

# A Low-power, Un-clocked “Chain” ADC For Use In Large Channel-Count Detector ASIC’s

Paul Stankus  
Oak Ridge National Laboratory

October 3, 2002

## Abstract

A simplistic and amateurish design for a low-power, un-clocked ADC is exhibited. It is hoped that this may serve as a starting point for actual engineers to execute a design with the desired features. Applications to very-large-channel-count detectors, such as future generation silicon strip or pixel hodoscopes, are briefly discussed.

## 1 Stating the problem

Particle detector systems with millions to tens of millions of elements, typically silicon pixel/pad/strip detectors, are currently being designed and constructed for a variety of experiments/applications. There is a natural motivation to design and utilize ASIC’s as front-end readout electronics for such systems, for reasons of space, mass and power consumption. Accordingly, there is a need for signal measurement circuits – ie, ADC’s and TDC’s – suitable for ASIC’s, and with the lowest possible power consumption, modest wafer space demands, acceptable conversion times, and ideally minimal need for distribution of high-frequency signals.

So, we set ourselves the question: is it possible to design an ADC suitable for inclusion in an ASIC which (i) can digitize a DC level in a time less than  $< \sim 1\mu\text{sec}$ , (ii) affords at least 8-bit precision, (iii) does *not* require a clock signal, and (iv) uses little or no quiescent power and (v) minimal power per conversion? An amateur “existence proof” of such a circuit is presented in

Section 2, and thoughts on applications in a multi-mega-element detector are discussed in Section 3.

## 2 A simple plan

The design here embodies the well-known “successive approximation” approach, with the difference that here the successive iterations are carried out through a chain of modules with no feedback, as opposed to cycling in time.

The basic one-bit module is shown in Fig. 1. Its inputs are a pair of reference voltages,  $(V_{Ref,Low}, V_{Ref,High})$ , that define the lower and upper limits of the working range; and  $V_{Test}$ , the voltage to be tested. For the moment we will suppose that  $V_{Test}$  and the reference levels are not varying with time. The one-bit module tests whether  $V_{Test}$  is in the upper or lower half of the reference range in the simplest way: the level  $V_{Mid}$  in the center of the reference range is generated with a symmetric voltage divider, and  $V_{Test}$  is compared to  $V_{Mid}$  with the standard comparator. The output of the one-bit module is effectively the MSB of the binary ADC code for  $V_{Test}$  relative to the reference range.

The next bit in the binary ADC code would be the result of testing  $V_{Test}$  in the same way against a new reference range, either the pair  $(V_{Ref,Low}, V_{Mid})$  or  $(V_{Mid}, V_{Ref,High})$ , whichever range contains  $V_{Test}$ . It is easy to have the one-bit module select and generate the appropriate new pair of levels for the next comparison using the analog relays as shown in Fig. 1B. This pair can then be passed down to another one-bit module, whose output will form the next-MSB in the ADC binary code, as shown in Fig. 2. It should be clear that this chain can then be repeated to as many stages as desired, with the result that a chain of  $N$  stages produces an ADC code of  $N$  bits relative to the original voltage reference range.

In practice the length of the chain will be limited by (among other things) the noise level present at the last comparator. If the original reference range spans, say, 4.0 V, then the width of the reference range going into the last stage of an 8-bit chain will be  $(4 \text{ V})/2^7 = 31 \text{ mV}$ . It will probably be possible to make the comparator stable against noise at this level, and so we would expect an 8-bit chain ADC to be achievable without heroic effort; but longer chains may be difficult. Eight bits is straightforward, ten bits might be possible with special care, twelve is probably out of the question with this technique. It should be remembered, though, that if the physics sig-

