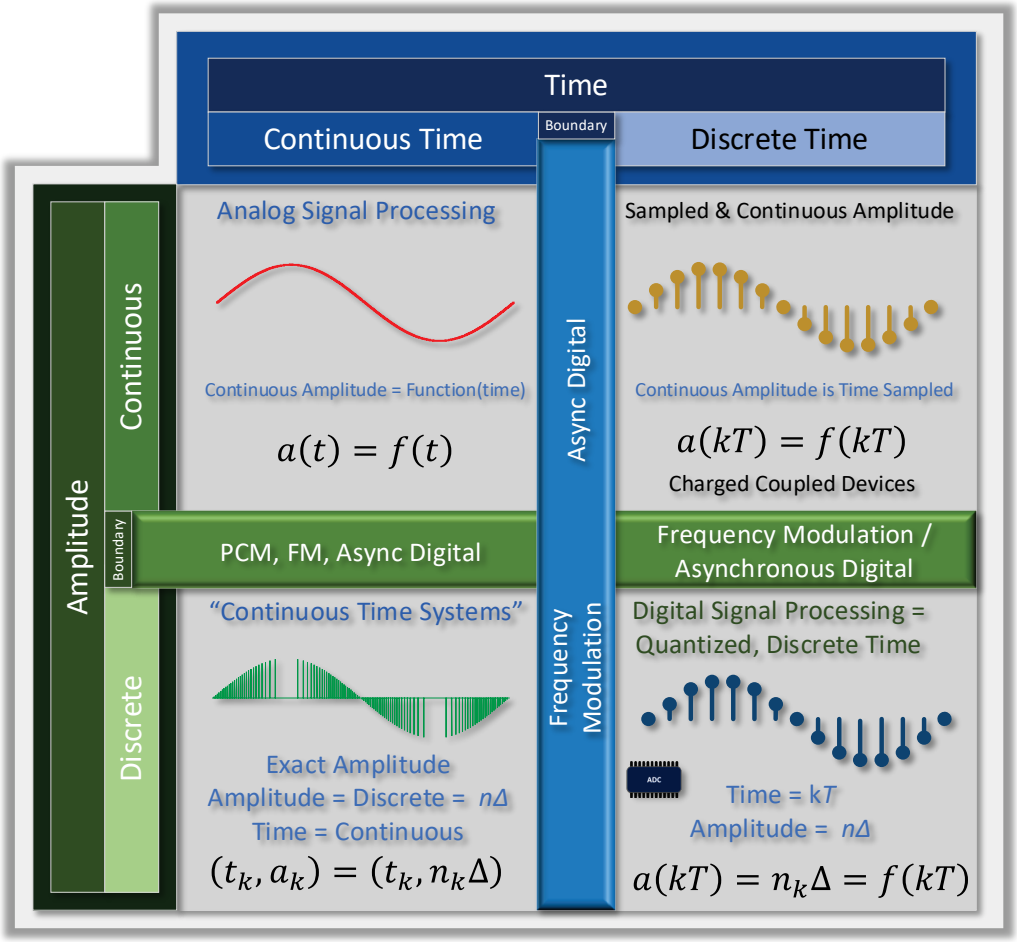


Fig. 1 Signal processing architectures. There are four signal processing domains.[14] Continuous and discrete for both time and amplitude. The boundary regions offer unique signal processing properties.[15] Discrete time is characterized by sample times = kT. Digital is quantized amplitude (amplitude = $n\Delta$) and sample times = kT. For continuous-time signals, time is continuous (not quantized). Discrete-amplitude continuous time, often shortened to continuous time, is characterized by equivalence times, the times, t_k, when the input signal equals a discrete value. At the equivalence times, the input signal exactly equals a discrete amplitude; there is no quantization error compared to digital signal processing. Continual signal processing is defined as exact "discrete" amplitude (no amplitude quantization error) and continuous time (time is not quantized).

In Fig. 2, conventional discrete time (uniform time steps) is compared to continuous-time quantized amplitude (nonuniform time steps, level-crossing sampling for quantized amplitude). For discrete time, each sample is a fixed time step apart (sample times = kT). For a continuous-time signal, the time steps are nonuniform and the "sampling frequency" is proportional to the input signal's slope. It self-adapts to the input signal's characteristics, and it is energy efficient.

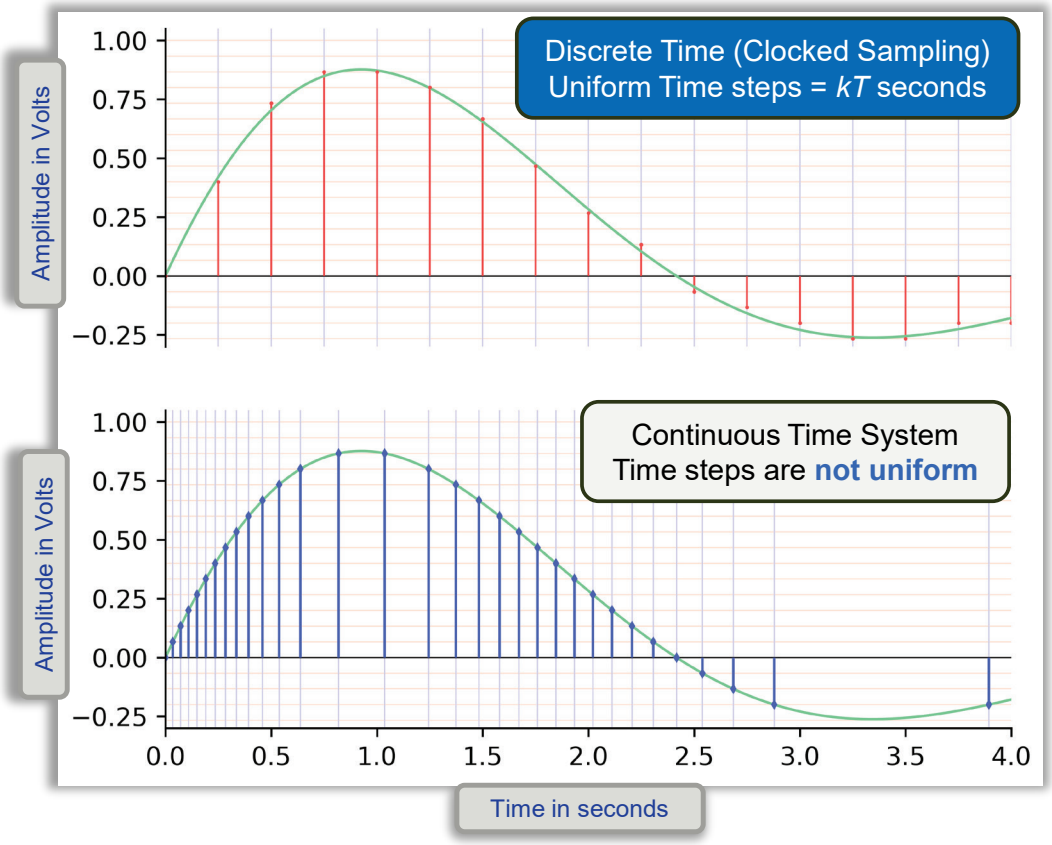


Fig. 2 Comparison of discrete time (uniform time steps) to continuous time (nonuniform time steps). For discrete time, each sample is a fixed time step apart (sample times = kT). For a continuous-time signal, the "sampling frequency" is proportional to the input signal's slope. The distance between samples is not constant (nonuniform time steps). The sampling frequency self-adapts to the input signal, and it is more energy efficient than DSP.

2.1 Analog Signal Processing

Figure 1 illustrates signal processing architectures. Analog signal processing is limited by device linearity and system noise power. Analog signal processing has its origins in the thermionic valve (tube) invented in 1908 by De Forest[17] and the superheterodyne (superhet)[18] receiver invented in 1918 by Armstrong.[19] The superhet used analog multiplication to convert the received radio frequencies to a much lower intermediate frequency for further signal processing. In 1927, Black

invented the negative feedback loop,[20] and in 1934 he published the foundational paper on negative feedback amplification. Philbrick developed a two-tube operational amplifier in 1948.[21]

Shannon illustrated how pulse code modulation overcomes system noise by trading noise reduction for quantization error. In 1949, Shannon showed how quantization can be used to overcome noise for an arbitrary long (unlimited length) telephone cable.[22] The National Television System Committee (NTSC) television standard for RGB television,[23,24] retired in 2010, was the most complex analog signal in widespread use.

2.2 Discrete-Time Processing

Discrete-time systems process signals that are uniformly sampled in time, with amplitudes which may be either continuous or quantized. A charge-coupled device (CCD) is a discrete-time device with a continuous amplitude. A CCD is an array of capacitors. Stored charge can be transferred from one CCD cell to a connected CCD cell. W Boyle and G Smith invented the CCD camera in 1969 at Bell Labs.[25] By the 1970s, CCDs were being used for common signal processing applications,[26,27] but by the 1980s, digital signal processing had surpassed analog signal processing.

2.3 Digital Signal Processing

Digital signal processing is a quantized-amplitude discrete-time signal: DSP = (quantized time = kT and quantized voltage or current = $n\Delta$). According to the IEEE, 1948 was the year several key developments opened the doors to create what today is called digital signal processing[28]:

- Shannon published the paper "A Mathematical Theory of Communication."
- Oliver, Pierce, and Shannon published the paper "The Philosophy of PCM."
- Bartlett and Tukey developed spectrum estimation techniques.
- Bell Labs announced the development of the transistor.
- Manchester University built the first operational stored program computer.
- Bennett published his paper on the quantization noise model.

In DSP, an analog signal is captured by sampling the signal at each clock tick and assigning it a quantization level. The clock operates at a fixed frequency, thus the samples are uniformly spaced in time. During this digitization process, three detrimental things happen. The frequency domain is folded, causing any out-of-band frequency components, including noise, to be frequency aliased into the

passband. The value of each sample is forced to take on the nearest round-down quantization level, resulting in quantization error. The Nyquist frequency component ($f_s/2$) of the original signal is corrupted. All these sampling limitations can be mitigated, but at a cost.

Shannon reconstruction of a uniformly sampled signal provided the basis for treating these systems as LTI. Signal reconstruction assumed an infinitely long signal, but real-world signals are limited in time. Therefore, reconstruction is more complex than Shannon's model. Typically, it is assumed that the input signal is periodic. A periodic signal has the advantage of discrete frequency lines. Efficient signal processing techniques like the fast Fourier transform (FFT) can be applied. The assumption of periodicity also carries the limitation that the signal is now limited to the class of signals that can be represented as a sum of complex exponentials (discrete frequency components or spectra lines).

Quantization error is a nonlinear effect, but Bennett demonstrated how it could be modeled as random noise in a linear-time invariant system.[29] He expressed it in terms of the number of quantization bits, n, relative to a full-scale signal. The maximum SNR relative to a full-scale sinusoid is SQNR* = $6.02n + 1.76$ dB. This equation allows system designers to specify the quantization step based on requirements for SNR and dynamic range.

Large-scale networking and digital signal processing started in the 1950s. During World War II, the British created the Chain Home radar stations to monitor incoming attacks from Germany.[30] In the 1950s, the Air Force developed the Semi-Automatic Ground Environment (SAGE) air defense system[31] radar network to monitor the North Pole for incoming bombers. SAGE was the first large scale interconnected network of system-of-systems. Starting on December 25, 1955, NORAD has tracked Santa Claus on his worldwide tour.[32] Today, pervasive computing, software-defined everything, and the Internet of Things (IoT) are rapidly creating intelligent networked everything.

2.4 Continuous-Time Signal Processing = Level-Crossing Analog Signal Processing

CT-DSP as defined by Tsividis[1] is continuous in time and discrete in amplitude. CT-DSP is "exact" discrete amplitude (no DSP quantization error) and continuous in time. Digital is defined as quantized amplitude and discrete time, and continual is defined as exact quantized amplitude and continuous time. Continuous-time signal processing properties are presented in the following list:

* Signal-to-quantization-noise ratio.

- CT signal = exact quantized amplitude and continuous time.
- Equivalence time when the input signal exactly equals a discrete level.
- CT has no quantization error (DSP limitation).
- CT is free of frequency aliasing (DSP limitation).
- Continuous-time systems do not have control lag (DSP limitation).

CT-DSP ADCs "sample" the input signal's amplitude when the input signal exactly equals a discrete voltage level. The sample times are nonuniformly spaced, and there is no quantization error and no frequency aliasing. A bandlimited signal can be fully reconstructed from this set of samples, as long as the average sampling rate is at or above the Nyquist frequency ($f_s/2$) for that bandwidth.[33] Unfortunately, this requires considerable computational effort. Instead, CT-DSP signals are typically processed as the asynchronous zero-order-hold (ZOH) representation of the sample set. These signals, as described by Inose et al.,[34] are easily processed in real time. The disadvantage is that the ZOH representation creates unwanted harmonics, but unlike DSP signals, harmonics that fall out of band are not aliased and do not contribute to noise and distortion.

Continuous-time systems were first considered in the 1950s for control systems. Continuous-time systems do not have control lag (time delay) like discrete time or digital systems. In 2003, Tsividis published research showing that continuous-time systems have useful properties, such as no frequency aliasing and no quantization error.[35] In 2010, Kurchuk[36] published a dissertation covering continuous-time signal processing. More recently, there has been significant research efforts to develop asynchronous FPGA implementations.[4–7] A recent small business innovative research call for research proposals sought to develop continuous-time systems for signal processing applications.[37]

2.5 Comparison of Signal-Processing Strengths and Limitations

Table 1 compares the signal processing architectures listed in Fig. 1. Signal processing applications for continuous-time systems are shown in Table 2. Current integrated circuit (IC) fabrication is moving toward lower voltage and finer line widths. Figure 3 compares current IC fabrication trends for DSP and CT-DSP. This technical report focuses on CT analog-to-digital conversion and a CT software-reconfigurable radio architecture.

Table 1 Signal-processing comparison

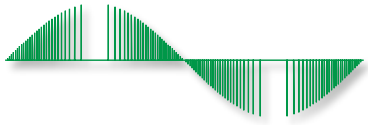
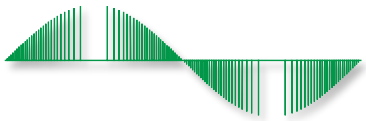
Technology	Advantages	Limitations
Analog 	• Mature technology • Low cost • Can be low power	• Limited to ~60 dB SNR (wideband NTSC television) • ~100 dB (narrowband)
Discrete Time 	• Mature technology • Low cost • Charged coupled devices	• Limited by aliased noise and • Control system lag
Digital Signal Processing 	• Mature technology • Low cost • Low complexity • Clocked systems	• Limited by frequency aliasing, quantization error, zero-order hold frequency response, and control system lag • Clock speed determines power
Continuous-Time Signal Processing 	• No quantization error • No frequency aliasing • No control system lag • Low power • Less sample points	• Research and development • Unclocked asynchronous design techniques required • ICs not commercially available • FPGA applications beginning for asynchronous systems

Table 2 Continuous-time signal processing

Technology	Promising application areas
Continuous-Time Signal Processing 	• Low-power medical applications • Low-power sensing applications • Control systems applications (no control system lag) • Analog-to-digital conversion • Software-reconfigurable radio applications

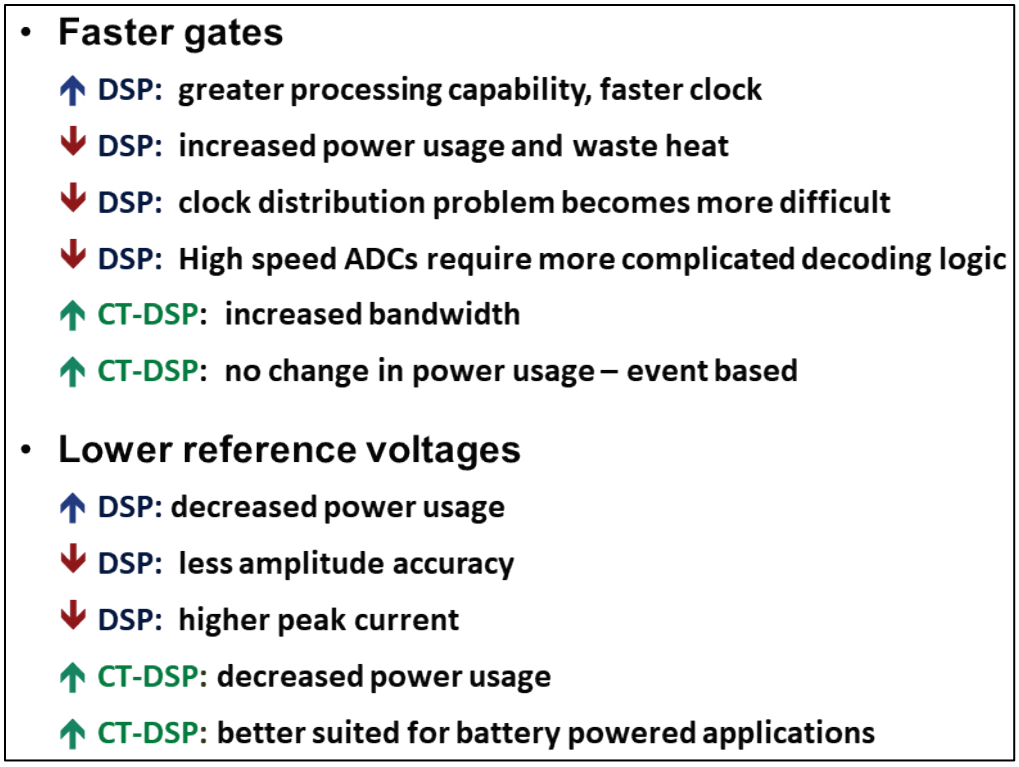


Fig. 3 Advances in IC fabrication: consequences to future signal processing

Figure 4 illustrates the advantages of continuous-time systems for electrocardiogram (ECG) signals.[38–40] The continuous-time level-crossing ADC only requires 225 samples compared to the 1250 samples for conventional DSP. Fewer samples result in lower energy. In Fig. 5, a frequency sweep illustrates how CT's "sample points" are proportional to the slope of the input signal. CT's "sampling frequency" self-adapts to save power. The CT signal has four sample points per half sine wave cycle. This is constant across the frequency sweep. The DT signal must have its sampling frequency computed for the highest frequency present in the frequency sweep. The discrete time signal's sampling frequency is $f_s > 2f_{bw}$, where f_{bw} is the bandwidth for the system. Due to the poor frequency response of a zero-order hold, practical systems typically use an oversampling ratio (OSR) of OSR = 8 to 16, so the sampling frequency is $f_s > 8 \cdot 2f_{bw} = 16f_{bw}$. The frequency sweep in Fig. 5 also illustrates the energy efficiency for continuous-time systems.

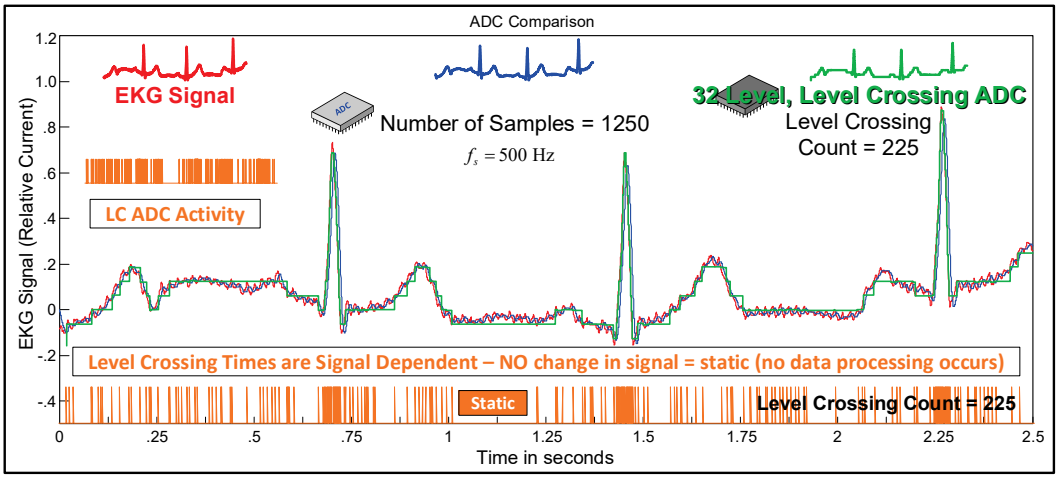


Fig. 4 Comparison of conventional DSP to CT for ECG signals. The 32-level, level-crossing ADC requires significantly fewer sample points compared to conventional Shannon sampling. This illustrates the low-power potential for nonuniform time step, level-crossing sampling.

