

Design of high speed pipelined ADC System

Dheepak Mohanraj
Department of EEE, AMET University
Chennai, India

Abstract—The goals for most of today’s receiver system are sampling at high-speed, with high resolution and with as few errors as possible. This project report describes the design of a high-speed ADC system. The system is designed with a parallel Analog-to-digital converter (ADC) architecture, also called time interleaving. It aims to increase the sampling speed of the system. The system described in this report uses eight 8-bit ADCs in parallel. Each ADC can sample at 1.7875 MHz and the total sampling speed will then theoretically become 14.3 Ms/s the system has been designed in a 0.25 μ m CMOS technology and simulated in WinSpice.

Keywords—Analog-to-digital converter, Nyquist frequency, sample-and-hold amplifier, Successive Approximation Register.

I. INTRODUCTION

Today, almost every communication system works with signals in the digital domain. This creates a need for a good Analog-to-digital conversion unit. This unit is called an Analog-to-digital converter (ADC). The ADC is a key component in systems for radio communication, digital signal processing and measuring. It also plays a key role in many other systems where an analog signal is an input. Ideally, the ADC has infinite resolution and is error free, but in reality the resolution is limited and errors are unavoidable even if they can be very small.

RF receiver systems use down conversion of a Radio Frequency (RF) signal to convert it to a low Intermediate Frequency (IF) or to base band. Instead of doing this down conversion, it would be much better if it was possible to just sample the signal where it is because down conversion introduces more noise and errors to the system. To be able to do such sampling, a very broad banded sampling system is needed. Beside of radio receiver systems, another application that will have benefits of increased bandwidth is signal reconnaissance with radar. With a larger bandwidth, the system could listen to a broader frequency band without sweeping the frequency.

To achieve the goal mentioned above (i.e. high speed), CMOS technology is very attractive for several reasons. First, the low cost and high integration level has made the CMOS technology superior over bipolar technology. Because of that, several CMOS design techniques have been developed. And with the scaled CMOS technology, it can achieve the high speed, which was once reserved for bipolar or other fast processes. With the above motivations, the goal of this project is to build a high speed Pipelined A to D converter based on time interleaved operation in CMOS technology, i.e. to design a high-speed broadband sampling system with time interleaved

ADCs architecture for a sampling rate of 14.3MS/s. For verification, the design is simulated using WinSpice

II. ANALOG TO DIGITAL CONVERSION

An ADC converts an analog time-continuous signal waveform to a time-discrete signal by sampling. Sampling is the technique to represent a continuous-time signal with a sequence of time-discrete values (in this case, binary values). The signal is usually band limited with bandwidth B and sampled at uniform time intervals, TS . This will in the frequency domain correspond to a sample frequency, $f_s=1/TS$. To ensure that the sampled signal can be reconstructed exactly from the samples, the sample frequency, f_s , is required to be at least two times the signals highest frequency component. This requirement is known as the Nyquist Theorem and $f_s/2$ is called the Nyquist frequency. Sampling at twice the signal frequency is called Nyquist sampling. If an analog signal has frequency components above the Nyquist frequency it will have image overlap and aliasing distortion because of the sampling, see the upper part of figure 1. With signals below the Nyquist frequency, these phenomena will be easily avoided. Low pass filtering the signal before sampling would be enough to overcome this problem. This is called anti-alias filtering

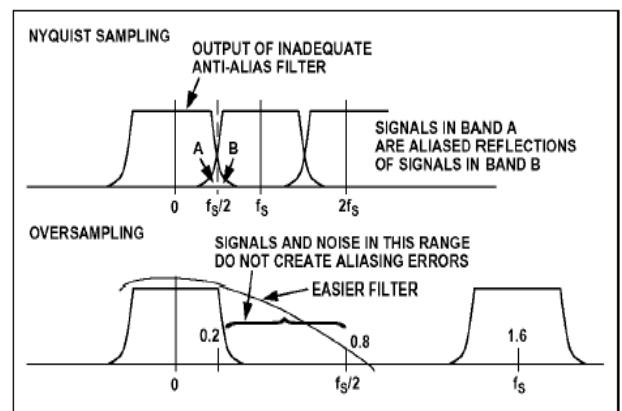


Fig 1. Nyquist sampling and oversampling.