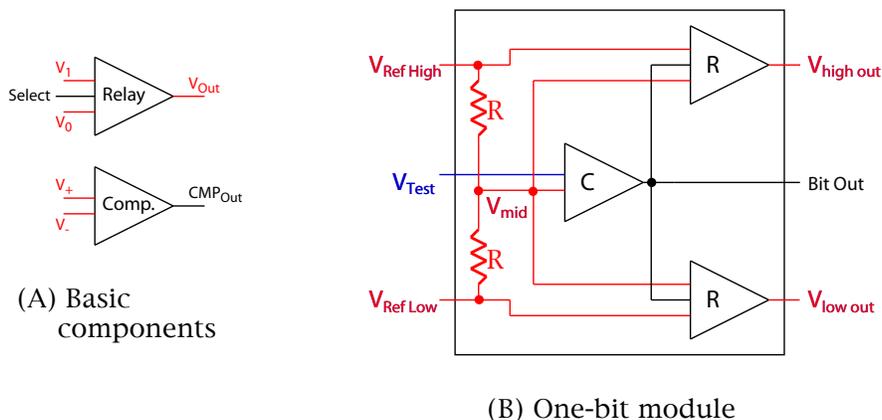


Figure 1: Basic ingredients of the chain ADC. (A) The two low-level modules: the analog relay sets its output to the upper input  $V_{Out} = V_1$  if the input Select is true, otherwise  $V_{Out} = V_0$ ; the comparator is the standard type: the output is true if  $V_+ > V_-$ , otherwise false. (B) The one-bit module using one comparator, two relays and a two-resistor voltage divider; see text. Logical signals are shown as black lines and analog levels as colored lines, here and in the figures that follow.

nal varies over a 12-bit or 14-bit *dynamic range*, but a 7- or 8-bit *precision* on any measurement would be sufficient, then we can simply build multiple parallel amplifier–ADC chains with different initial gains; the highest gain circuit which is not saturated can then be selected as a result with reasonable precision.

The design in Fig.’s 1 and 2 should not be taken too seriously – for example, a real engineer could come up with something better than the resistor divider chain to generate  $V_{Mid}$ , which is probably a waste of power and wafer space – but still exhibits many useful properties, and meets all of the design goals originally named at the outset. [Ed. – *No surprise! the fix was in.*] For example, the chain design is built from repeated instances of a small number of elements, a desirable feature for circuits built in ASICs. There is no feedback in the chain, avoiding a possible generator of noise. There is no need for an external clock, but the ADC is still fairly fast: each stage needs to have the comparator settle, followed by the relays, which could probably be made to happen in 100 nsec or less. This means that

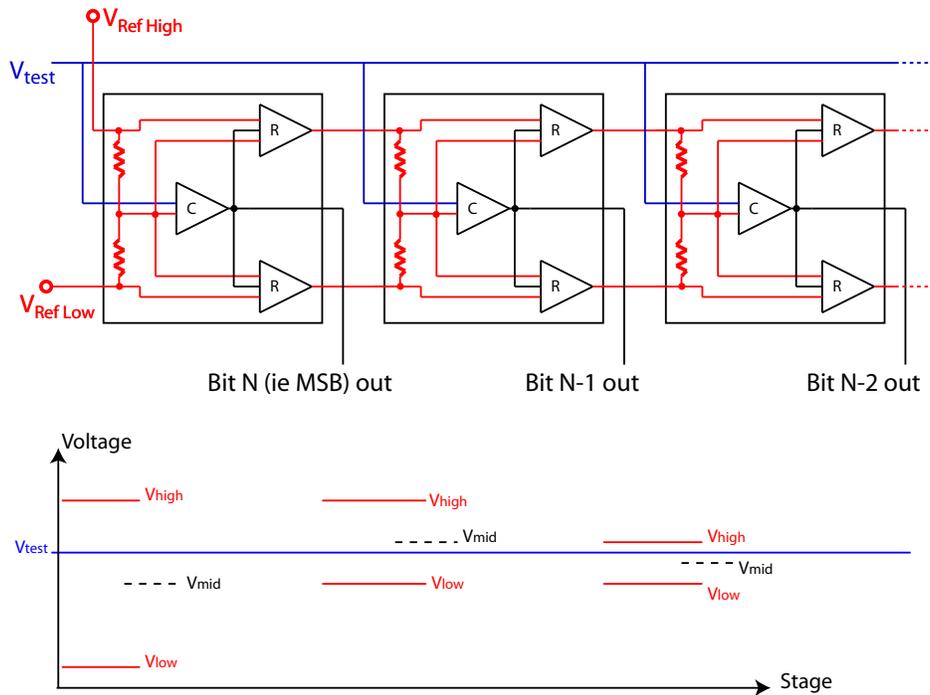


Figure 2: The top 3 bits of an  $N$ -bit chain ADC. The upper row shows the circuit, using a series of one-bit modules chained together. The inputs  $V_{Ref High}$  and  $V_{Ref Low}$  define the reference voltage range, and  $V_{Test}$  is the voltage being converted. Each stage of the chain compares its input to a pair of reference levels, and passes a new pair of references bracketing  $V_{Test}$  but with half the spread down to the next stage. The lower graph shows schematically the voltage levels used and generated at each stage of the chain for some particular value of  $V_{Test}$ ; in the case shown here the uppermost three bits of the ADC code would be 101. In principle the chain can be extended to any desired number of bits, subject only to the noise level in the comparator.

an 8-bit conversion would be complete in less than  $1 \mu\text{sec}$ , which is not stunning but should suffice for most applications. The circuit uses minimal power while quiescent, since no elements are changing state. And, after  $V_{Test}$  or the reference voltages are changed to new DC levels the conversion is accomplished with each element changing state at most once or twice, leading to very low energy cost per conversion (almost as low as can be imagined, in

fact).