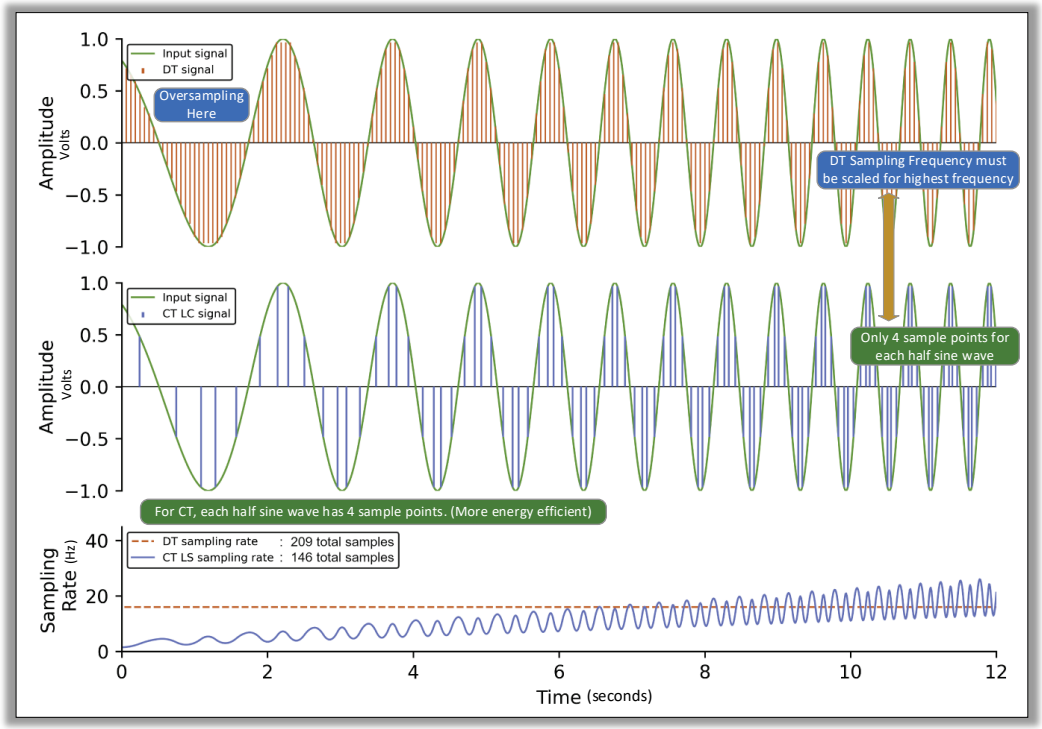


Fig. 5 Comparison of conventional DSP to CT for frequency chirp signal. For continuous time, the chirp signal has a constant number of sample points per half sine wave cycle. Frequency chirp will be a useful waveform for testing continuous-time systems.

2.6 Analog-to-Digital Conversion

Continuous-time level crossing and conventional flash ADCs have the same structure. A conventional flash ADC has a bank of comparators to simultaneously compare a discrete set of voltage levels to the input signal. A digital latch is used to sample the binary estimate of the input signal. As shown in Fig. 6, the CT flash ADC has the same structure as a conventional flash ADC. The surprising difference is the CT flash ADC is *simpler* than the conventional flash ADC; no clocked latch is required to sample the output value. Since the CT flash's sampling frequency is directly proportional to the input signal's slope, it is more energy efficient. The equivalence time is the time when the input signal exactly equals a discrete voltage level. Since the amplitude exactly equals the input signal at each equivalence time, there is no quantization error like conventional DSP.

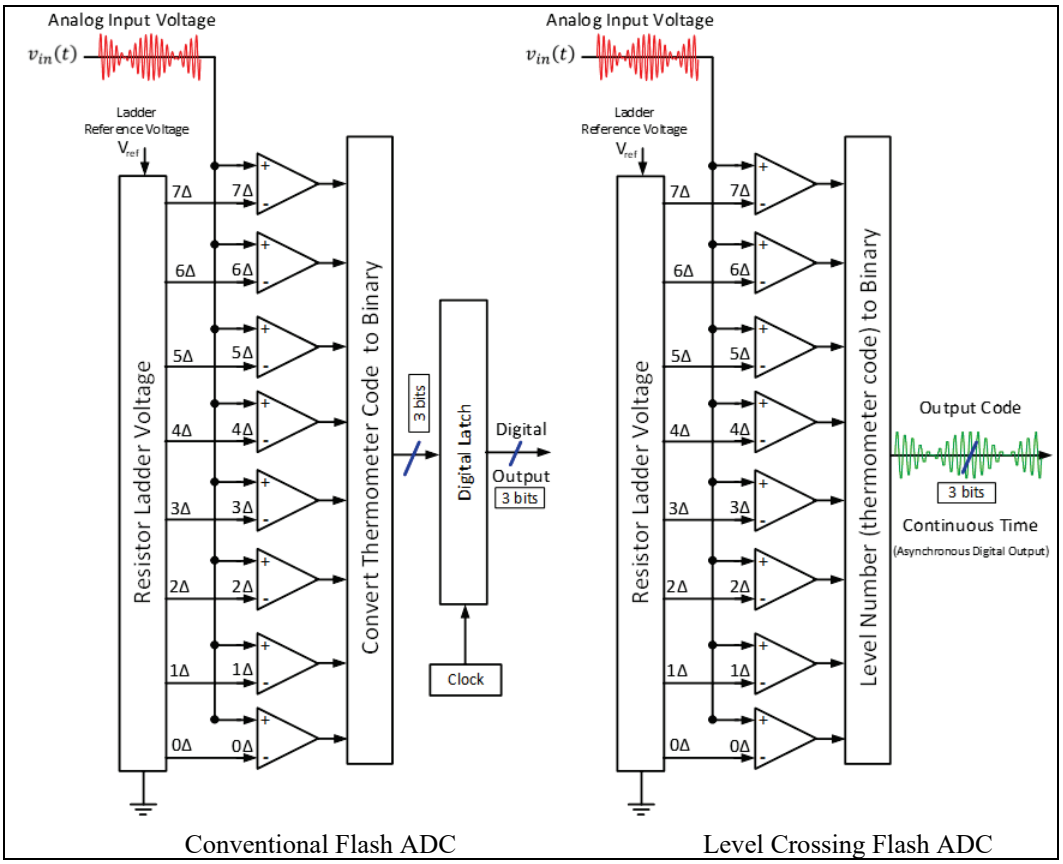


Fig. 6 Conventional flash ADC and CT level-crossing flash ADC. CT level-crossing flash ADC has the same structure as a conventional flash without the clocked latch. The CT flash also is more energy efficient.

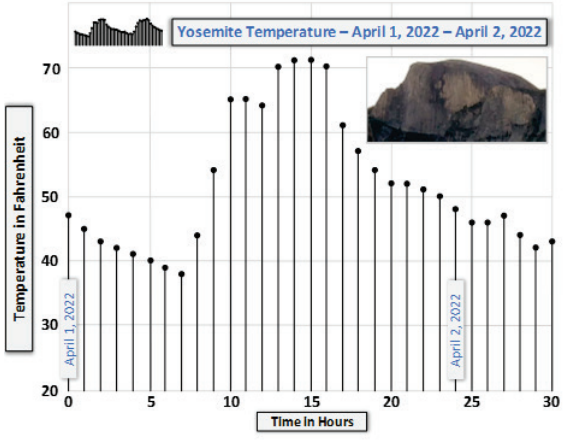
2.7 Software-Reconfigurable Radio (SRR) Architecture

A software-defined radio (SDR) replaces traditional analog components, mixers, filters, limiters, and others with software implementations. For an SDR architecture, the main advantage is the real-time reconfigurability (reprogrammable). A software-based digital filter consists of filter coefficients, additions, multiplications, and delay elements. For example, a digital filter can be changed by simply changing the stored filter coefficients. Rohde[41] published the first paper on SDRs in February 1984. Johnson[42] published the first open publication about SDRs in May 1985.

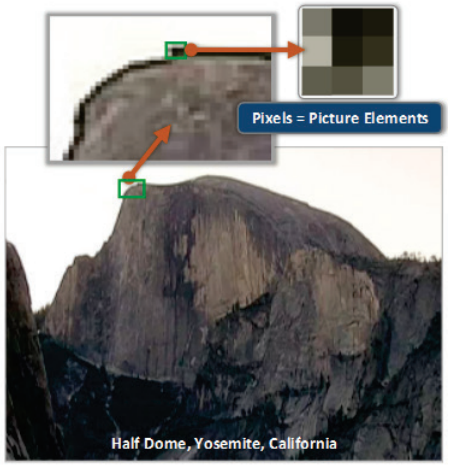
As technology has improved, the interface between analog and digital has moved closer to the antenna. Current SDRs are limited by the structure of the SDR and require high sampling frequencies, which means high power. A second limitation is that the analog front end causes high intermodulation distortion. In this report, a new SRR using a CT-pipeline ADC is presented. The new architecture overcomes limitations present in current SDRs.

3. Discrete-Time Systems and Digital Systems

Figure 7 illustrates two example discrete systems. Figure 7a is a temperature graph in hours for Yosemite, California, 2022 April 1–2.[43] Figure 7b is the Half Dome at Yosemite National Park.[44] The temperature graph and image illustrate two features of discrete systems. Discrete-time systems can only move forward in time. An image is a 2D discrete system. A pixel position is located using both positive and negative coordinates.



(a) Discrete Time – 1D



(b) Discrete Time – 2D

Fig. 7 Discrete systems: (a) 1D (time) (data courtesy of California Data Exchange Center, Department of Water Resources); (b) 2D (space) (image courtesy of the National Park Service)

Modern thermometers are accurate to ±0.2 °F (±0.1 °C). The lowest temperature[45] ever recorded is –128.6 °F on July 21, 1983, at Vostok, Antarctica (77° 32' S, 106° 40' E, 11,220 ft [3420 m] elevation). The highest temperature[46] recorded is 134° F on July 7, 1913, at Furnace Creek Ranch, California (36° 27' N, 116° 51' W, –179 ft [–54.6 m] elevation). Temperature range is –128.6 °F through 134 °F, which equals 263 °F. For a temperature resolution of 0.1 °F, there are about 2700 temperature steps (11.4 bits). Compare this to the resolution for a compact disc with 65,536 steps (16 bits).[10,11] For a temperature rate of change in °F/min, Great Falls, Montana, on January 11, 1980, recorded a 47 °F temperature rise in 7 min.[47] The approximate bandwidth for this rise time of 47 °F/ 7 min is about 3 Hz (Eq. 1).

$$t_{rise} = \frac{0.32}{bandwidth} \Rightarrow bw = \frac{0.32}{t_{rise}} = \frac{0.32}{47^\circ \text{F}/_{7\ min}} \cong 3\frac{^\circ\text{F}}{\text{sec}} \text{ (rise time for 2nd-order control system)} \quad (1)$$

In the everyday world, time scales are relatively long and data points do not have to be highly accurate. For everyday systems like temperature, time scales and rates of change are small; sampling errors can be "large" compared to a compact disc system and still not change a measured value. For engineering systems, time scales are short, and high accuracy is required. Errors in timing can drastically reduce system performance.

In engineering systems, Shannon sampling requires high accuracy. The next section discusses Shannon sampling.

3.1 Shannon Sampling

Shannon's sampling theorem[22] proves that a bandlimited input signal, $x_{in}(t)$, can be sampled and perfectly reconstructed. The sampling theorem is based on the model presented in Fig. 8. The sampling model assumes (1) sampling frequency, $f_s > 2f_{bw}$, where f_{bw} is the bandwidth of the input signal; (2) the sampling function is infinite in time; and (3) the reconstruction filter is an ideal low-pass filter. Unfortunately, an ideal low-pass filter cannot be built; an approximation called a zero-order hold is used. The infinite time extent assumption is a reasonable assumption for nearly all systems.

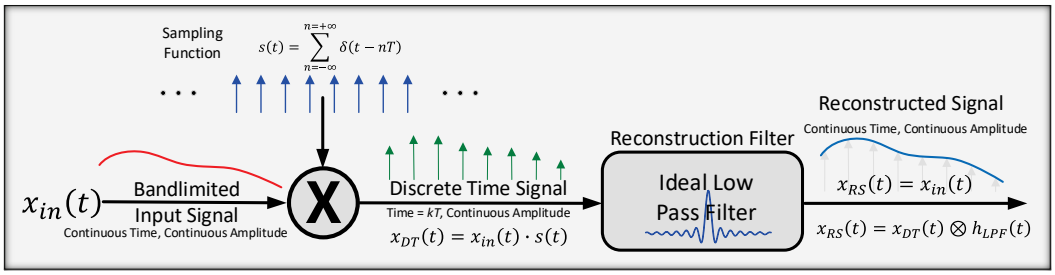
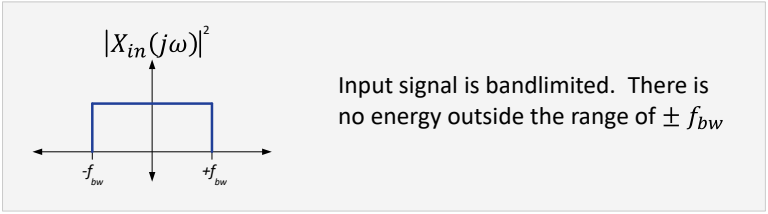


Fig. 8 Block diagram for Shannon's sampling and reconstruction theorem

The sampling function, $s(t)$, is a periodic array of delta functions, and it extends from time = -∞ to time = +∞. A band-limited signal has no signal energy outside the range of $\pm f_{bw}$ (here we have assumed a double-sided power spectral density). The input signal is required to be band-limited as shown in Eq. 2.



(2)

Shannon's sampling theorem can be graphically proven using a few Fourier transform identities. Figure 9 shows the Fourier transform pair for the sampling function. The time domain sampling function and frequency domain sampling function only differ by scaling constants; on the graph, they look the same. The radian sampling frequency is $\omega_s = \frac{2\pi}{T}$, where T is the time step between delta functions. The sampling frequency is $f_s = \frac{\omega_s}{2\pi} = \frac{1}{T}$.

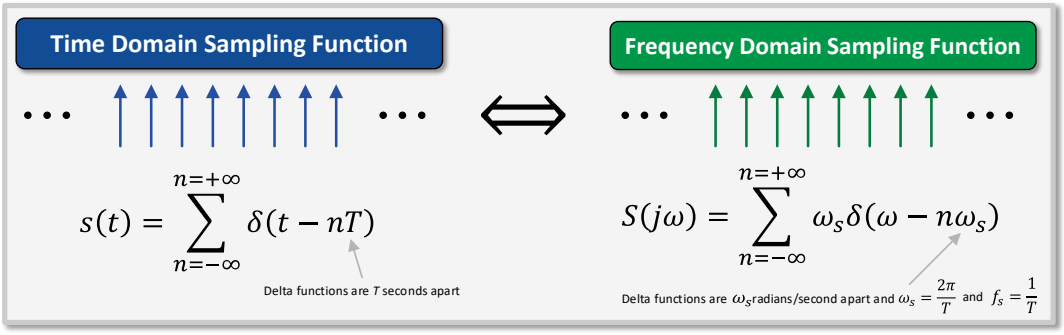


Fig. 9 Sampling function Fourier transform pair

As illustrated in Fig. 10, by grouping pairs of delta functions, the sampling function can be rewritten in terms of an infinite sum of cosine terms. Figure 8 shows a block diagram for Shannon sampling. The discrete-time signal (also called a sample data signal) is found in Eq. 3. The sampling function is composed of an infinite number of cosine terms with carrier frequencies of $n\omega_s$. The product results in an infinite number of amplitude-modulated signals. The ideal low-pass reconstruction filter removes the infinite number of amplitude-modulated signals to recover the input signal, $x_{in}(t)$. The infinite number of amplitude-modulated signals helps explain frequency aliasing.

$$x_{DT}(t) = x_{in}(t) \cdot s(t) = x_{in}(t) \sum_{n=-\infty}^{\infty} \delta(t - nT) = x_{in}(t)\left[1 + 2\sum_{n=1}^{\infty} cos(n\omega_s t)\right] \quad (3)$$

The sum is an infinite number of carrier frequencies

Carrier Frequencies

The product is an infinite number of amplitude modulated signals.

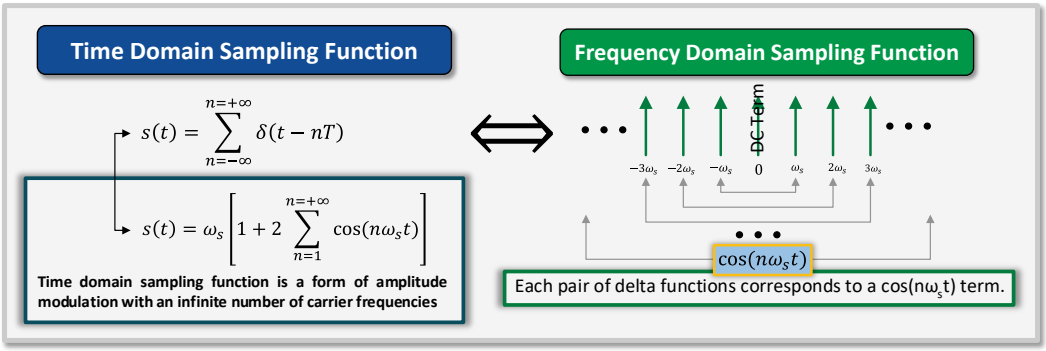
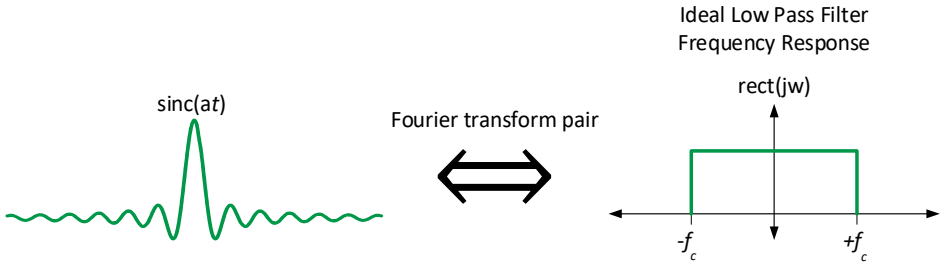


Fig. 10 Time domain sampling function is an infinite sum of cosine terms. By grouping pairs of delta functions in the frequency domain, an infinite sum of cosine terms is found. The infinite sum helps explain the origins of frequency aliasing.

Equation 4 gives the Fourier transform pair for the ideal low-pass filter. The sinc(a*t*) function is the impulse response for the ideal low-pass filter. The sinc(a*t*) exists before the time = 0. The sinc(a*t*) impulse response breaks cause and effect and the sinc(a*t*) filter cannot be built. An approximation to the sinc(a*t*) function called a zero-order hold is used. Figure 11 illustrates Shannon sampling in the time domain.



(4)

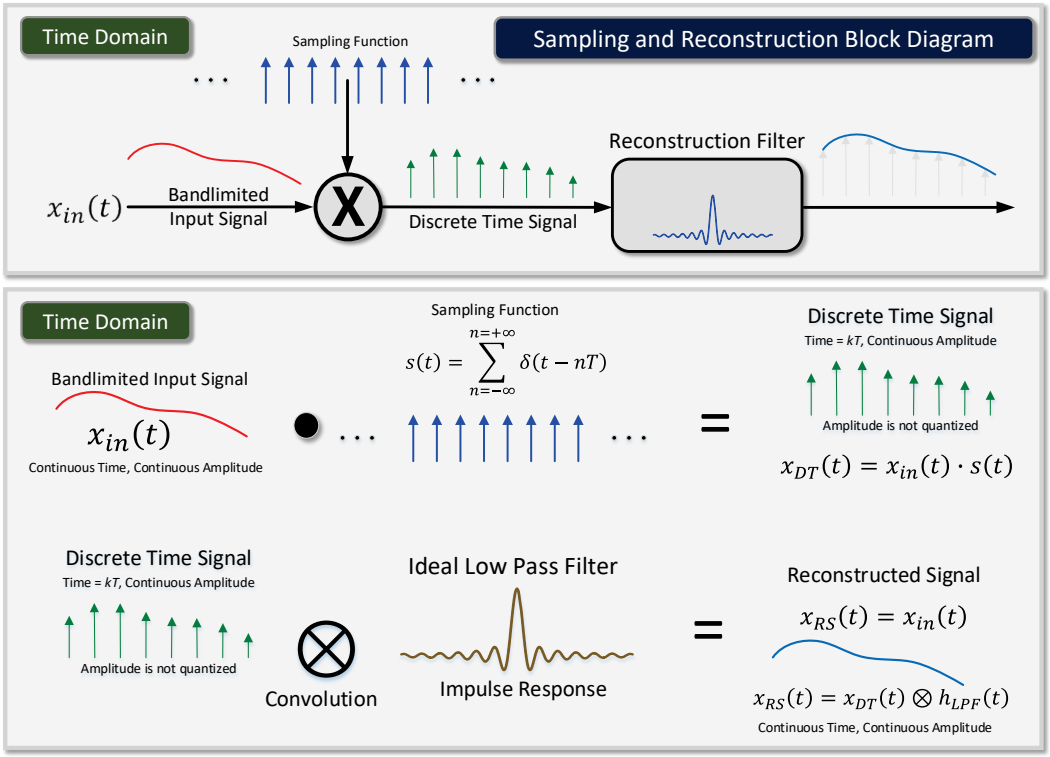


Fig. 11 Shannon sampling algorithm in the time domain. The input signal is sampled using an array of delta functions. The input signal is exactly recovered using an ideal low-pass filter with a sinc(a*t*) impulse response.

Figure 12 converts the Shannon sampling block diagram to the frequency domain. The input signal's frequency spectrum is sampled by convolving it with an array of delta functions. The discrete-time signal's frequency spectrum is the input signal's spectrum replicated across each delta function. By applying an ideal low-pass filter, the frequency spectrum of the input signal is exactly recovered. As described in Fig. 13, frequency aliasing occurs when the sampling frequency is too low and the input signals' spectrums overlap. Frequency aliasing cannot be removed. The AM signals for $n \geq 1$ cause frequency aliasing. If the input signal is a cosine, the product of $\cos(2\pi f_1 t) \cdot \cos(2\pi f_2 t)$ results in sum and difference frequency terms in $\cos[2\pi(f_1 \pm f_2)t]$. The sum and difference frequencies, $f_1 \pm f_2$, are what cause frequency aliasing.