Sampling at a higher rate than 2 times the analog signal frequency is called oversampling. One of the advantages with oversampling is that only a simpler anti-aliasing filter is required. This is because, with a higher sampling rate, the mirrored signal images will be more separated from each other and that will ease the requirements in the transition band of the filter. If an oversampled signal is digitally decimated to a rate closer to Nyquist, an advantage called conversion gain will occur. A conversion gain of 3dB is achieved for every “factor-

of-two" decimation due to a reduction in quantization noise with 3dB. Undersampling is not quite the opposite of oversampling. Undersampling takes advantage from the frequency translation caused by aliasing

Quantization is when the time-continuous analog signal with an infinite number of amplitude levels maps into a set or finite values represented by a limited number of bits. Quantization always introduces an error because some information in the signal is lost. When the number of levels in the quantizer is large and the input signal is sufficiently random, the quantization error is well approximated as a random white noise process, which is uncorrelated with the input signal. Most commonly used is uniform quantization, where input thresholds and output values are evenly spaced

III. CHOOSING AN ADC

It seems that an ever increasing number of projects require an Analogue to Digital Converter (ADC). This is of course no surprise. Many projects are interfacing with the real world, which as far as we can tell is analogue. There are a few main items to consider, before selecting which ADC is best suited to your project.

- Availability/Cost
- Precision/Resolution (number of bits)
- Accuracy (error in conversion)
- Speed (time for one conversion, different from clock speed)
- Voltage Range
- Multiplexing
- Ease of Use

The first is **availability**. Are there chips readily available that could meet the needs of your project? The most readily available chip is the ADC0809, which is an 8 bit, 8 channels, Successive Approximation Register (SAR) ADC. More information on what this means will follow. The chip can be obtained from the Electronics "Store" in the electrical engineering building, but will this chip meet your needs? Also think about how much a chip **costs**. A more expensive chip is only necessary if it is the only one that will actually meet your needs

Precision or **resolution** is the next consideration. How many bits of resolution will you require? When you are considering this, you must also keep in mind the size of the FPGA that you are interfacing with. Added resolution will not help if you cannot process it. (The ADC0809 has 8 bit resolution)

When comparing the **accuracy** of different ADCs you need to look at the amount of error in the LSB. comparing the **accuracy** of different ADCs you need to look at the amount of error in the LSB. Different types of ADCs are better at producing more accurate results while others are faster.

There is an obvious trade-off between accuracy, and **speed**. It takes longer to do a more accurate conversion. When considering the speed of an ADC, you need to look at

how long it takes to do a conversion, not the clock speed. That is because certain types of ADCs require multiple clock cycles to complete a single conversion. Consider how often you would require new data, and select an ADC accordingly. Also decide if speed is more important and accuracy and vice versa. Also consider how long your input signal will remain stable. ADCs require input to be stable the length of a conversion to be accurate.

The **range of voltage** over which the ADC is capable of doing conversions, essentially the positive reference voltage and the negative reference voltage levels. Voltage dividers can also be used to bring the input voltage within the range of the converter.

Multiplexing does your project require analog input from more than one source? If so consider multiplexing the data. It will save a lot of wiring and input pins on the FPGA. The drawback is that the interface becomes more complicated. Many ADCs are available with multiple channels. (The ADC0809 has 8 channels available)

Ease of use is also an important factor to consider. The less time you have to spend getting an ADC to work the more time you'll have to make your project do really cool things. The less control signals required to run the converter, the simpler the interface will be. Also if there are application notes and previously written code available it should be easier to use as well.

IV. PIPELINED ADC ARCHITECTURE

Pipeline ADCs provide an optimum balance of size, speed, resolution, power dissipation, and analog design effort, they have become increasingly attractive to major data-converter manufacturers and their designers. Also known as sub ranging quantizes, pipeline ADCs consist of numerous consecutive stages, each containing a track/hold (T/H), a low-resolution ADC and DAC, and a summing circuit that includes an interstage amplifier to provide gain.

Target applications for pipeline ADCs include communication systems, in which total harmonic distortion (THD), spurious-free dynamic range (SFDR), and other frequency-domain specifications are significant; CCD-based imaging systems, in which favorable time-domain specifications for noise, bandwidth, and fast transient response guarantee quick settling; and data-acquisition systems, in which time- and frequency-domain characteristics are both important are both important.

This section explains the architecture and operation of pipelined analog-to-digital converters (ADCs). The pipelined analog-to-digital converter (ADC) has become the most popular ADC architecture for sampling rates from a few mega samples per second (MS/s) up to 100MS/s, with resolutions from 8 bits at the faster sample rates up to 16 bits at the lower rates. These kinds of resolutions and sampling rates cover a wide range of applications, including CCD imaging, ultrasonic medical imaging, digital receiver, base station, digital video (for example, HDTV), xDSL, cable modem, and fast Ethernet. Lower-sampling-rate applications are still the domain of the successive approximation register (SAR) and integrating architectures (and more recently over sampling/sigma-delta ADCs), whereas the highest sampling rates (a few hundred

MS/s or higher) are still obtained using flash ADCs and their variants. However, it is safe to say that pipelined ADCs of various forms have improved greatly in speed, resolution, dynamic performance, and low power in recent years.

The ADC pipeline architecture effectively overcomes the limitations of the flash architecture. A pipelined converter divides the conversion task into several consecutive stages. Each of these stages consists of a sample and hold circuit, an m-bit ADC (e.g., a flash converter), and an m-bit D/A converter (DAC).