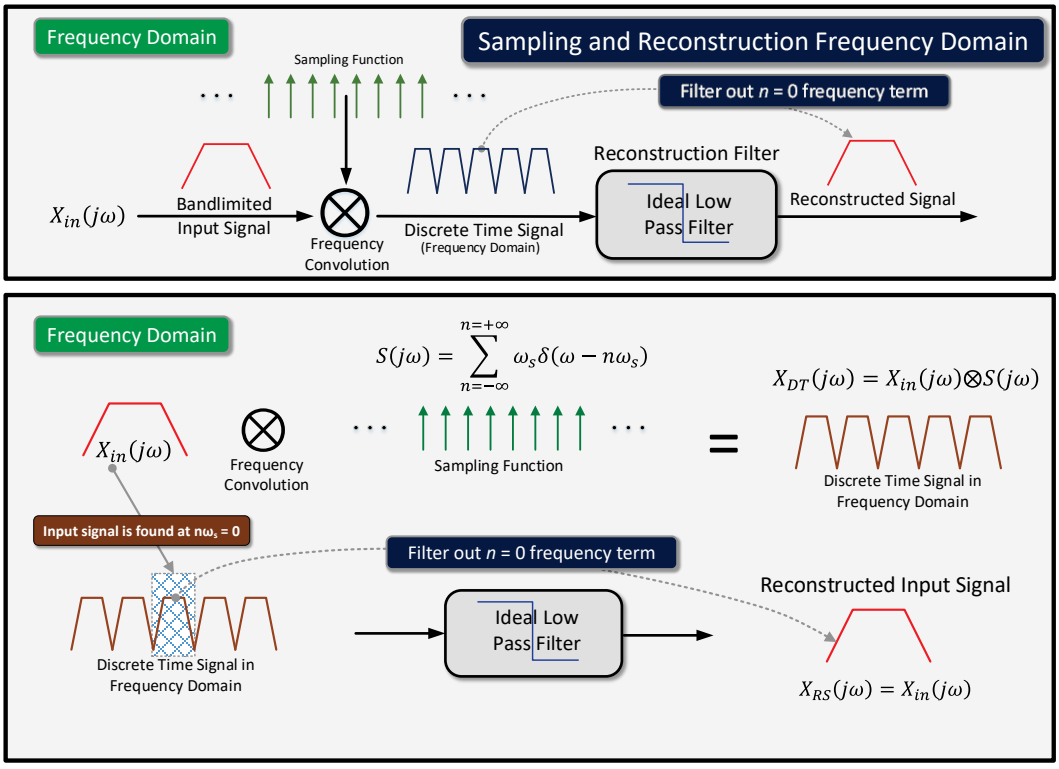


Fig. 12 Shannon sampling in the frequency domain. The input signal's frequency spectrum is sampled by convolving it with an array of delta functions. The discrete-time signal's frequency spectrum is the input signal's spectrum replicated across each delta function. By applying an ideal low-pass filter, the frequency spectrum of the input signal is exactly recovered.

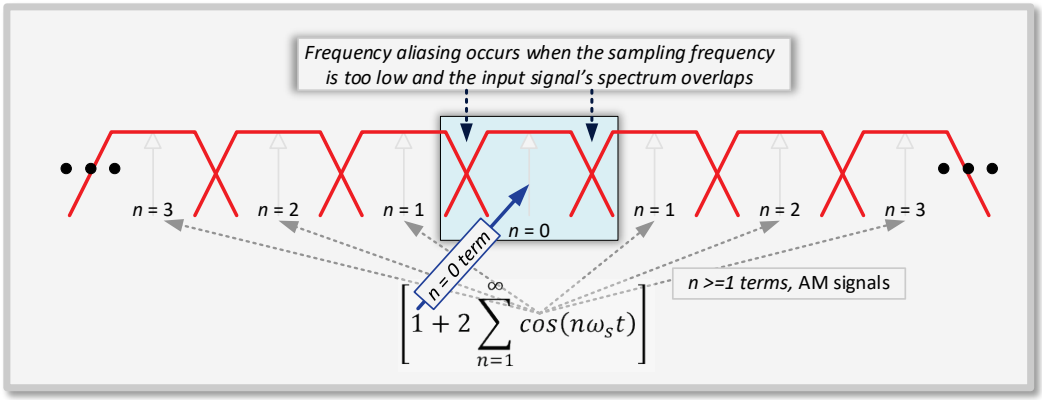


Fig. 13 Frequency aliasing occurs when the sampling frequency is too low and the input signals' spectrums overlap. Frequency aliasing cannot be removed. If the input signal is a cosine, the product of $\cos(2\pi f_1 t) \cdot \cos(2\pi f_2 t)$ results in $\cos[2\pi(f_1 \pm f_2)t]$. The sum and difference frequencies, $f_1 \pm f_2$, cause frequency aliasing.

Figure 14 shows an approximation for the sampling function with 50 terms and 100 terms. The infinite summation for the sampling function in Fig. 10 converges very slowly. The slow convergence causes limitations for real-world ADCs.

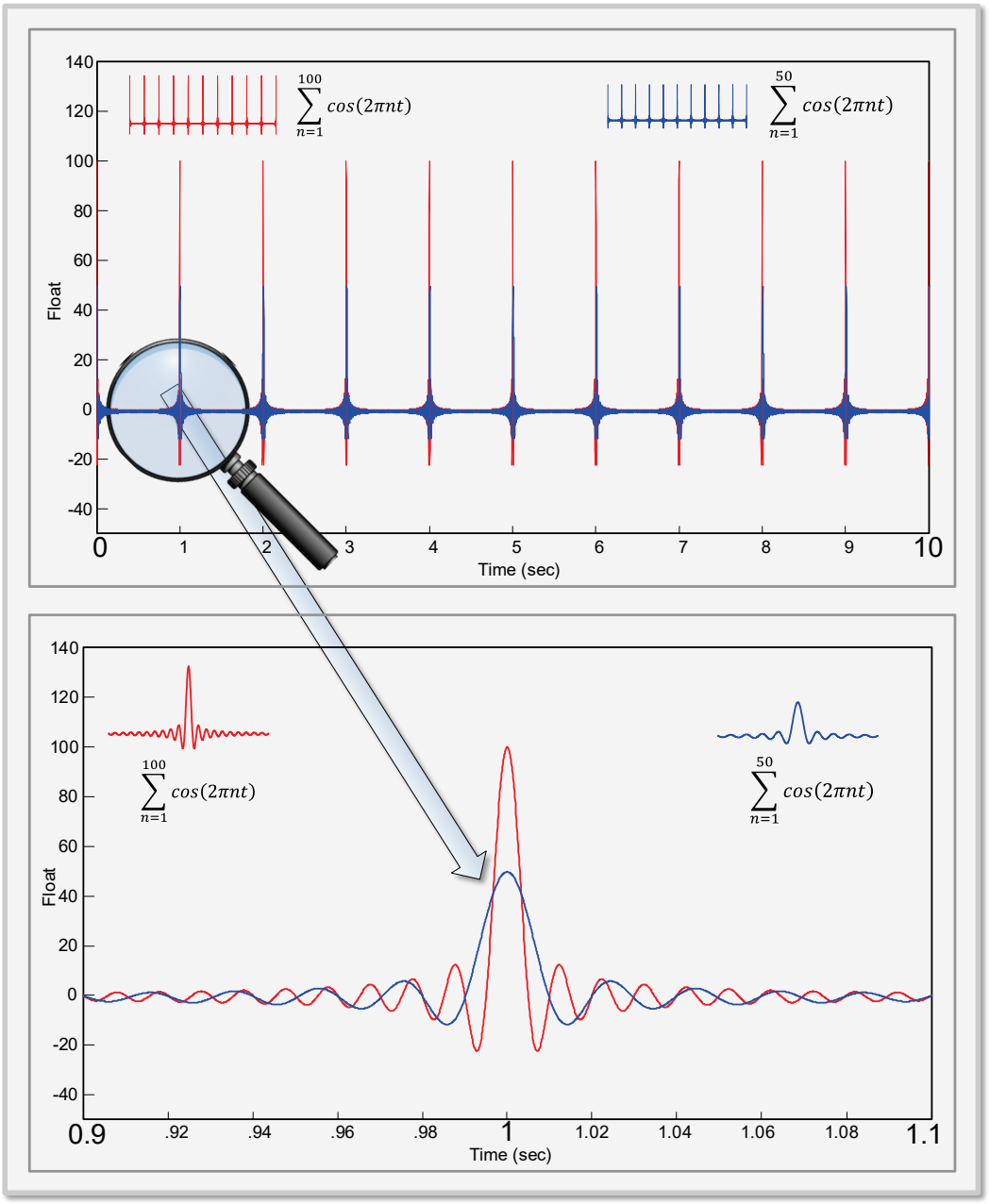


Fig. 14 Sampling function approximation. The sampling function has a slow convergence. For 50 terms, the peak amplitude is 50, and for 100 terms, it is 100. Each term adds 1 to the height. The sum converges very slowly. The slow convergence causes limitations for ADCs.

For a discrete-time system, Fig. 15 shows exact reconstruction of the input signal using weighted sinc(a*t*) functions. The weights are from the discrete time values as shown in the top of the figure. The middle graph shows 16 weighted sinc(a*t*)

functions, and the lower graph shows the reconstructed input signal. Since sinc(*at*) converges slowly, a very large number of terms is required for reconstruction. In the ideal case with an infinite number of terms, reconstruction is exact (see Fig. 8).

Ideal Shannon sampling is limited by a high sampling frequency. Shannon sampling does not consider the slope of the input signal. For regions with a low slope or high slope, the same number of samples are taken. This is not power efficient. As illustrated in Figs. 12 and 13, if the sampling frequency it too low, frequency aliasing occurs. Frequency aliasing cannot be removed after it has occurred. The ideal low-pass reconstruction filter cannot be built. An approximation to the ideal low-pass filter called a zero-order hold is used. The zero-order hold will be used to create a practical model for an ADC. The practical model is used to derive performance characteristics for an ADC.

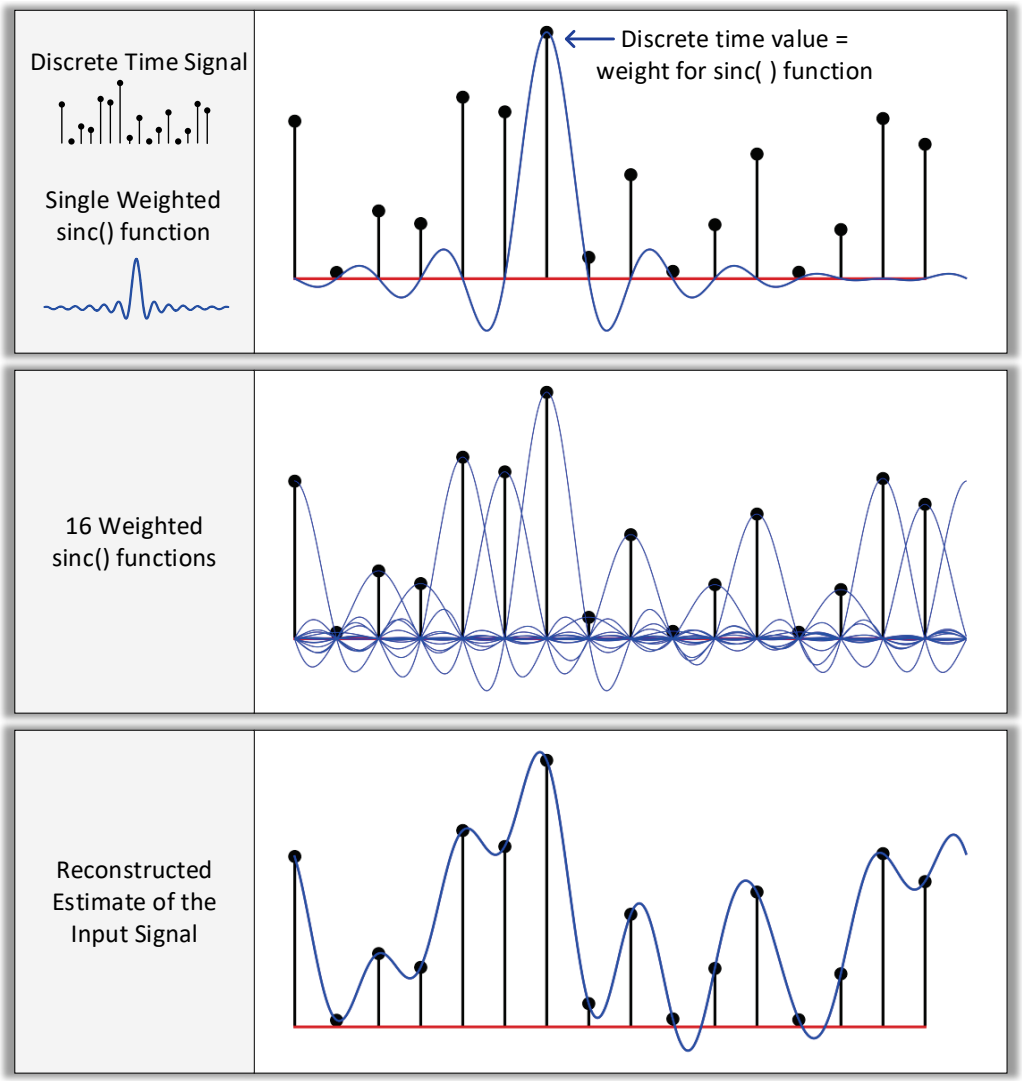


Fig. 15 Illustrates signal reconstruction using weighted sinc(*at*) functions placed at each sample value. The weights are equal to the discrete-time signal's amplitude (top graph).

3.2 Amplitude Quantization

Figure 16 shows the transfer function for a 3-bit quantizer with a 0.0- to 3.5-V input range mapped to the eight discrete output values of 0Δ through 7Δ. For an input voltage = 2.3 V, the quantizer outputs 5Δ, where Δ is the volts/step for the quantizer. Bennett's linear quantization error model calculates the "noise" power for quantization error.

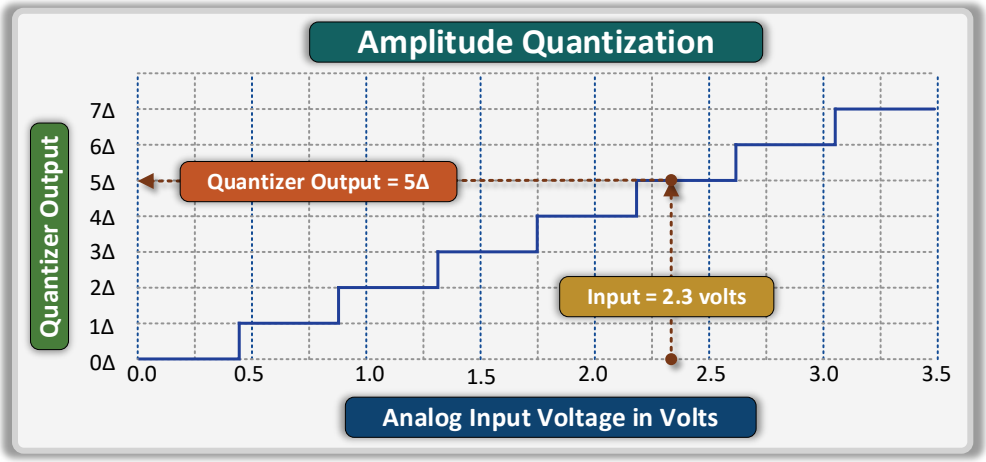


Fig. 16 Input voltage/output quantization transfer function

Bennett's quantization error model[29] is a linear model. The quantization error is modeled as a uniform random process called quantization "noise." The following list describes the conditions for Bennett's quantization error model[48]:

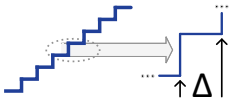
- Input signal does not exceed the input range of the quantizer.
- There are many quantization steps.
- The quantization step, Δ, is small.
- The current input to the quantizer does not depend on previous values (e.g., no feedback).

Bennett's quantization noise power equation is found in Eq. 5, where Δ is the quantization step. Since Bennett's quantization noise model has a uniform power spectral density, by oversampling, the noise power in a small bandwidth, f_{bw}, is reduced by 10log(OSR) dB. The OSR = f_s/f_r where f_s = the oversampled sampling frequency, f_r = sampling frequency required for a much smaller bandwidth and $f_r = 2f_{bw}$ where $f_s \gg f_r$.

$$\sigma_B^2 = \frac{\Delta^2}{12}\,\mathrm{W} \qquad (5)$$

Bennett's quantization noise power equation

Δ = quantization step in volts/step



3.3 ADC Model

Figure 17 illustrates a practical model for an ADC. The quantizer uses half-step offset quantization to convert the input voltage to a quantized value. Half-step offset quantization in Fig. 18 has a smaller quantization error than round-down quantization. Equation 5 is Bennett's quantization noise power equation. The signal power is calculated using a full-amplitude sine wave that completely fills the input voltage range as illustrated in Fig. 19. The ADC model in Fig. 17, Bennett's quantization noise model[29] in Eq. 5, and a full-amplitude sine wave (Fig. 19) are used to calculate the signal-to-quantization-noise ratio for an ADC. Equation 6 gives the SQNR equation for an n bit quantizer with oversampling. Equation 6 is the fundamental SNR metric for analog-to-digital and digital-to-analog converters (DACs), where n is the number of converter bits and OSR is the oversampling ratio. A compact disc has SQNR $\approx 6 \cdot 16 = 96$ dB for a full-amplitude sine wave. If the input signal is 0.1·full scale, the SNR is 76 dB (equivalent to 12.3-bit resolution), not 96 dB (equivalent to 16-bit resolution).

$$SQNR(n, OSR) = 6.02n + 1.76 + 10log(OSR)\ \text{dB} \quad \textbf{(Fundamental ADC and DAC Equation)}$$

$$OSR = \frac{f_s}{f_r} = \frac{f_s}{2f_{bw}} \qquad (6)$$

where n = number of quantizer bits, $OSR \geq 1$, and f_{bw} is the bandwidth of the low bandwidth signal.

ADCs are named for the conversion algorithm. Common ADCs include dual slope, flash, pipeline flash, successive approximation register, and delta sigma (also called sigma delta). Figure 6 illustrates a flash ADC.

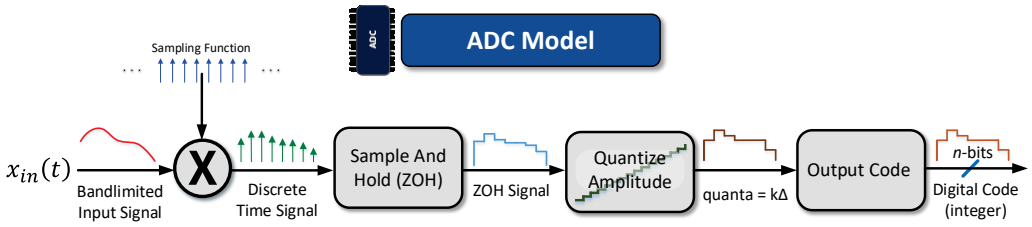


Fig. 17 Conventional ADC model. Shannon's sampling theorem requires the input signal to be bandlimited and the sampling frequency to be $f_s > 2f_{bw}$. A zero-order hold provides a constant value for 1 time step. The quantizer converts the ZOH signal to a quantized value. The coder stage assigns a digital code to the quantized value.

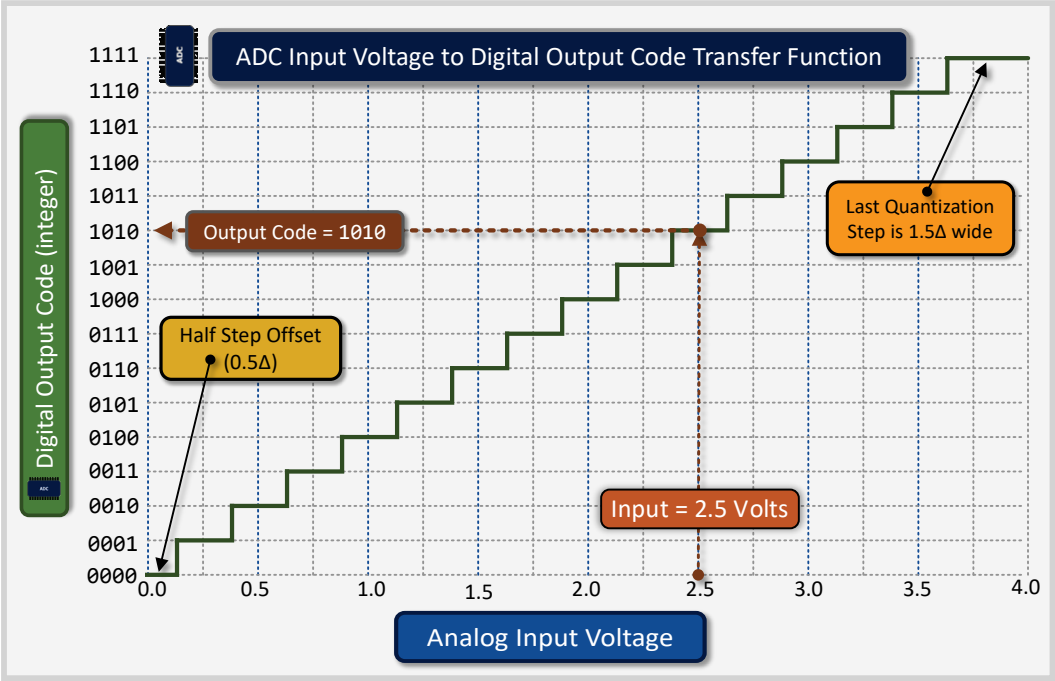


Fig. 18 Half-step offset quantization. For round-down quantization, the quantization error is bounded by $0 \leq$ quantization error $\leq \Delta$. For half-step offset quantization, the quantization error is bounded by $\frac{-1}{2}\Delta \leq$ quantization error $\leq \frac{+1}{2}\Delta$. Half-step offset quantization has half the maximum quantization error.

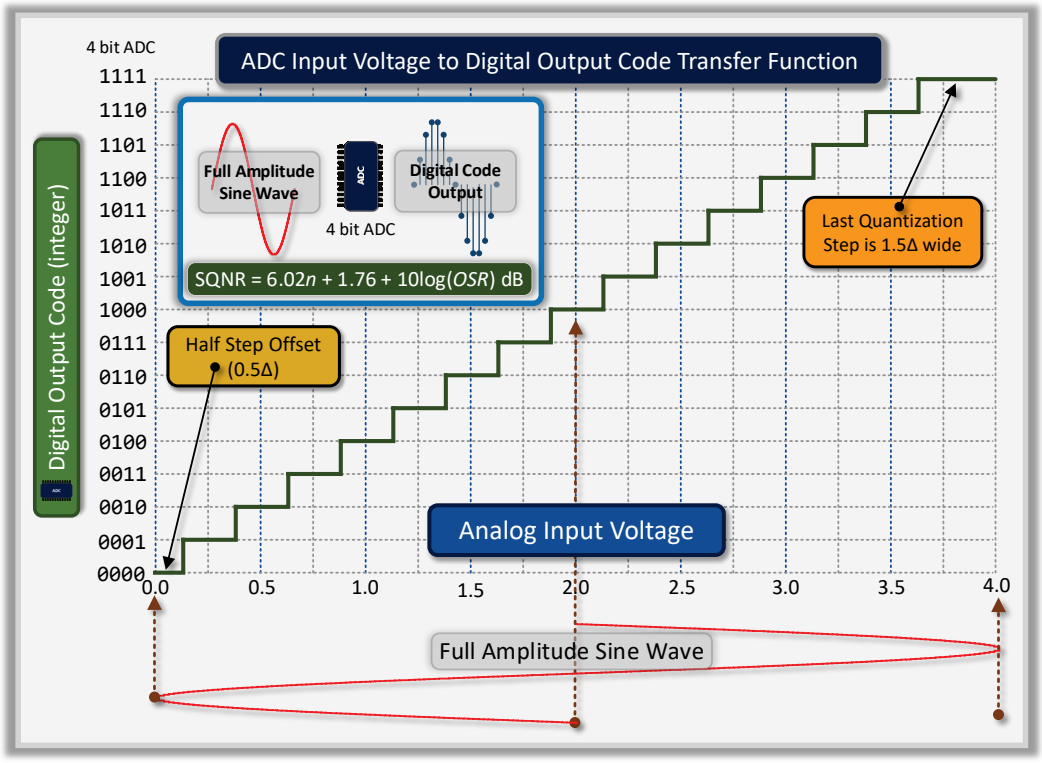


Fig. 19 Full-amplitude input sine wave. A sine wave that completely covers the input voltage range is used to calculate the signal-to-quantization noise ratio for an ADC.

4. Continuous-Time Signal Processing

Continuous-time systems (exact quantized amplitude, and time is continuous) offer several advantages over Shannon sampling:

- Free of frequency aliasing.[14]
- SWaP-C up to 10 times power savings in some cases.[1]
- Greater than 100 dB[9] SNR has been demonstrated.[2]
- No control system lag[2] (no time delay).
- Computation cost is proportional to input signal's instantaneous slope (signal activity).
- Since CT is not band limited, it can potentially detect super-oscillations (localized fast transients that can occur in amplitudes of widely different scales[49]) that cannot be detected by Shannon sampling methods.

With the development of asynchronous designs for FPGAs,[4–7] applications for CT-DSP are becoming practical for systems engineering.

Continuous time systems started back in the 1950s as discrete analog voltage levels without control system lag. Discrete-time systems are clocked and delayed by 1-time step (control system lag). Inose et al.[34] developed the asynchronous delta sigma ADC back in 1966. Lazar and Tóth[50] further developed the asynchronous delta sigma converter by developing a much more accurate delta sigma demodulation technique in 2004. Kurchuk et al.[51] developed a simple noise model for continuous-time systems in 2010. A gigahertz-speed continuous-time ADC[51] was developed in 2012. This technical report focuses on a pipeline continuous-time ADC and a continuous-time software-reconfigurable radio architecture.

Figure 20 illustrates a topographic map for the North and South Forks of the Kings River.[52] The contour lines are the 2D equivalent of a continuous-time signal. As pointed out in Section 3, the discrete voltage levels must be very accurate compared to the elevation contour lines in Fig. 20.

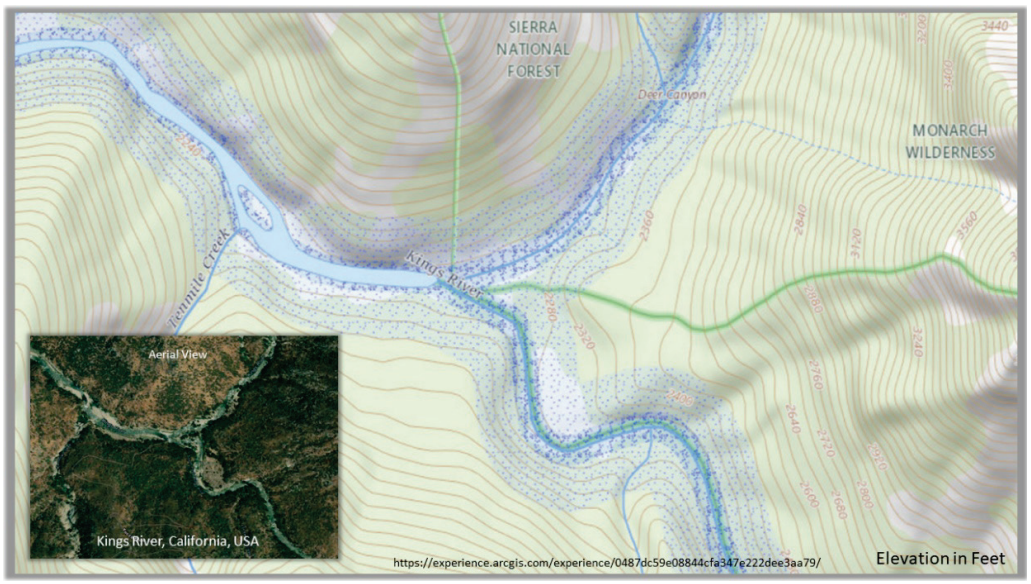


Fig. 20 North Fork and South Fork Kings River topographic map. Topographic map is a 2D equivalent of a continuous-time signal.

Figures 21 and 22 illustrate the advantages of continuous time for ECG signals. An ECG signal has a slowly varying region and a rapid pulse region. Continuous time only captures a few samples in the slowly varying region and a large number of samples for the pulse region. Marisa et al.[53] has demonstrated a low-power, 0.6-μW, continuous-time ADC equivalent to a conventional 8-bit ADC. For pulse signals like ECG, continuous time is signal activity dependent; no samples are generated in regions where the input signal is static. For ECG data, continuous-time signal processing only requires less than 20% of the samples required for conventional Shannon sampling. For battery-powered medical equipment, continuous-time systems are showing significant advantages.[51,53–55]

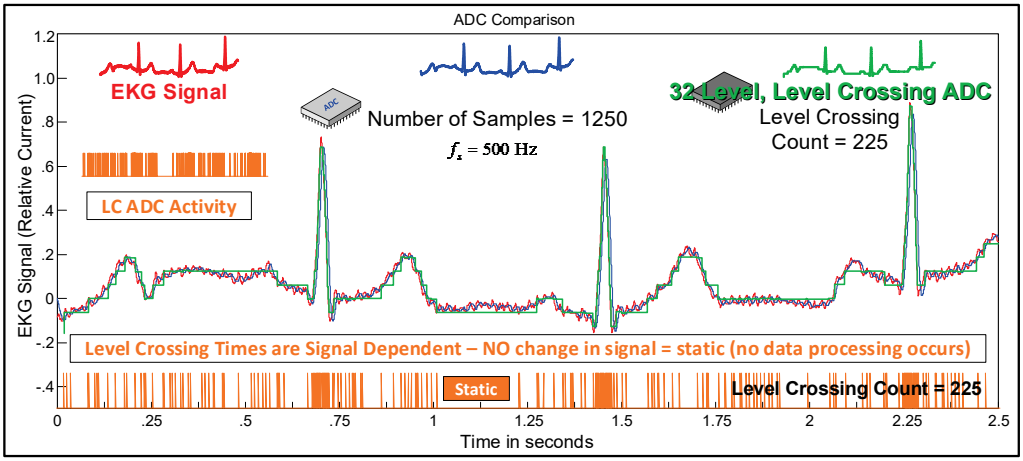


Fig. 21 Conventional ADC and level-crossing ADC for ECG data. For a conventional ADC, 1250 samples are required. For a 32-level, level-crossing ADC, only 225 samples are required.

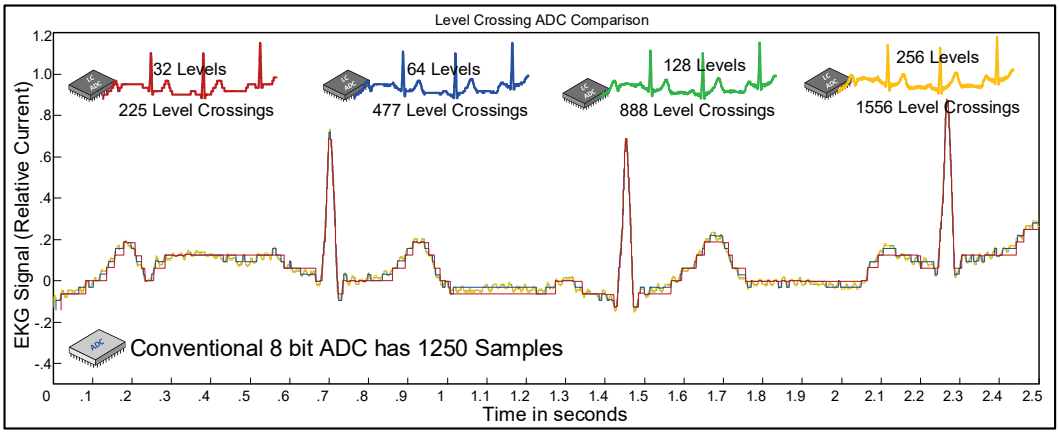


Fig. 22 Conventional 8-bit ADC has 1250 samples, f_s = 500 Hz. Tsividis[2] shows a level-crossing ADC has about 20 dB better SNR compared to an "equivalent" conventional ADC. To achieve an extra 20-dB SNR, a conventional ADC requires an oversample factor of *OSR* = 100 to achieve a 10log(100) = 20-dB SNR improvement. The total number of samples is 1250·100 = 125,000 which is much greater than 1556 for a level-crossing ADC.

4.1 Level-Crossing ADC and Asynchronous Zero-Order Hold Reconstruction

A 16-level, level-crossing ADC is simulated in Fig. 23. The equivalence times, t_k, are the times when the input signal exactly equals a discrete voltage level (gray dashed lines). The equivalence times, t_k, are not a fixed distance apart (nonuniform sampling). Shannon sampling is a form of amplitude modulation where the sum and difference frequency terms cause frequency aliasing. For continuous-time level-crossing "sampling", there is no modulation process, and it is free of frequency aliasing (DSP limitation). The difficulty with continuous-time systems is the nonuniform time steps. Similar to conventional DSP, an asynchronous zero-order hold is used to hold the output of the level-crossing converter constant until the next level crossing occurs.

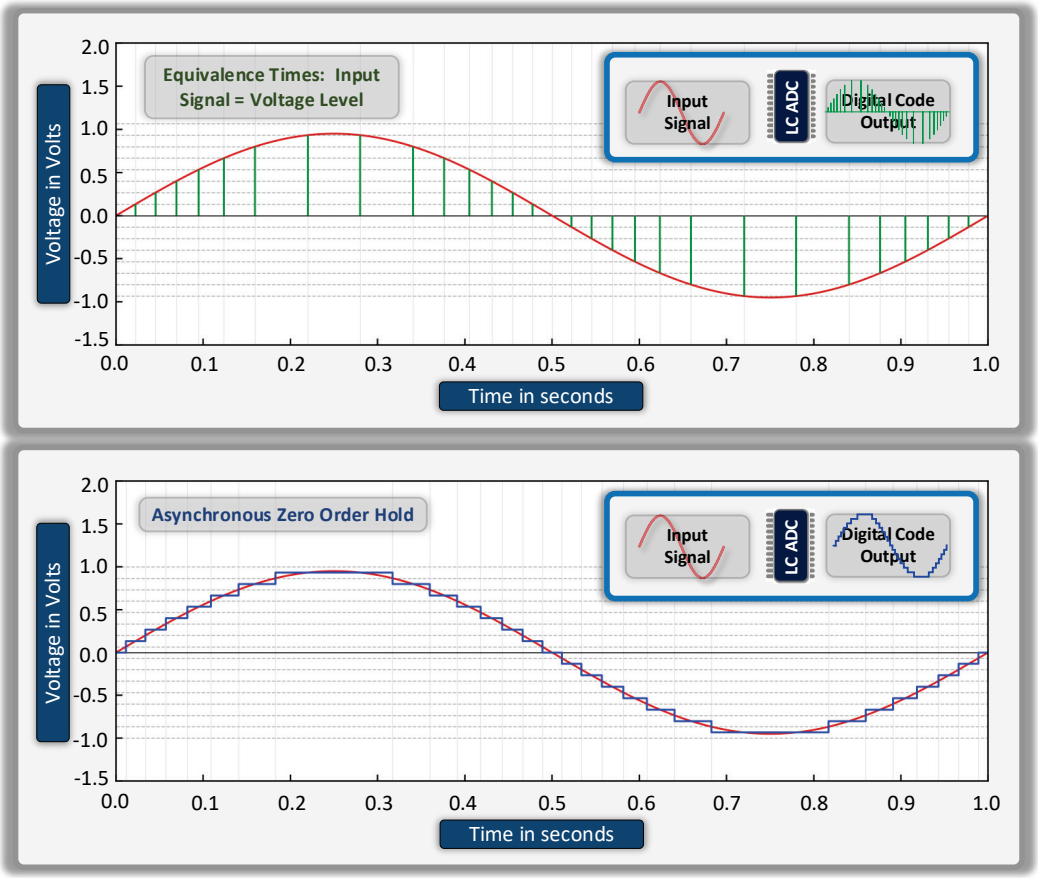


Fig. 23 Level-crossing analog-to-digital conversion. A level-crossing ADC creates a pulse when the input signal exactly equals a discrete voltage level. The equivalence times, t_k, occur when the input signal exactly equals a voltage level. The pulse amplitude is equal to the voltage level crossed. An asynchronous (nonuniform) zero-order hold will keep the output value constant until the next level crossing occurs.

Level-crossing analog-to-digital conversion contains more information than amplitude samples.[15] Figure 24 shows the cumulative level crossings as a function of time are equal to the integral of the input signal's slope. This illustrates that the level-crossing time differences, $t_k - t_{k-1}$, are inversely proportional to the slope of the input signal, and the effective (instantaneous) sampling frequency is proportional to the input signal's slope. This extra information can also be used for input signal reconstruction (also called interpolation). In practical terms, a large oversampling factor is required to calculate the quantized time differences, $t_k - t_{k-1}$.

In Section 5, a pipeline version of a continuous-time level-crossing ADC is presented along with a software-reconfigurable radio application in Section 6.

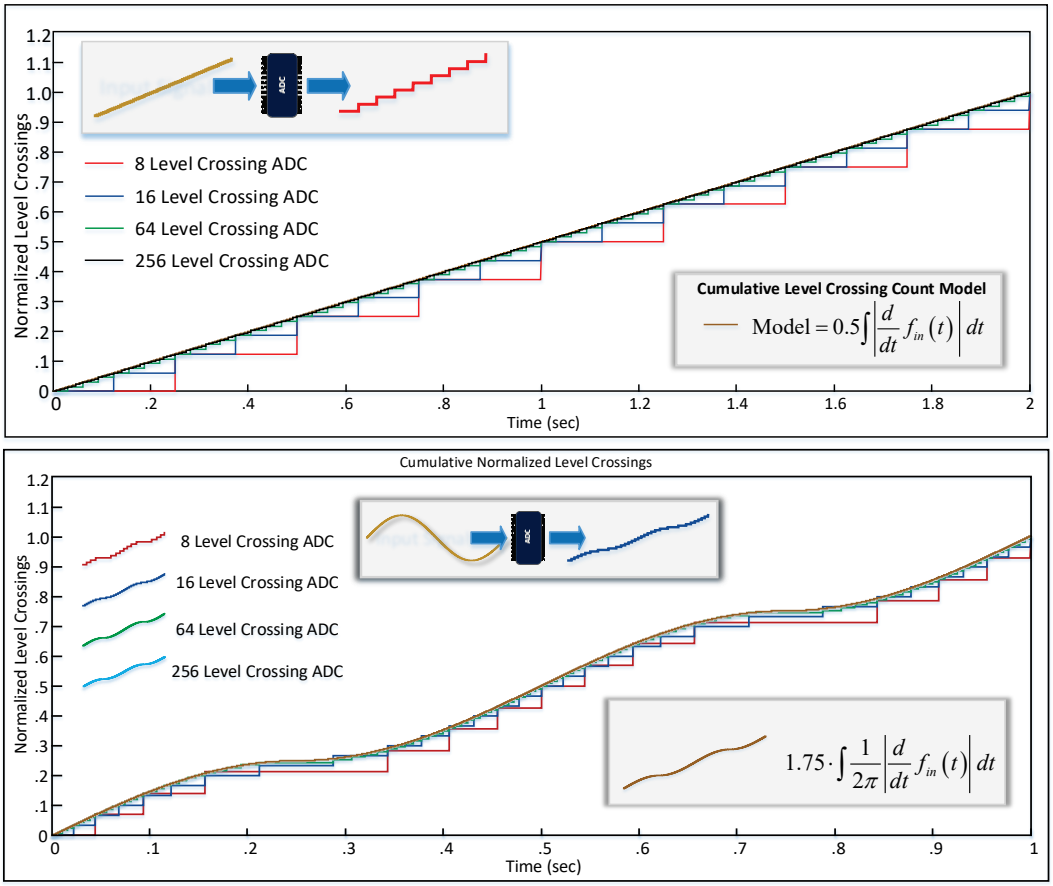


Fig. 24 For level-crossing analog-to-digital conversion, the level-crossing time differences, $t_k - t_{k-1}$, are inversely proportional to the slope of the input signal. The effective sampling frequency is proportional to the slope of the input signal. Cumulative level-crossing count is proportional to the integral of the input signal's slope. The top graph shows a ramp input signal, and the bottom graph shows a sine wave input signal. Level-crossing times are shown in Fig. 23.

4.2 Irregular-Step Level-Crossing ADCs

Figures 25 and 26 illustrate nonuniform voltage level steps. Figure 25 shows coarse voltage levels with fine levels at ±10% offset. The coarse and fine offset levels can provide better slope information. Figure 26 illustrates a level-crossing ADC with logarithmic voltage levels.

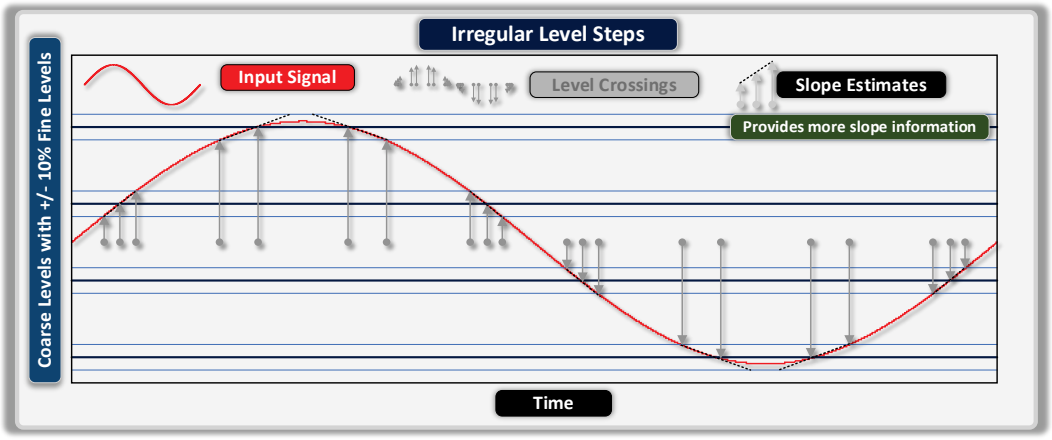


Fig. 25 Irregular level steps. By placing a fine level ±10% above and below a coarse level, the level-crossing ADC can measure more details about the input signal's slope.