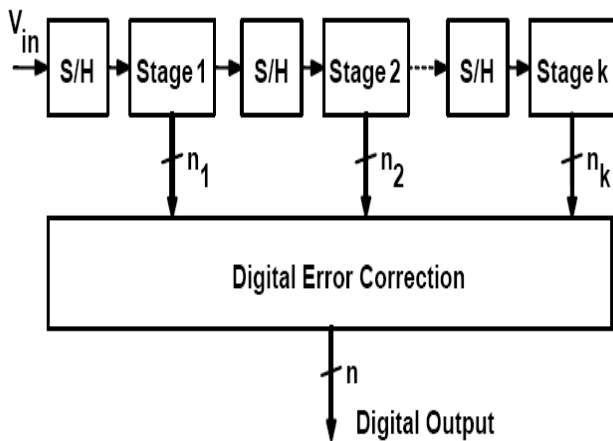


Fig 2. Pipelined ADC architecture

First the sample and hold circuit of the first stage acquires the signal. The m-bit flash converter then converts the sampled signal to digital data. The conversion result forms the most significant bits of the digital output. This same digital output is fed into an m-bit digital-to-analog converter, and its output is subtracted from the original sampled signal. The residual analog signal is then amplified and sent on to the next stage in the pipeline to be sampled and converted as it was in the first stage. This process is repeated through as many stages as are necessary to achieve the desired resolution. Pipelined converters achieve higher resolutions than flash converters containing a similar number of comparators

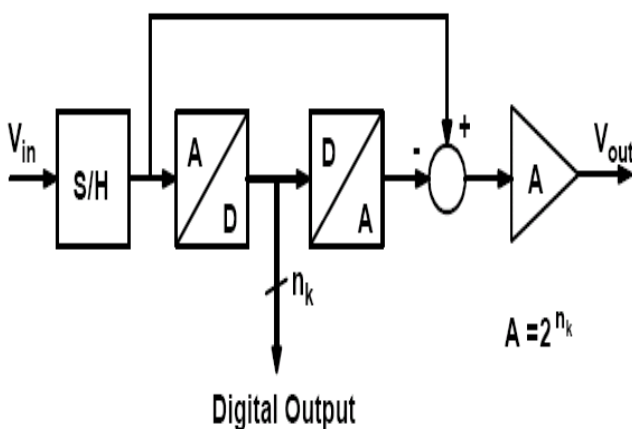


Fig 3. One pipelined stage of ADC

This block diagram contains a sample-and-hold amplifier (SHA), a low-resolution analog-to-digital converter (ADSC), a low-resolution digital-to-analog converter (DAC), and subtracted.

In operation, each stage initially sample and holds the output from the previous stage. Next, the held input is converted into a low-resolution digital code by the ADSC and back into an analog signal by the DAC. Finally, the DAC output is subtracted from the held input, producing a residue that is sent to the next stage for further conversion.

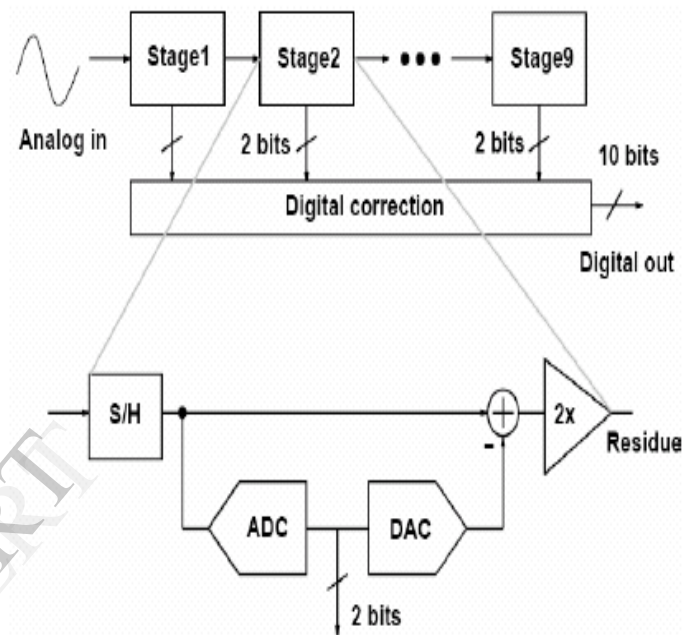


Fig 4. Architecture of 1.5 bit/stage

A single 1.5 bits stage consists of an S/H, an ADC, a DAC and a summing gain of 2 amplifiers as can be seen in figure 1.2. One bit of resolution of stage contributes to the over all resolution of the ADC, resolving 1 bit per stage, and reserving 0.5bits for error correction

The implementation of the ADC consists of a S/H followed by passing output of the S/H through an algorithm to generate the output bits. The algorithm is based on a pipeline ADC and follows.