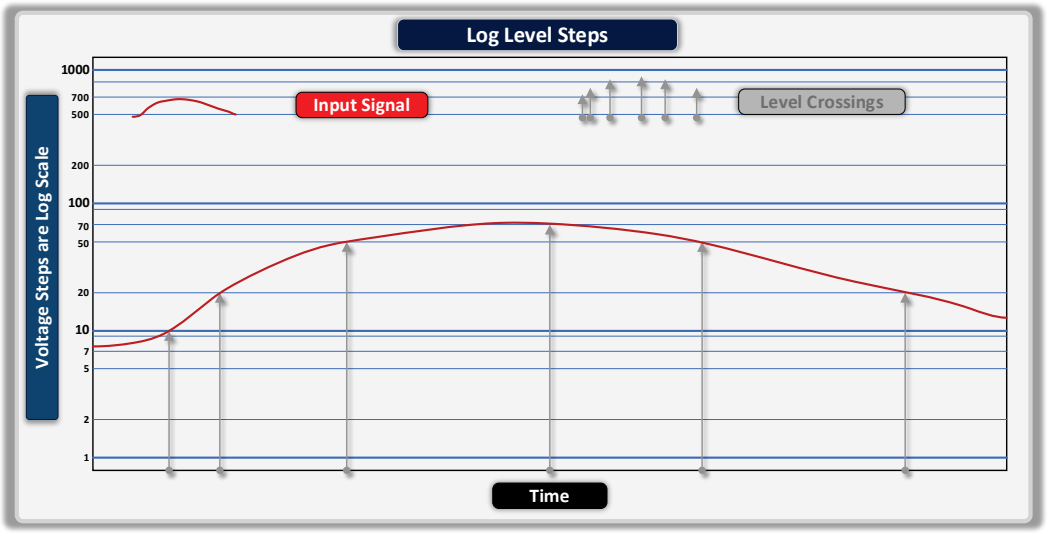


Fig. 26 Log level steps. Level-crossing ADC with log level steps.

4.3 Continuous-Time CCD System

It is possible to create a continuous-time CCD for analog signal processing and neural network applications. Figure 27 illustrates a CT CCD implementation for analog signal processing. A level-crossing ADC provides the "clock" signal to transfer charge across the array of CCD cells. The CCD cell outputs can then be processed by a traditional analog signal-processing system. Since Fig. 25 uses CT, not Shannon sampling, it is free of frequency aliasing and aliased noise.

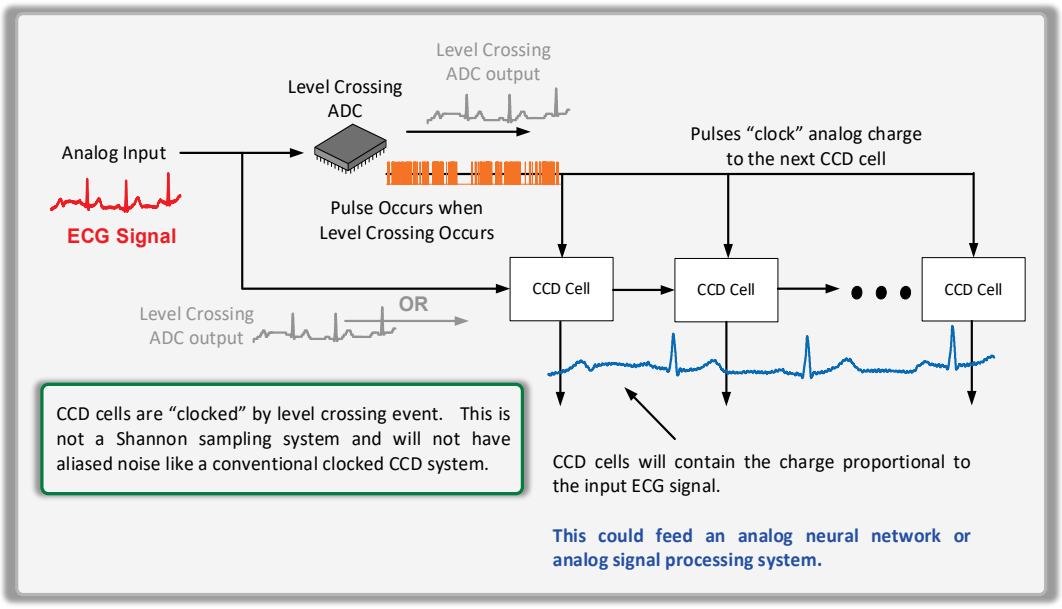


Fig. 27 Continuous-time analog CCD system. With a level-crossing ADC, an array of CCD cells can be clocked with the level-crossing times to create an analog CCD continuous-time system. The CCD cells' outputs can then be used in a fully analog signal-processing system.

The CCD invention described in Fig. 27 shows the benefits provided by continuous-time systems for signal- and image-processing systems limited by aliasing and aliased noise. The CT analog signal-processing concept described in Fig. 27 can also be applied to flash memory cells, memristors, and any other analog memory.

5. CT Pipeline Flash, Level-Crossing ADC Introduction

A conventional 2-bit flash ADC is illustrated in Fig. 28. A 2-bit flash requires three comparators. The input signal is simultaneously compared to all the discrete voltage levels from the resistor ladder. A conventional flash ADC has exponential complexity. Each bit of resolution doubles the number of comparators. An n-bit flash ADC requires $2^n - 1$ comparators. A 3-bit flash requires seven comparators. A 4-bit flash requires 15 comparators. A 10-bit flash requires 1023 comparators.

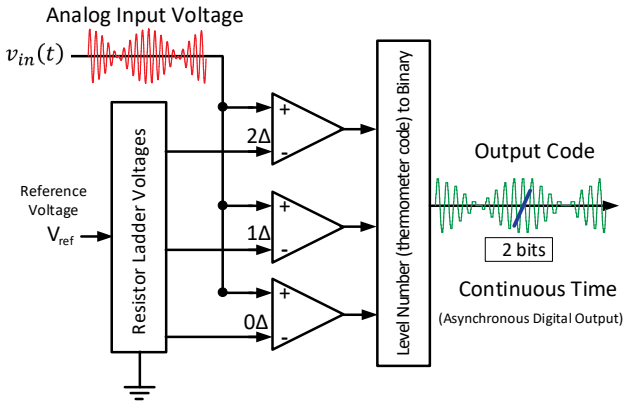


Fig. 28 Prior art 2-bit* continuous-time level-crossing ADC. A level-crossing ADC is similar to a conventional flash ADC without the clocked latch. An *n*-bit flash ADC requires $2^n - 1$ comparators and has exponential complexity. Each bit of resolution doubles the number of comparators. (*Costs more than 25 cents.)

Figure 29 shows a prior art conventional pipeline flash ADC. A pipeline flash ADC uses several pipeline stages where each stage includes a 2- to 4-bit flash ADC. The pipeline flash considerably reduces the complexity. A 12-bit pipeline flash only requires three 4-bit flash stages for a total of 45 comparators instead of 4095.

In this technical report, a new pipeline version of a continuous-time level-crossing ADC is presented along with a software-reconfigurable radio application. Prior art pipeline flash ADCs require complex amplitude error corrections, complex timing error corrections, and significant calibration to achieve high resolution.

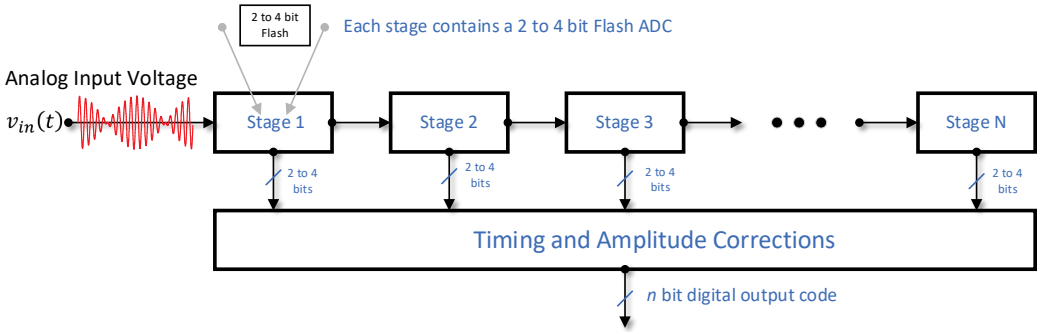


Fig. 29 Prior art conventional pipeline flash ADC. A pipeline flash ADC consists of *n*-stages, where the error voltage (stage input – digital output) is fed into the next stage and repeated.

5.1 CT Pipeline Flash, Level-Crossing ADC

Five key features of the present ADC invention are (1) non-overlapping pipeline-stage voltage levels; (2) design requires only a single resistor ladder network for all pipeline stages; (3) design guarantees that the input signal can be equal to only one voltage level at a time for all pipeline stages; (4) key design feature (3) greatly simplifies timing error correction; and (5) key design feature (3) greatly simplifies asynchronous finite-state machine controller and gray code implementation.

An unclocked (continuous-time) pipeline level-crossing ADC has several properties that provide for a better ADC. Only one level crossing may occur at a time *for all stages* in the pipeline. The amplitude level can change by only one level at a time, $Level(n+1) = Level(n) \pm 1$. As illustrated in Figs. 30 and 31, for the two-stage pipeline level-crossing ADC in Fig. 32, the coarse and fine levels never overlap. Figure 31 shows the input voltage to output digital code transfer function for Fig. 30. For a transition from a fine level to a coarse level, only one level crossing occurs. A block diagram for a two-stage pipeline level-crossing ADC is shown in Fig. 32. The resistor ladder provides six voltage levels. The coarse level crossing ADC (stage 1) uses all six voltage levels. The fine ADC (stage 2) only uses four of the six levels (the top level and bottom level are not used). By not using the top and bottom levels, the design guarantees that only one level crossing can occur at a time for all pipeline stages. In prior art conventional pipeline ADCs, 2- to 4-bits in each pipeline stage can simultaneously change. Multiple bits in the output digital code can change on every clock. Many simultaneously changing bits severely complicate amplitude and timing error corrections for prior art conventional pipeline ADCs. As illustrated in Figs. 25 and 26, nonuniform voltage level steps can be used. For the continuous time, n-stage pipeline level-crossing ADC, a more complicated circuit is required. Since the voltage steps are not a fixed distance apart, a variable gain amplifier is required.

The properties of the new unclocked pipeline level-crossing ADC overcome time delay and amplitude limitations present in current pipeline ADCs. The amplitude and timing corrections are much simpler for the new invention. Since only one level in a single pipeline stage can change at a time, no timing corrections may be required. Amplitude compensation for the stage gain blocks is required; however, gain calibration is simple compared to timing error compensation. Continuous-time SRR takes advantages of the performance provided by the continuous-time pipeline level-crossing ADC.

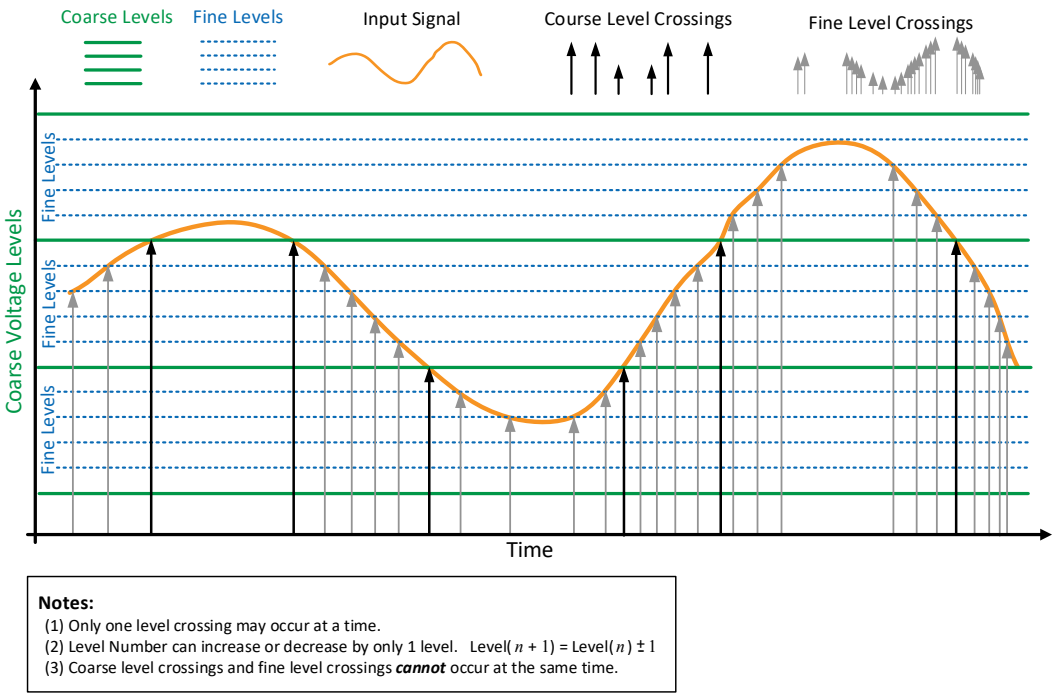


Fig. 30 The operation of a continuous-time pipeline level-crossing ADC. The first pipeline stage provides the coarse-level output. The second pipeline stage has its four fine levels inside the coarse voltage levels. This way there is no overlap between the coarse and fine levels. This ensures that only one level can be crossed at a time for all pipeline stages. The amplitude scaling for the coarse and fine voltage levels is a key feature of this invention. By only allowing for a single level to be crossed at a time, the present invention overcomes timing and amplitude accuracy errors present in conventional pipeline flash ADCs.

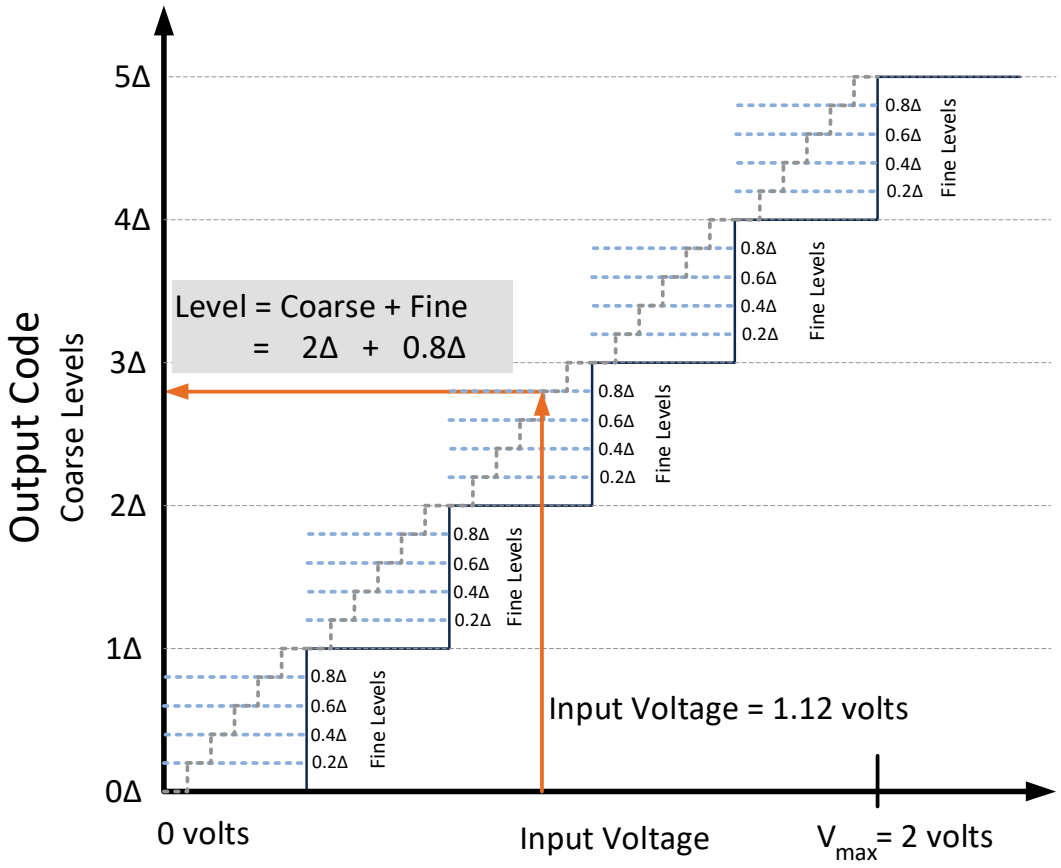


Fig. 31 An input voltage to output-level (digital code) transfer function. Coarse levels are located at 0Δ, 1Δ, 2Δ, 3Δ, 4Δ, and 5Δ. The error voltage between the input signal and coarse level value is fed into the fine level-crossing stage. The fine level-crossing voltage levels are 0.2Δ, 0.4Δ, 0.6Δ, and 0.8Δ. The fine voltage levels of 0.0Δ and 1.0Δ are not used. The fine voltage levels of 0.0Δ and 1.0Δ would overlap with the coarse voltage levels of 0Δ, 1Δ, 2Δ, 3Δ, 4Δ, and 5Δ. The pipeline design guarantees that only one level in any pipeline stage may be crossed at a time. An example input voltage of 1.12 V is mapped to coarse level = 2Δ and fine level = 0.8Δ for an output value = 2.8Δ.

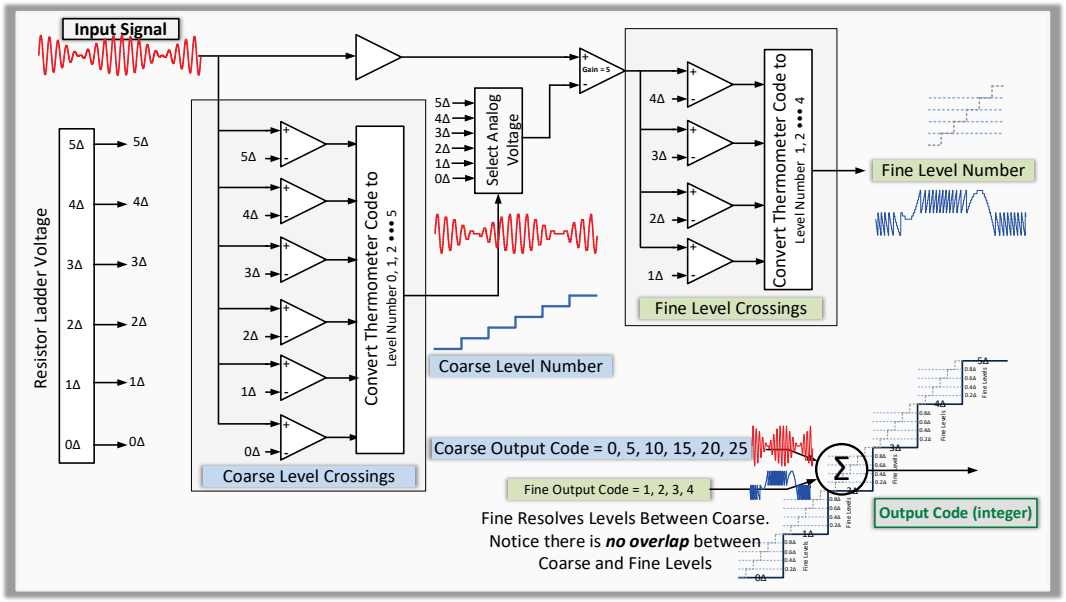


Fig. 32 A simplified block diagram for a continuous-time two-pipeline-stage level-crossing ADC. The single resistor ladder network is used for all analog voltage levels in all pipeline stages. The gain block for the error signal scales the stage error voltage to fit in between the coarse voltage levels from stage 1. The fine voltage levels 0.0Δ and 1.0Δ overlap with the coarse voltage levels and are not used. The resistor ladder, stage voltage levels, and gain scaling are key features of the present invention.

5.2 Three-Stage Pipeline Level-Crossing ADC Simulation

Figures 33–37 show simulated outputs for each stage in a continuous-time three-stage pipeline level-crossing ADC. Figure 37 compares the number of level crossings for a 512 level-crossing ADC to a three-stage pipeline level-crossing ADC. A 512-level, level-crossing ADC requires 512 comparators. A three-stage, pipeline level-crossing ADC requires only 26 comparators (stage 1 = 10, stage 2 = 8, and stage 3 = 8). Stage 1 is almost static compared to stage 3. Since each stage consists of a small number of voltage levels (8 to 10 voltage levels), the low-level activity for nearly static stage 1 helps save energy (Fig. 37).

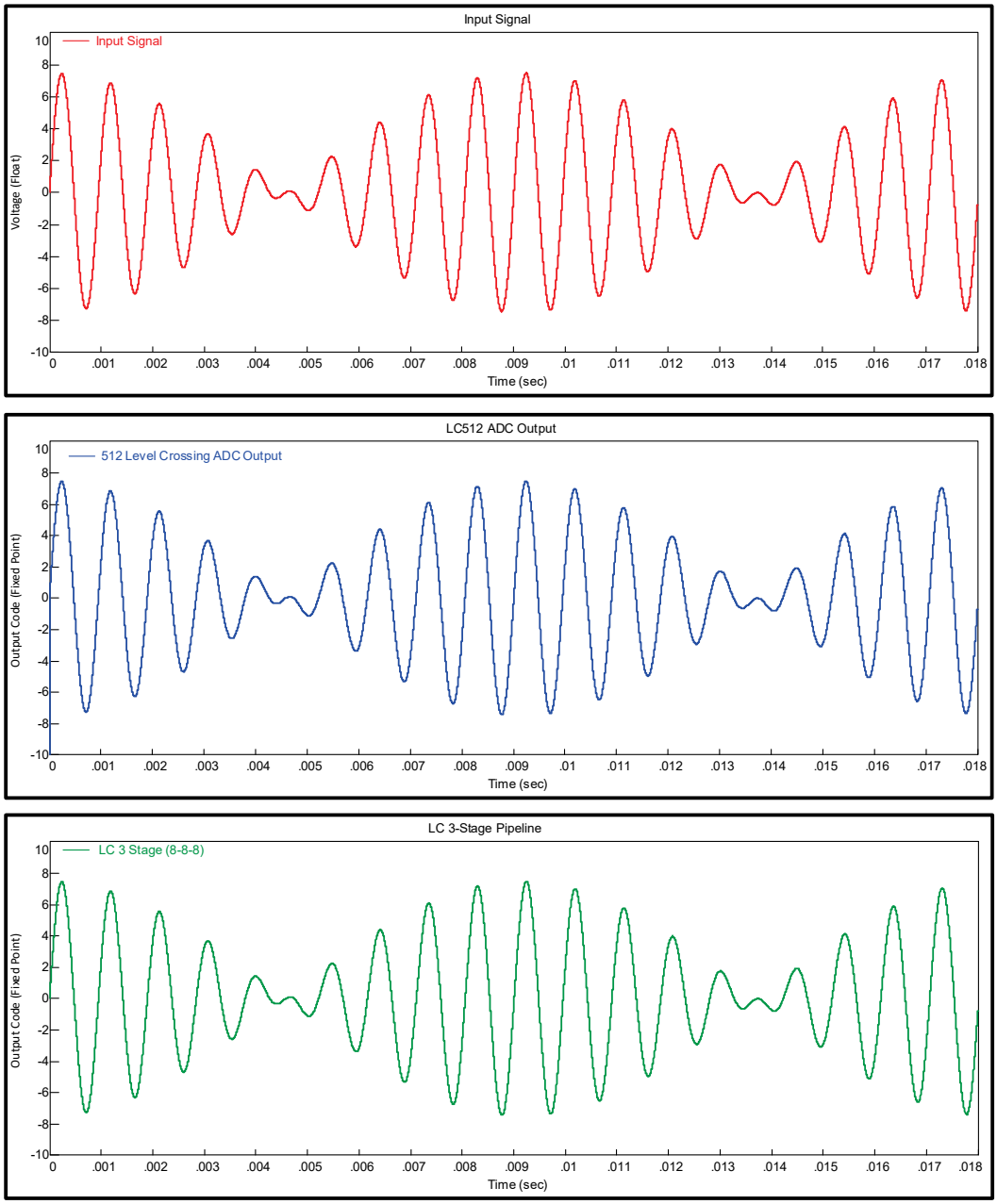


Fig. 33 Comparison of a simulated three-stage pipeline level-crossing ADC (requires 26 comparators and it is equivalent to a 512-level ADC) to a simulated 512-level, level-crossing ADC (requires 512 comparators). A two-tone input signal is used to compare the performance of the three-stage pipeline level-crossing ADC to a 512-level ADC.

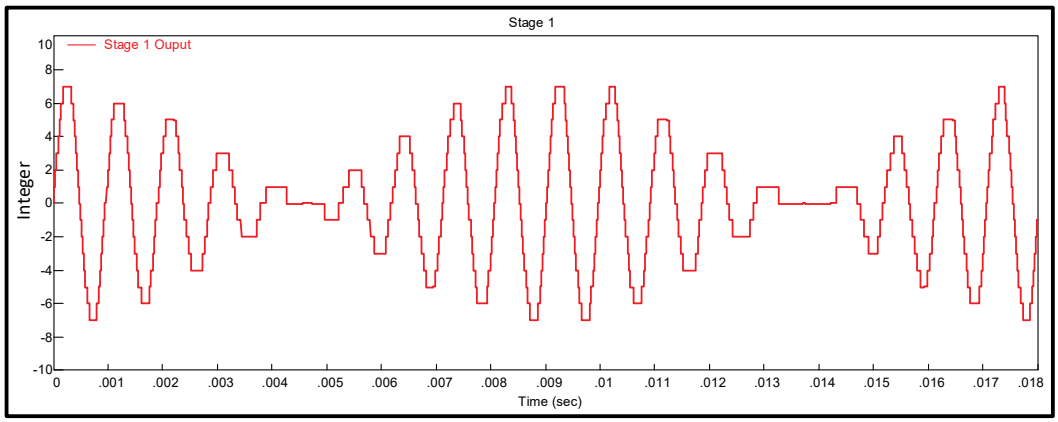


Fig. 34 Simulated pipeline stage 1 output for a three-stage pipeline level-crossing ADC

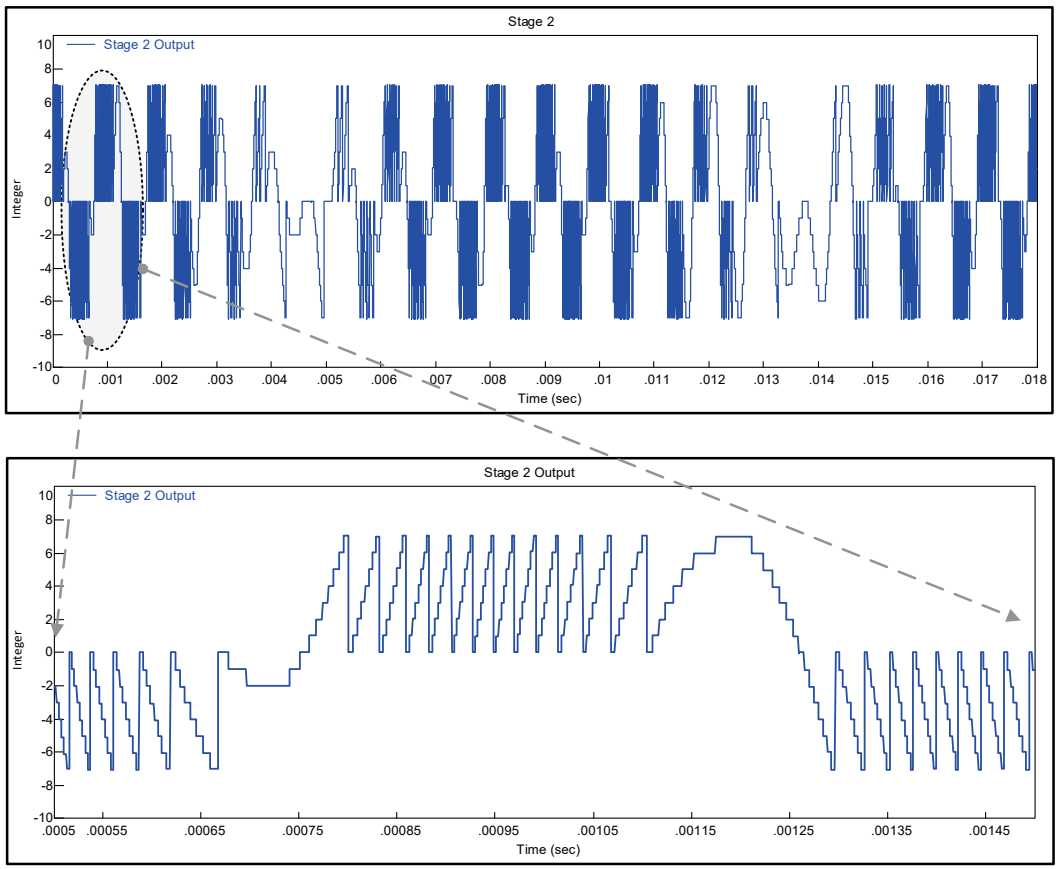


Fig. 35 Simulated pipeline stage 2 output for a three-stage pipeline level-crossing ADC

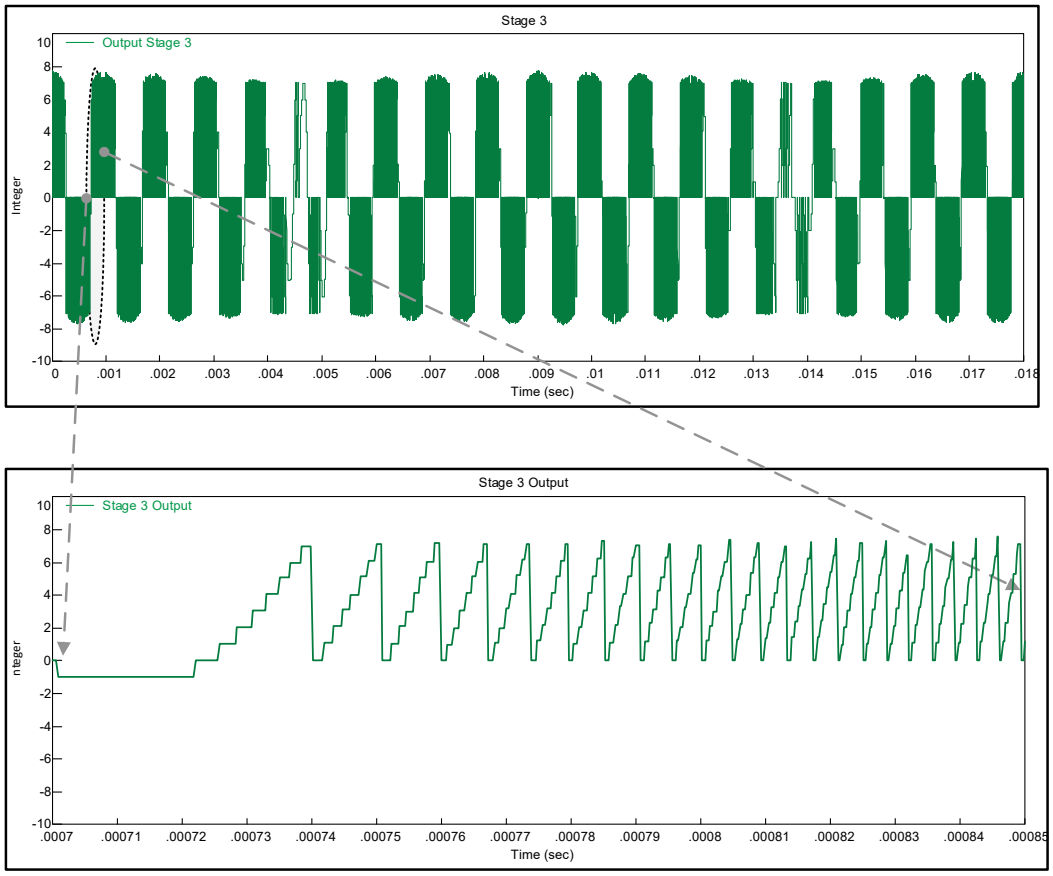


Fig. 36 Simulated pipeline stage 3 output for a three-stage pipeline level-crossing ADC

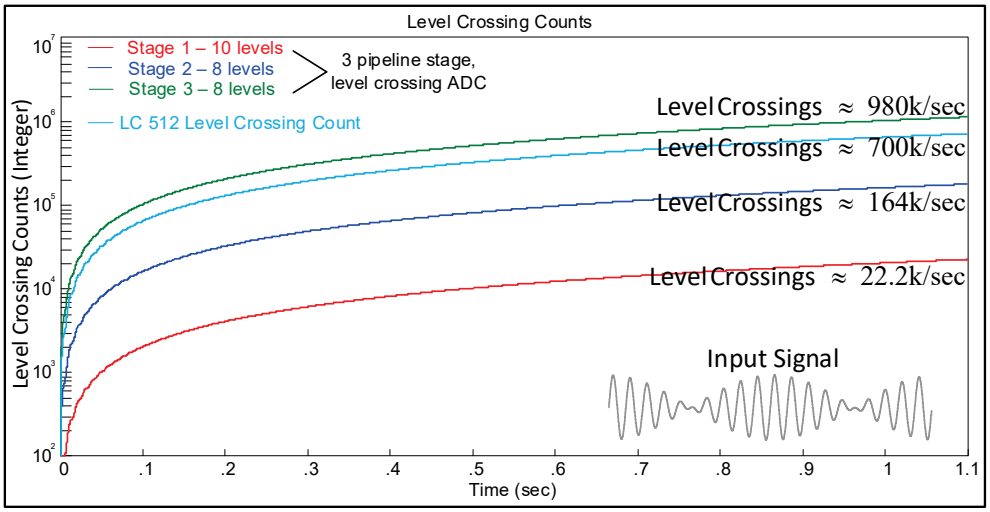


Fig. 37 Comparison of the number of level crossings for each stage in a three-stage pipeline level-crossing ADC to a 512 level-crossing ADC. Figures 33–36 show the stage activity for a two-tone test signal. Stage 1 has a level-crossing activity of 22.2k levels/s. Stage 1 is nearly static compared to stage 3 with 980k levels/s. The light blue line shows that the level-crossing activity for a 512-level ADC is slightly less than stage 3 at 700k levels/s.

Figure 38 compares a two-tone intermodulation distortion test for a simulated three-stage pipeline level-crossing ADC to a 512 level-crossing ADC. In Fig. 35, the level crossings for the three-stage pipeline ADC are slightly larger than the conventional 512 level-crossing ADC. Figure 38 shows that the two-tone IMD for the three-stage pipeline level-crossing ADC is significantly better. This is another advantage of the present invention. For comparison, a conventional 9-bit ADC has an SQNR = 6.02(9) + 1.76 = 56 dB. An SQNR = 56 dB is a much lower performance metric compared to a two-tone intermodulation distortion (IMD) = 57 dB. The pipeline level-crossing ADC is significantly more linear than a comparable 9-bit conventional flash converter.

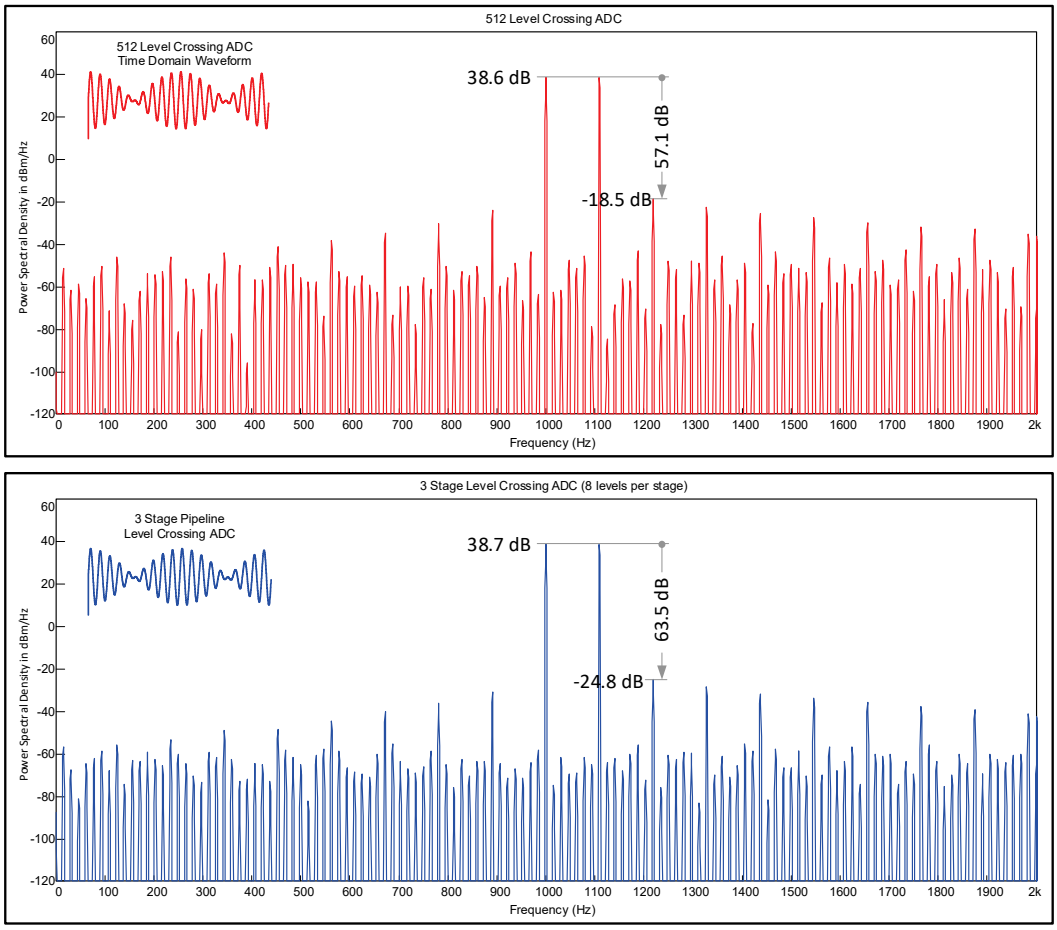


Fig. 38 Comparison of the present continuous-time three-stage pipeline level-crossing ADC to a conventional 512 level-crossing ADC. The present invention provides better intermodulation distortion compared to a conventional level-crossing ADC. The three-stage level-crossing ADC only requires 26 comparators compared to 512 for the 512 level-crossing ADC.

6. CT Software-Reconfigurable Radio

Prior art pipeline flash ADCs suffer from timing and amplitude errors. Complex amplitude error corrections, timing error corrections, and calibration are required. A continuous-time pipeline level-crossing ADC only allows a single level crossing to occur at a time. As illustrated in Fig. 32, each pipeline stage uses the same resistor ladder voltages. The stage gain places the error voltage in between two coarse levels, thereby preventing stage voltage levels from overlapping. This feature is a central part of this invention.

Prior art SRRs are limited by (1) poor intermodulation distortion; (2) a more complicated down-conversion signal chain; and (3) poor performance under jamming and impulse noise. Prior art SRRs using conventional pipeline ADCs suffer from amplitude and time delay errors from the structure of the ADC. Conventional pipeline ADC errors are significant when the input signal crosses a boundary between pipeline stages causing multiple simultaneous bit changes.

The continuous-time SRR offers significantly improved performance over a conventional software-reconfigurable radio. A CT software-reconfigurable radio with a continuous-time pipeline level-crossing ADC offers the follow advantages: (1) Significantly improved intermodulation distortion; (2) simpler digital down-conversion (frequency translation) (continuous-time ADC's output can be multiplied by +1/–1 in phase and quadrature phase square waves for frequency translation); (3) a continuous-time ADC can change only one level at a time; (4) better front-end signal-processing performance using simpler hardware; and (5) better performance under impulse noise conditions.

6.1 Conventional Software-Reconfigurable Radio

Weakly nonlinear devices,[56] like analog multipliers (mixers), have a third-order nonlinearity, $a_3 v_{in}^3(t)$, as illustrated in Eq. 7. A two-tone test signal in Eq. 8 uses two frequencies spaced 10% apart. For a two-tone test signal, a third-order nonlinearity causes intermodulation distortion to occur at $cos2\pi(2f_1 - f_2)t$ and $cos2\pi(2f_2 - f_1)t$. Since $2f_1 - f_2$ and $2f_2 - f_1$ are approximately equal to f_1, both terms fall within the passband. The graph in Fig. 39 shows the power contained in the input signal, $cos2\pi f_1 t$, on the x-axis and the power contained in the intermodulation distortion term, $cos2\pi(2f_1 - f_2)t$, on the y-axis. The input signal's power has a slope of 10 dB/10 dB. The third-order nonlinear term, $cos2\pi(2f_1 - f_2)t$, has a slope of 30 dB/10 dB. As illustrated in Fig. 39, for analog mixers at high-input-signal power levels, the intermodulation distortion power dominates and determines the SNR.

$$v_{out}(t) = a_3 v_{in}^3(t) + a_2 v_{in}^2(t) + a_1 v_{in}(t) + a_0$$

For a two-tone test signal, the $a_3 v_{in}^3(t)$, causes intermodulation distortion at $2f_1 - f_2$ and $2f_2 - f_1$ (7)

$$v_{in}(t) = cos2\pi f_1 t + cos2\pi f_2 t$$

For a two-tone test signal the two frequencies are 10% apart: $f_2 = 1.1f_1$ (8)

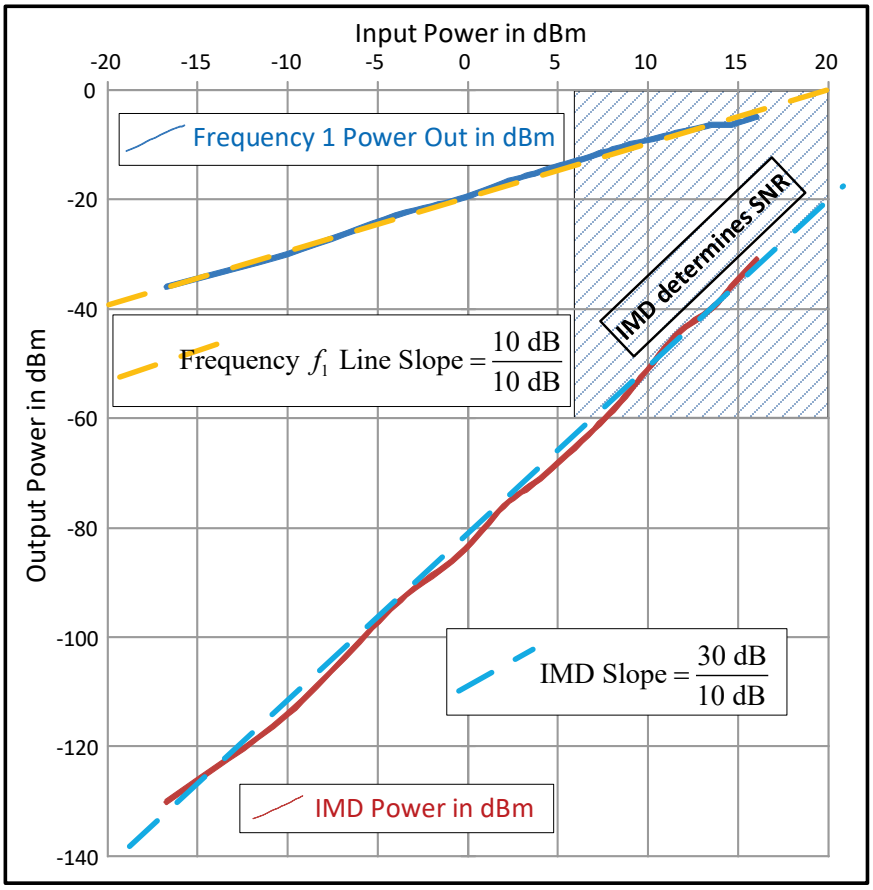


Fig. 39 Typical analog mixer third-order intermodulation distortion. The intermodulation distortion graph shows intermodulation distortion[57] for a typical analog mixer (multiplier). The analog mixer has an intermodulation distortion power with a slope of 30 dB/decade. The input signal power has a slope of 10 dB/decade. At high-input-signal powers, the intermodulation distortion power dominates and determines the SNR.