1. The input signal is sampled and hold.
2. This hold signal is input to a comparator that compares the input value to a ref voltage.
3. If the input signal is greater than the ref voltage, the output bit is set to a high and the reference signal is subtracted from the input. The difference is multiplied by two and passed to the output of stage.
4. If the input signal is less than the ref voltage, the output bit is set low. The input signal is multiplied by two and passed to the output of the stage.
5. This output is used as the input to the next stage and steps 2,

3 and 4 are repeated. This continues for N stages (N-No. of bits in the ADC).

V. SIMULATION RESULTS

This section gives the simulation results obtained in WinSpice for the S/H, ADC based on pipeline algorithm, 8:1 multiplexer and also that for the complete time interleaved topology.

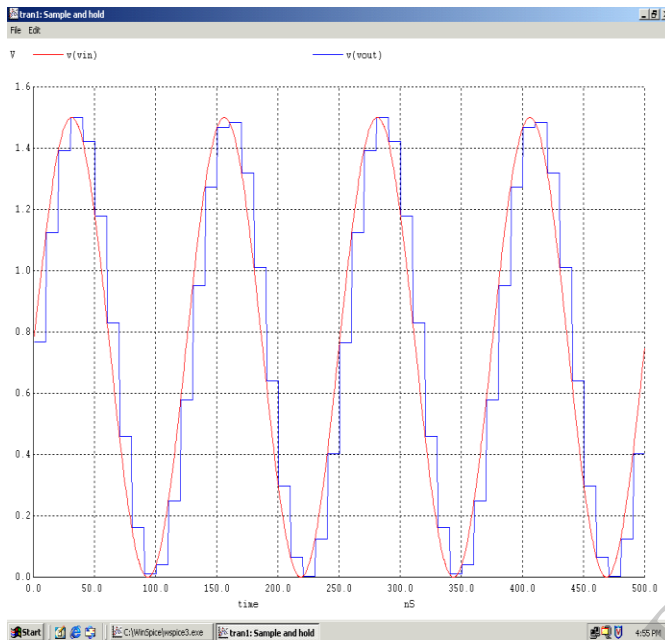


Fig 5. Output of the S/H

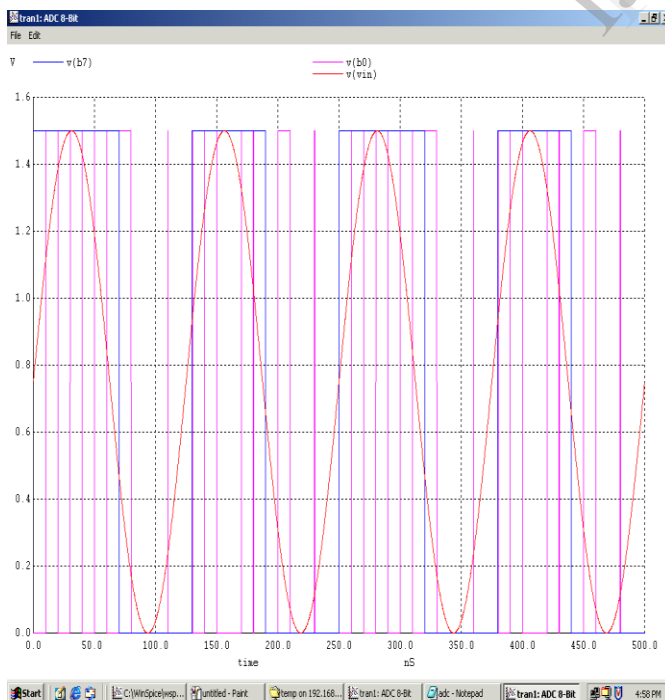


Fig6.Output of the individual ADC

VI. CONCLUSION

There are many possibilities to develop this sampling system further. The components available on the shelf will become better and better and then it will be easier to design a system with better performance than this. Finding better circuits than I have found will increase performance quite a lot. Finding a faster ADC with at least 12-bits is also of high priority.

In an approach to increase the systems performance even further, a timing error estimation algorithm can be used on the sampled data. Reduction of supply voltage is also one possible improvement.

REFERENCES

- [1] R.Jacob Baker, Harry W. Li, David E. Boyce. "CMOS Circuit Design, Layout and Simulation".
- [2] David Johns, Ken Martin. "Analog Integrated Circuit Design".
- [3] Behzad Razavi. "Design of Analog CMOS Integrated Circuits".
- [4] Phillip E. Allen, Douglas "CMOS Analog Circuit Design".
- [5] R.Jacob Baker. "CMOS Mixed Signal Circuit Design".
- [6] Technical article, "Multiply Your Sampling Rate with Time-Interleaved Data Converters,"