Figure 40 compares the intermodulation distortion power for a simulated analog mixer to a 64-level, level-crossing ADC. For high-input-signal power levels, the level-crossing ADC is clearly superior (significantly lower intermodulation distortion power) compared to a conventional analog mixer. An example conventional software-reconfigurable radio block diagram is shown in Fig. 41. The analog front-end mixer limits intermodulation distortion performance. Figures 40 and 41 illustrate the potential for a continuous-time software-reconfigurable radio

architecture. In the next section, a continuous-time software-reconfigurable radio architecture is presented.

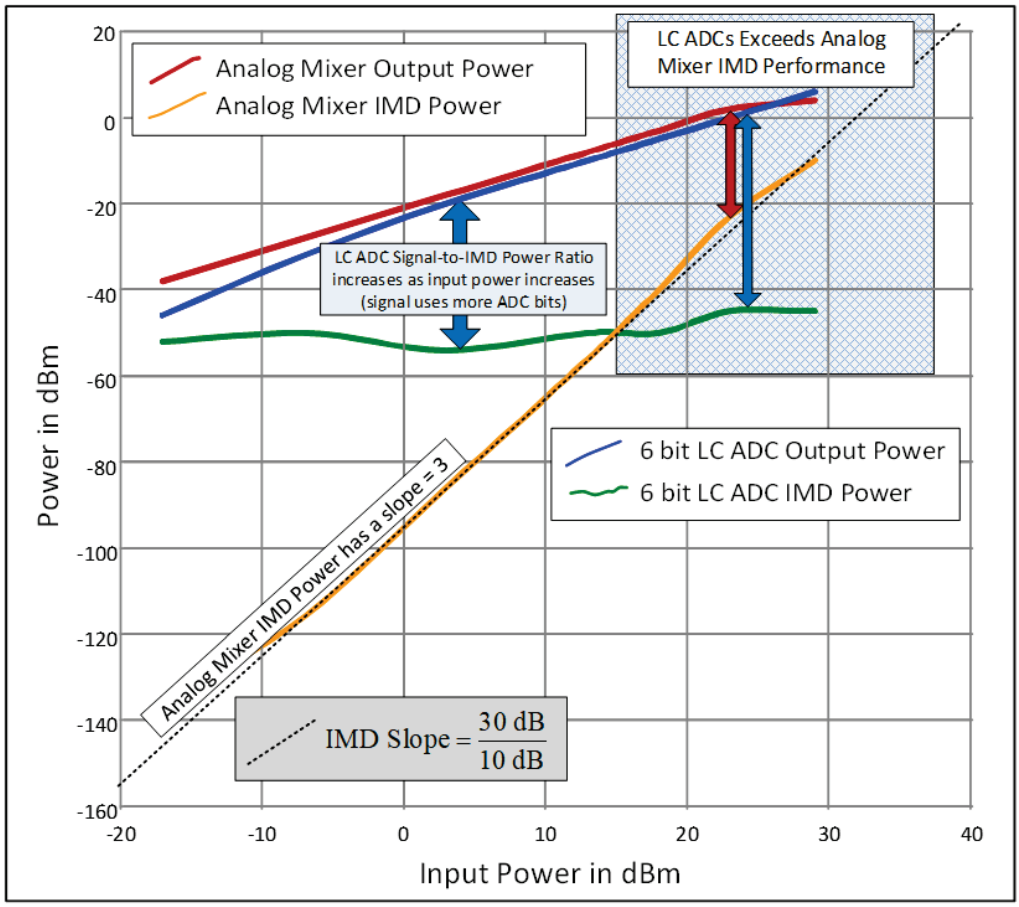


Fig. 40 Comparison of analog mixer (multiplier) to 64-level (6-bit), level-crossing ADC. Level-crossing ADC does not have a significant increase in intermodulation distortion power for high-signal-input powers.

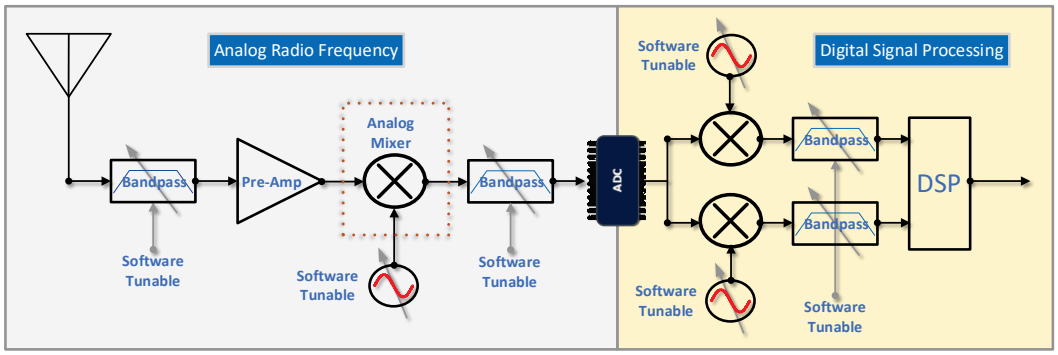


Fig. 41 A conventional software-reconfigurable radio. Analog mixer limits performance of a software-reconfigurable radio. Mixers are limited by dynamic range, intermodulation distortion, mixer conversion loss, and mixer noise.

6.2 Continuous-Time Software-Reconfigurable Radio

Figure 42 illustrates an example continuous-time software-reconfigurable radio architecture. Signal processing may be completely continuous time, or back-end processing can use traditional DSP. The continuous-time ADC is limited by the switching times for the individual comparators. The level-crossing ADC software-reconfigurable radio trades intermodulation distortion for timing jitter, which is approximately constant. For a conventional software-reconfigurable radio architecture, the intermodulation distortion becomes worse as the input signal's power increases.

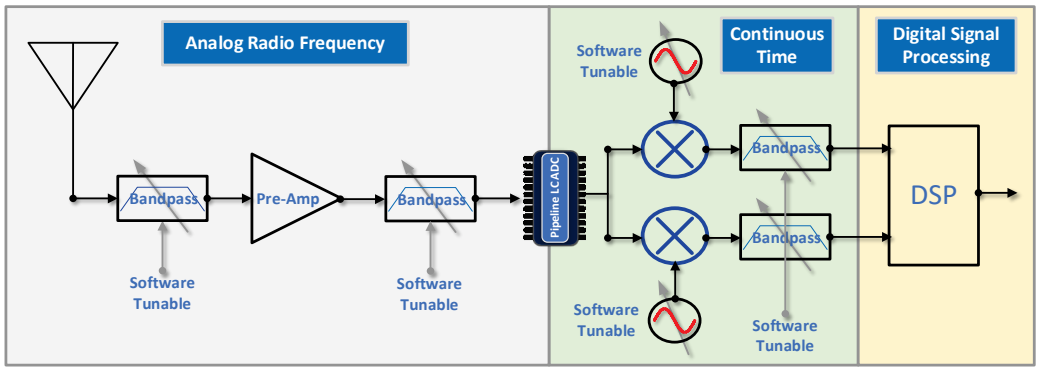


Fig. 42 Software-reconfigurable radio with pipeline level-crossing ADC. Continuous-time multipliers are asynchronous digital multipliers, not analog, and not clocked digital. Asynchronous digital multipliers do not have the intermodulation distortion limitations present in conventional analog mixers (multipliers). All signal processing can be done in continuous time; however, conventional digital signal processors and FPGAs are clocked. By using a clocked latch, continuous time can be simply converted to digital for conventional DSP processing.

Two continuous-time software-reconfigurable radio architectures were simulated using in-phase and quadrature (I/Q) signals equal to cos/sine and in-phase and quadrature-phase square waves as shown in Fig. 43. Figures 44–47 compare in-phase and quadrature down-conversion for a superheterodyne receiver architecture. In a traditional SRR, an analog front end is used to down-convert the signal of interest. Figure 44 uses a complex tone for frequency down-conversion. A +1/–1 amplitude complex square wave is used for down-conversion in Fig. 45. A +1/0 amplitude complex square wave is used for frequency down-conversion in Fig. 46. The complex tone and +1/–1 complex square waves have about the same noise floor level. The +1/–1 complex square wave has an image frequency present near 1600 Hz. The +1/0 complex square wave has a higher noise floor with multiple image frequencies.

Figure 46 compares a continuous-time software-reconfigurable radio using a simulated continuous-time three-stage pipeline level-crossing ADC to an "equivalent" conventional software-reconfigurable radio. The new continuous-time

software-reconfigurable radio clearly shows superior intermodulation distortion. The new architecture has significantly better (lower power level) intermodulation distortion compared to a conventional SRR.

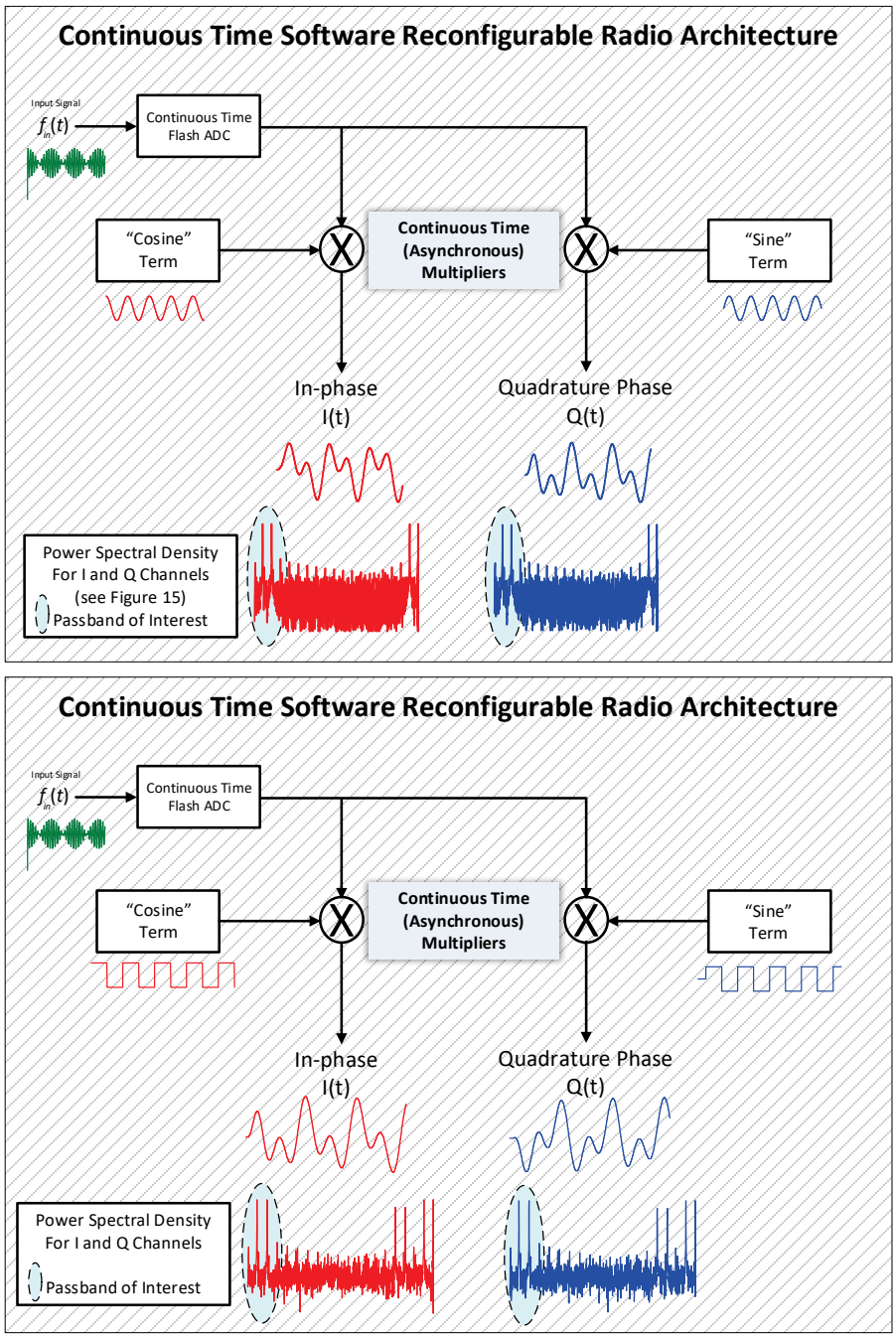


Fig. 43 Continuous-time software-reconfigurable radio simulation. Multipliers are continuous time (asynchronous digital, not analog). Simulations for cosine/sine down-conversion and in-phase square wave and quadrature-phase square wave down-conversion are shown. Both architectures have similar power spectral density graphs. The passband of interest, containing the frequency terms 100 Hz and 211 Hz, is highlighted.

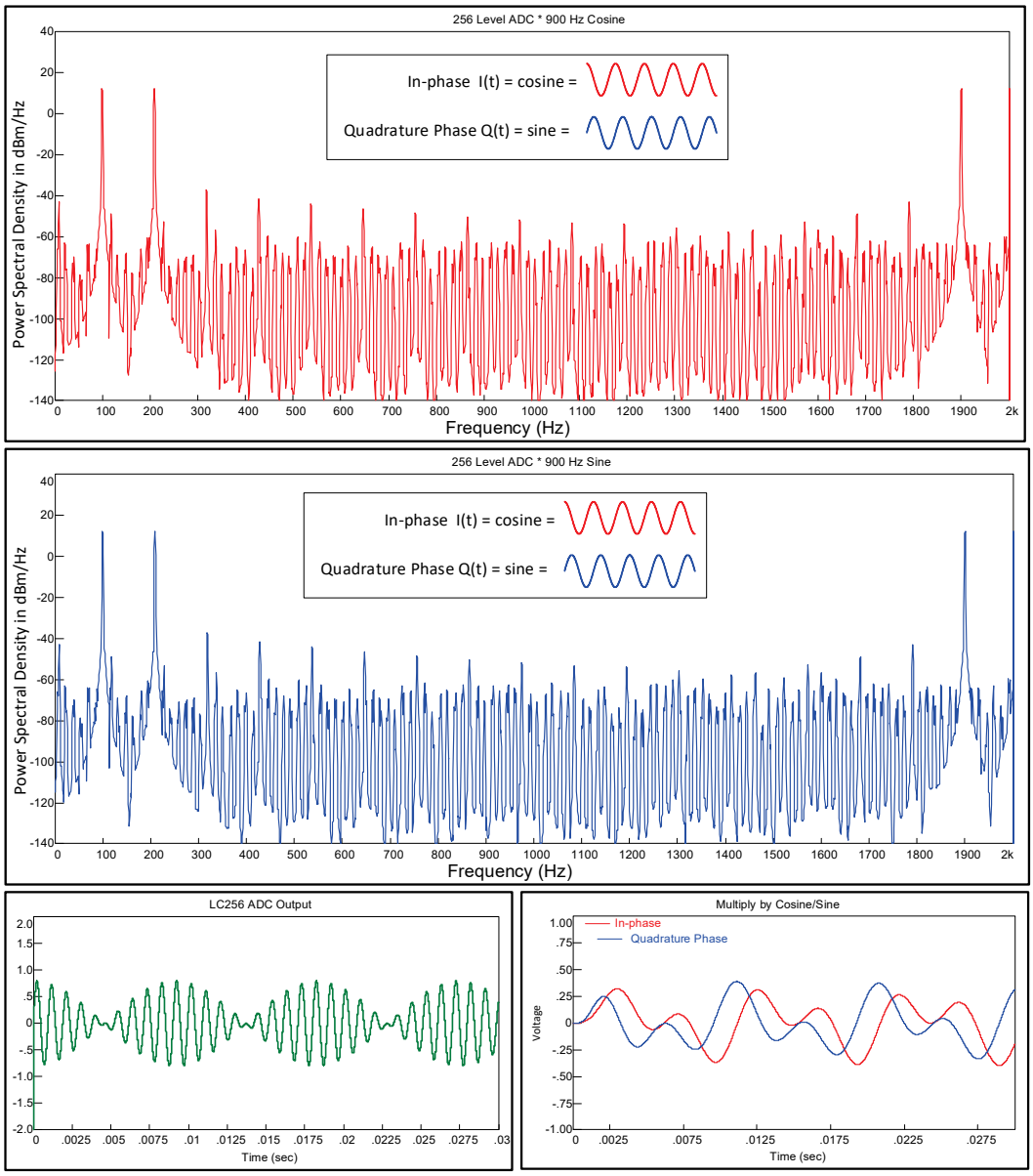


Fig. 44 Software-reconfigurable radio simulation using cosine/sine in-phase and quadrature-phase sine waves. A 256-level, level-crossing ADC was used for the simulations. Input signal is 1.0- and 1.11-kHz sine waves. I/Q modulator is 0.9-kHz complex sine wave. Frequency down-conversion for I/Q channels is shown. Tones at present at 100 and 211 Hz. Bottom graphs are level-crossing ADC time domain waveforms and I/Q channels.

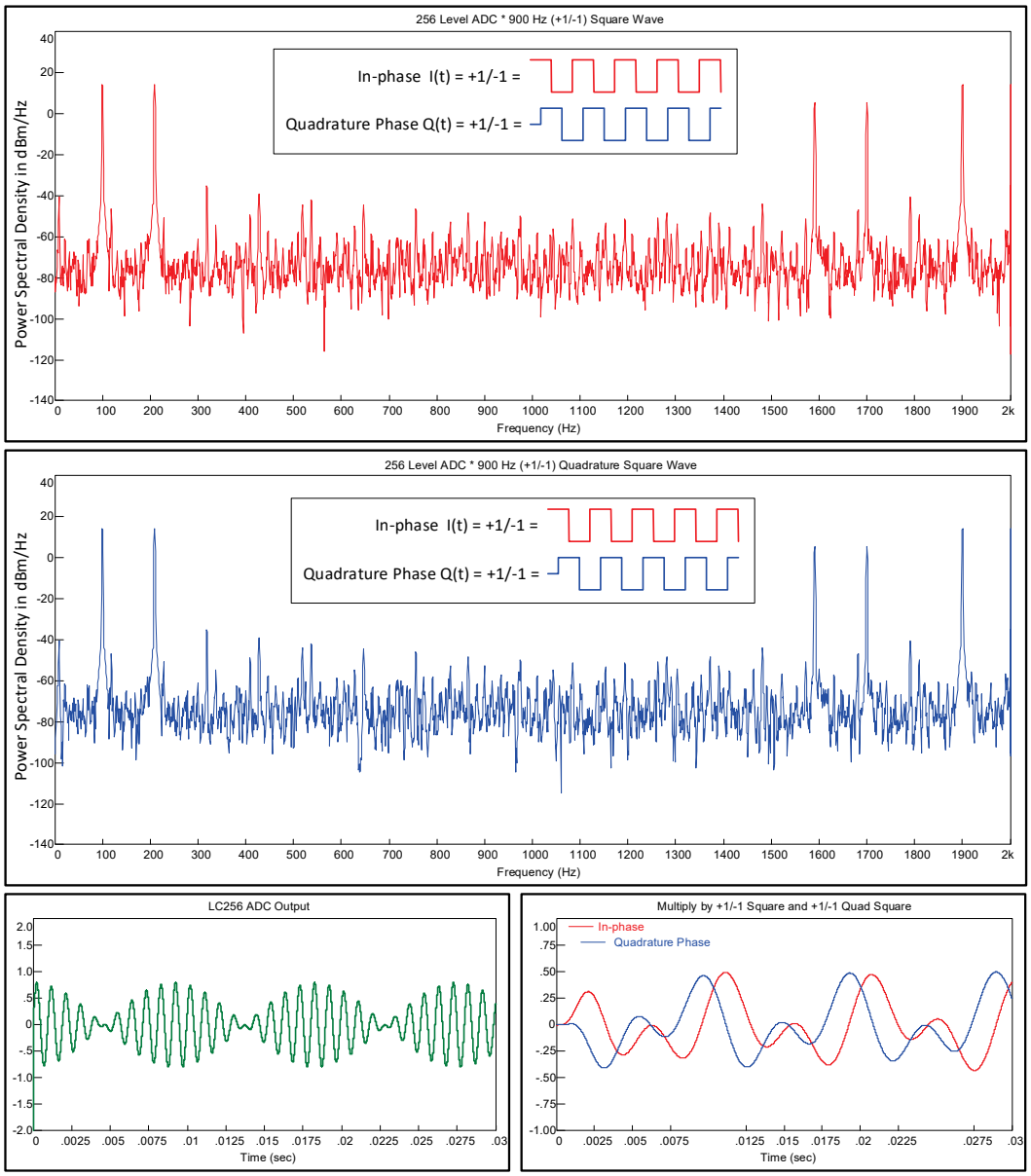


Fig. 45 Software-reconfigurable radio simulation using +1/-1 in-phase and quadrature-phase square waves. A 256-level, level-crossing ADC was used for the simulations. Input signal is 1.0- and 1.11-kHz sine waves. I/Q modulator is 0.9-kHz complex square wave with +1/–1 amplitude. Frequency down-conversion for I/Q channels is shown. Tones are present at 100 and 211 Hz. Image frequencies occur at 1.6 and 1.7 kHz. Bottom graphs are level-crossing ADC time domain waveforms and I/Q channels.

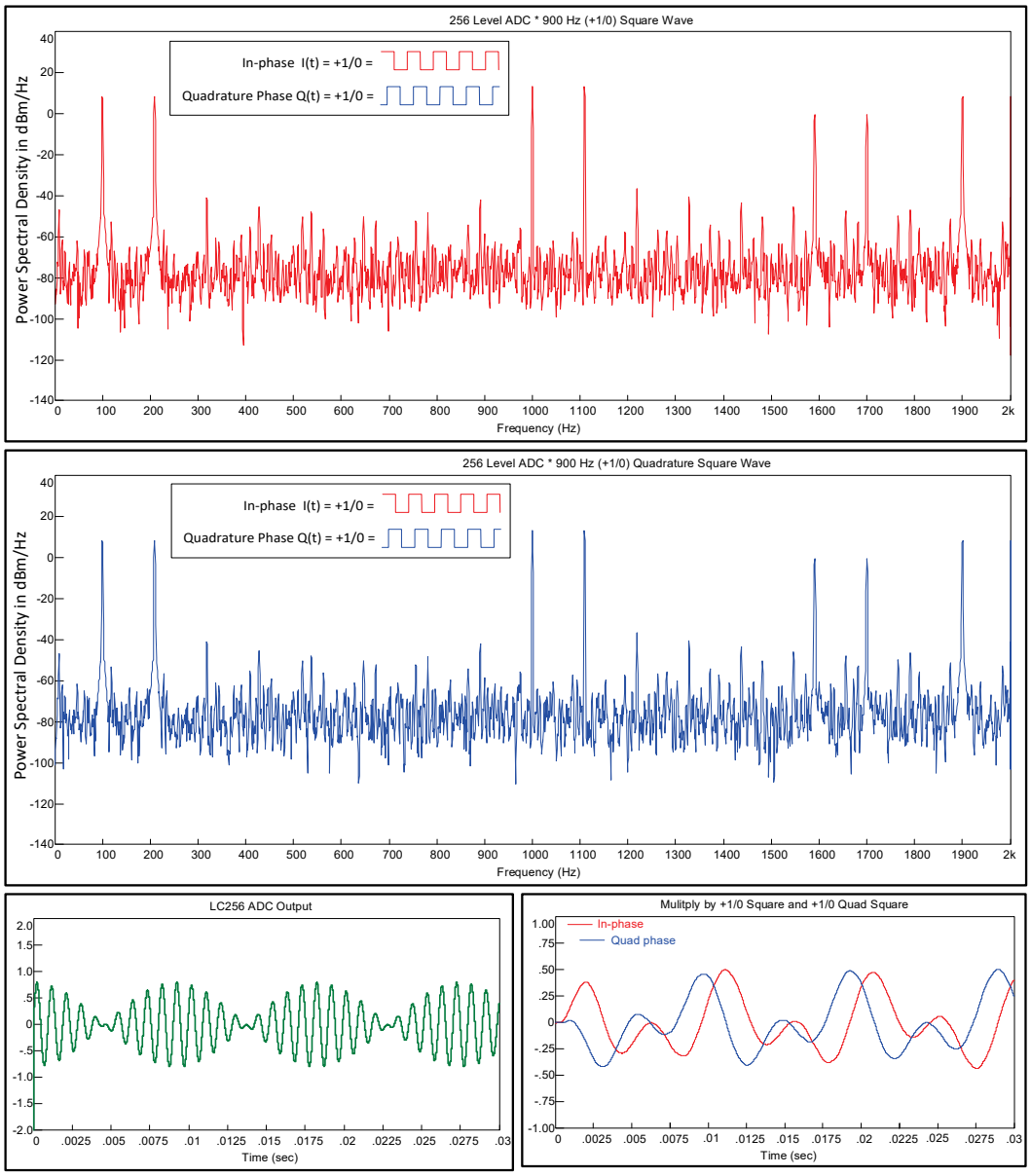


Fig. 46 Software-reconfigurable radio simulation using a +1/0 in-phase and quadrature-phase square waves. A 256-level, level-crossing ADC was used for the simulations. Input signal is 1.0 and 1.11 kHz. I/Q modulator is 0.9-kHz quadrature square waves with +1/0 amplitudes. Frequency down-conversion for I/Q channels are shown. There are tones present at 100 and 211 Hz. There are image frequencies at 1.0, 1.1, 1.6, and 1.7 kHz. Bottom graphs are level-crossing ADC time domain waveforms and I/Q channels.

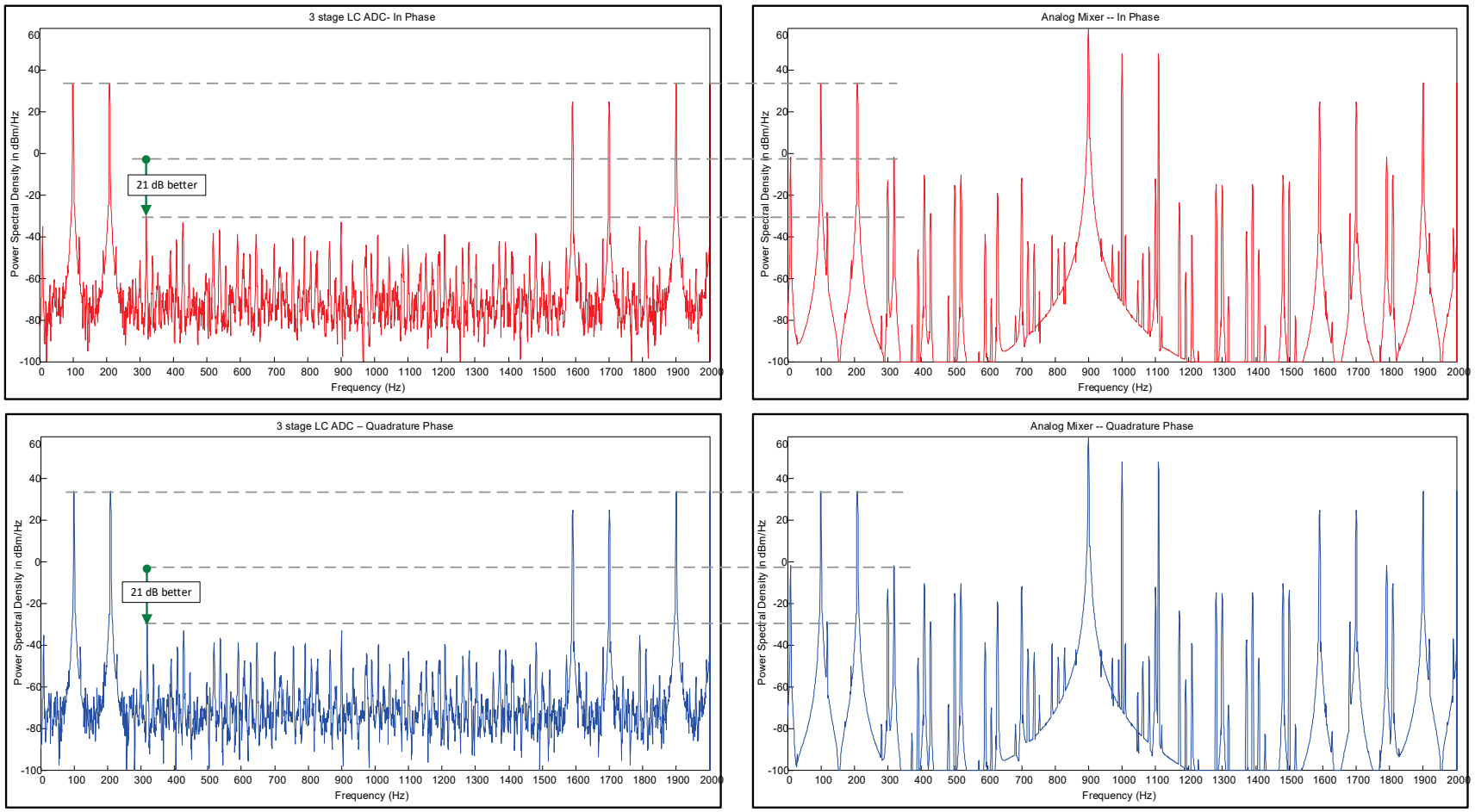


Fig. 47 Comparison of conventional simulated SRR with analog mixer front end and continuous-time SRR. This figure presents a simulation for the continuous-time software-reconfigurable radio using a continuous-time three-stage pipeline level-crossing ADC. Performance is compared to an "equivalent" conventional analog mixer front-end SRR. The graphs demonstrate a significant performance improvement for intermodulation distortion for the continuous-time SRR and continuous-time pipeline level-crossing ADC.

7. Conclusion

In this technical report, the potential for a pipeline level-crossing ADC was investigated. The main advantage of the pipeline level-crossing ADC is that only one state change can occur in a single pipeline stage at a time. This greatly simplifies error correction. The design concepts can also be applied to create a clocked conventional ADC by simply adding a digital output latch.

Conventional software-reconfigurable radios are limited by intermodulation distortion present in analog mixers. A software-reconfigurable radio architecture based on a pipeline level-crossing ADC was also researched. Simulations show a potential 20-dB improvement in intermodulation distortion power in Fig. 46. A provisional patent application has been filed for the continuous-time pipeline level-crossing ADC and continuous-time software-reconfigurable radio architecture.

8. References

1. Aeschlimann F, Allier E, Fesquet L, Renaudin M. Asynchronous FIR filters: towards a new digital processing chain. 10th International Symposium on Asynchronous Circuits and Systems; 2004 Apr. p. 198–206. doi: 10.1109/ASYNC.2004.1299303.

2. Tsividis Y. Event-driven data acquisition and digital signal processing—a tutorial. IEEE Transactions on Circuits and Systems—II: Express Briefs. 2010;57(8).

3. Jungwirth P, Crowe W, inventors. Continuous time pipeline level crossing ADC and continuous time software reconfigurable radio architecture. US Provisional Patent Application No. 63353038. 2022 June.

4. Teifel J, Manohar R. An asynchronous dataflow FPGA architecture. IEEE Transactions on Computers. 2004 Nov;53(11):1376–1392. doi: 10.1109/TC.2004.88.

5. Oliveira DL, Bompean D, Faria LA, Oliveira JLV. Design of asynchronous systems on FPGA using direct mapping and synchronous specification. 2013 International Conference on Reconfigurable Computing and FPGAs (Reconfig); 2013. p. 1–6. doi: 10.1109/ReConFig.2013.6732276.

6. Kato H, Saito H. Design of asynchronous CNN circuits on commercial FPGA from synchronous CNN circuits. 2019 IEEE 13th International Symposium on Embedded Multicore/Many-Core Systems-on-Chip (MCSOC); 2019. p. 61–67. doi: 10.1109/MCSoC.2019.00016.

7. Dashkin R, Manohar R. General approach to asynchronous circuits simulation using synchronous FPGAs. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems. doi: 10.1109/TCAD.2021.3131546.

8. Box GEP. Robustness in the strategy of scientific model building. In: Launer RL, Wilkinson GN, editors. Robustness in statistics. Academic Press; 1979. p. 201–236. doi:10.1016/B978-0-12-438150-6.50018-2, ISBN 9781483263366.

9. Philips. The history of the CD [accessed 2022 May 9]. https://www.philips.com/ a-w/research/technologies/cd/technology.cs.

10. Immink K. Compact disc story. Journal of the Audio Engineering Society. 1998 May;46(5):458–464. https://www.researchgate.net/publication/294484774_Compact_disc_story.

11. Peek J. Communications aspects of the compact disc digital audio system. IEEE Communications Magazine. 1985 Feb;23(2):7–15. doi: 10.1109/MCOM.1985.1092514.

12. John von Neumann. Wikipedia; last updated 2022 June 23. https://en.wikiquote.org/wiki/John_von_Neumann.

13. Lecky WEH. Democracy and liberty. 1899.

14. Tsividis Y. Continuous-time digital signal processing. Electronics Letters. 2003;39(21):1551.

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

16. Li Y, Shepard K, Tsividis Y. Continuous-time digital signal processors. 2005 Apr. p. 138–143. doi: 10.1109/ASYNC.2005.15.

17. DeForest L, inventor. Space telegraphy. United States patent US879532A. 1908 Feb.

18. Armstrong E. A new system of short-wave amplification. Proceedings of the IRE. 1921;(9):3–11.

19. Armstrong E, inventor. Method of receiving high frequency oscillations. United States patent 1,342,885. 1920 June.

20. Kline R. Harold Black and the negative-feedback amplifier. IEEE Control Systems Magazine. 1993 Aug;13(4):82–85. doi: 10.1109/37.229565.

21. Philbrick G. Designing industrial controllers by analog. Electronics. 1948 June;108–111.

22. Shannon C. A mathematical theory of communications. Bell Systems Technical Journal. 1948 July; 27.

23. FCC. NTSC signal specifications. 1953 Dec. https://antiqueradio.org/art/NTSC%20Signal%20Specifications.pdf.

24. RCA. Radio and Electronics Research-Engineering Journal. 1953 June;14(2).

25. Bell Labs. The breakthrough that enabled digital imaging — from DSLR cameras to medical endoscopes; n.d. [accessed 2022 May 16]. http://www.bell-labs.com/about/history/innovation-stories/charge-coupled-device/#gref.

26. Barbe DF, Baker WD, Davis KL. Signal processing with charge-coupled devices. IEEE Journal of Solid State Circuits. 1978 Feb;SC-13(1):34–51.

27. Tower J, Elliott L, Gandolfo D. Analog-binary programmable transversal filter. RCA Advance Technology Laboratories; 1980 Feb. Final Technical Report No.: RADC-TR-79-331.

28. IEEE. Fifty years of signal processing: The IEEE Signal Processing Society and its technologies 1948–1998. The IEEE Signal Processing Society; 1998. https://signalprocessingsociety.org/uploads/history/history.pdf.

29. Bennett W. Spectra of quantized signals. Bell System Technical Journal. 1948 July;27:446–471.

30. Peter TV. Radar, past, present and future. IEEE Transactions. 1975 Oct:236–239.

31. MIT-Lincoln Lab. SAGE: Semi-Automatic Ground Environment Air Defense System; n.d. [accessed 2022 May 16]. https://www.ll.mit.edu/about/history/sage-semi-automatic-ground-environment-air-defense-system.

32. NORAD tracks Santa. https://www.noradsanta.org/en/.

33. Beutler FJ. Error-free recovery of signals from irregularly spaced samples. SIAM Rev. 1966 July;8(3):328–335. doi: 10.1137/1008065.

34. Inose H, Aoki T, Watanabe K. Asynchronous delta-modulation system. Electronics Letters. 1966 Mar;2(3):95–96. doi: 10.1049/el:19660077.

35. Tsividis Y. Continuous-time digital signal processing. Electronics Letters. 2003;39(21):1551.

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

37. PEO M&S. Continuous-time digital signal processing (DSP) using reconfigurable devices status. PEO M&S; 2019 Dec. SBIR A20-080. https://www.sbir.gov/node/1654463.

38. PhysioNet. Open Source ECG Data Base. Ver. 1.0.0. MIT; 2014 March 6. https://physionet.org/content/ecgiddb/1.0.0/.

39. Antony A, Paulson SR, Moni DJ. Asynchronous adaptive threshold level crossing ADC for wearable ECG sensors. J Med Syst. 201943:78. https://doi.org/10.1007/s10916-019-1186-8.

40. Zhang X, Lian Y. A 300-mV 220-nW event-driven ADC with real-time QRS detection for wearable ECG sensors. IEEE Transactions on Biomedical Circuits and Systems. 2014 Dec:8(6):834–843. doi: 10.1109/TBCAS.2013.2296942.

41. Rodhe U. Digital HF radio: a sampling of techniques. Third International Conference on HF Communication Systems and Techniques; 1984 Feb; London. https://www.microwavejournal.com/articles/27714-dr-ulrich-l-rohde-gives-the-6th-sir-j-c-bose-memorial-lecture-at-ieee-hyderabad-section.

42. Johnson P. New research lab leads to unique radio receiver. E-Systems Team. 1985;5(4):6–7. http://chordite.com/team.pdf.

43. California Dept of Water Resources. Yosemite at Yosemite Valley; 2022 Apr. https://cdec.water.ca.gov/dynamicapp/selectQuery?Stations=YYV&SensorNums=4&dur_code=H&Start=2022-04-01T00%3A00&End=2022-04-30T23%3A00.

44. National Park Service. Half Dome, Yosemite National Park, 2022 Apr. https://www.nps.gov/media/webcam/view.htm?id=81B464BC-1DD8-B71B-0B0F6F15CBA4A034.

45. Arizona State University. World Meteorological Organization's World Weather & Climate Extremes Archive; 2022 Apr. https://wmo.asu.edu/content/world-lowest-temperature.

46. Arizona State University. World Meteorological Organization's World Weather & Climate Extremes Archive; 2022 Apr. https://wmo.asu.edu/content/world-highest-temperature.

47. Burt C. Extreme short-duration temperature changes in the U.S; 2018 Mar. https://www.wunderground.com/cat6/extreme-short-duration-temperature-changes-us.

48. Grey R. Quantization. IEEE Transactions on Information Theory. 1998 Oct;44(6).

49. Lee D, Ferreira P. Direct construction of superoscillations. IEEE Transactions on Signal Processing. 2014 June 15;62(12):3125–3134.

50. Lazar A, Tóth L. Perfect recovery sensitivity analysis of time encoded bandlimited signals. IEEE Trans Circuits Syst. I Reg Papers. 2004 Oct;51(10):2060–2073.

51. Kurchuk M, Weltin-Wu C, Morche D, Tsividis Y. Event-driven GHz-range continuous-time digital signal processor with activity-dependent power

dissipation. IEEE Journal of Solid-State Circuits. 2012 Sep;47(9):2164–2173. doi:10.1109/JSSC.2012.2203459.

52. Sequoia & Kings Canyon National Parks. Park atlas; 2022 Apr. https://experience.arcgis.com/experience/0487dc59e08844cfa347e222dee3aa79/.

53. Marisa T, Niederhauser T, Haeberlin A, Wildhaber RA, Vogel R, Goette J, Jacomet M. Pseudo asynchronous level crossing ADC for ECG signal acquisition. IEEE Transactions on Biomedical Circuits and Systems. 2017 Apr;11(2):267–278.

54. Hou Y, Yousef K, Atef M, Wang G, Lian Y. A 1-to-1-kHz, 4.2-to-544-nW, multi-level comparator based level-crossing ADC for IoT applications. IEEE Transactions on Circuits and Systems II: Express Briefs. 2018 Oct;65(10):1390–1394.

55. Pavan S. Shibata H. Continuous-time pipelined analog-to-digital converters: a mini-tutorial. IEEE Transactions on Circuits and Systems II: Express Briefs. 2021 Mar;68(3):810–815. doi: 10.1109/TCSII.2020.3048850.

56. McVay F. Don't guess the spurious levels of an amplifier. Electronics Design. 1967 Feb;70–73.

57. Mini-Circuits. Improve two-tone, third-order intermodulation testing; n.d. [accessed 2022 July 8]. https://www.minicircuits.com/appdoc/AN00-008.html.

List of Symbols, Abbreviations, and Acronyms

ADC	analog-to-digital converter
ARL	Army Research Laboratory
ASIC	application-specific integrated circuit
CCD	charge-coupled device
CT	continuous time
CT-ADC	continuous-time, pipeline analog-to-digital converter
CT-DSP	continuous-time digital signal processing
CT-SRR	continuous-time software-reconfigurable radio
DAC	digital-to-analog converter
DEVCOM	US Army Combat Capabilities Development Command
DFT	discrete Fourier transform
DSP	digital signal processing
ECG	electrocardiogram (also EKG)
FFT	fast Fourier transform
FPGA	field-programmable gate array
FM	frequency modulation
IC	integrated circuit
I/Q	in-phase and quadrature
IoT	Internet of Things
LTI	linear time invariant
NTSC	National Television System Committee
OSR	oversampling ratio
SAGE	Semi-Automatic Ground Environment
SDR	software-defined radio
SNR	signal-to-noise ratio
SQNR	signal-to-quantization noise ratio

SRR	software-reconfigurable radio
SWaP	size, weight, and power
ZOH	zero-order hold

1 (PDF)	DEFENSE TECHNICAL INFORMATION CTR DTIC OCA
1 (PDF)	DEVCOM ARL FCDD RLD DCI TECH LIB
1 (PDF)	DEVCOM ARL FCDD RLC NC P JUNGWIRTH