### 3 Hunting the megapixel beast

The suggestion we make here is that a megapixel detector front-end would be built with one chain ADC per channel, rather than sharing a more complicated ADC among groups of channels; and that the ADC would always run live, ie self-triggered, instead of being enabled by an external trigger condition. We will first illustrate a sample per-channel circuit, then discuss the possible advantages over some other common approaches.

Figure 3 illustrates the use of the chain ADC to process the live signal from one detector element. In the quiescent state the “Peak Detect/ADC Enable” (PD/ADCE) condition is false and  $V_{Ref\ Low}$  is applied to both reference inputs to the chain, ensuring that no current flows in the voltage dividers. Generally, no elements are changing state when there is no signal input. When a signal arrives from the detector the preamp/shaper converts it to, let us suppose, a positive voltage pulse of reasonable magnitude. The capacitor tracks the voltage from the preamp/shaper (minus one diode drop) as long as it is rising, then retains its peak level once the voltage pulse starts to fall. On the rising edge the comparator is off, since the voltage at the (+) input sits one diode drop below the (-) input. Once the voltage pulse passes its peak and reaches one diode drop below the peak the comparator switches on the PD/ADCE level goes true, enabling the ADC.

The PD/ADCE signal now causes  $V_{Ref\ High}$  to be applied to the upper reference input to the chain. By this time the  $V_{Peak\ Sample}$  level acting as  $V_{Test}$  for the chain is already a DC voltage, and so the chain begins to settle directly to its final state. Each element in the chain can change state at most once or twice, so the digital output is valid after a fixed amount of propagation time following the rise of PD/ADCE. This condition can be indicated at the output simply by delaying a copy of PD/ADCE, which can be done with a parallel chain as shown. The value of the delay is not critical as long as each stage of the delay is guaranteed to be longer than the settling time for each stage of the ADC chain, which should also be straightforward to arrange.

Now, why might it be worth the effort and wafer space to equip each channel with its own ADC? and why would we want an ADC that runs live all the time, instead of operating only when an external trigger condition

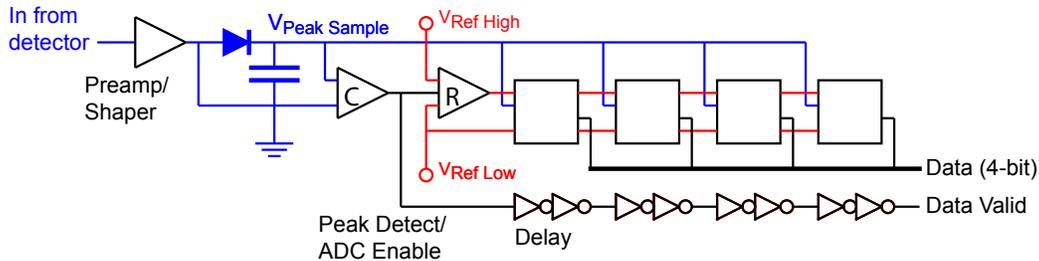


Figure 3: A simplistic “serving suggestion” for the chain ADC – in this example strung out to four bits – to process the signal from one detector element, employing the crudest imaginable peak detect and sample/hold; see text for details. The “C” and “R” elements are the same comparator and analog relay as in the previous circuits.

signals that the pulse is of physics interest? The answer depends, of course, on the alternative we are comparing against.

An alternative scheme which is currently used in many designs is to buffer the signal in some way, and then examine the previous contents of the buffer when a trigger is indicated. This could be done with analog buffering, in which case the system’s ADC’s might only have to operate very infrequently – something of a power savings, to be sure – and further it could allow multiple channels to share the same ADC, on which more attention and wafer space could then be lavished. Or the buffering could be digital, recording the output of a flash ADC into digital memory, for example. Building one 8-bit flash ADC for each of several million channels might be expensive in space and power, but even if it weren’t there is a more basic objection to buffering in either approach.

The drawback to any design with constant buffering, either analog or digital, is that there is no way to avoid paying the power cost of simply changing the address lines back and forth! In any buffered design there have to be, inescapably, at least 5-10 elements in the per-detector-element circuitry which are changing logic states at substantial frequencies, on the order of 10 MHz in a particle collider experiment. And each of these changes consumes power.

In the design illustrated here the ADC’s operate once per particle hit on an element; this might be much more than the rate of useful triggered events, but the per-hit power consumption is very low. The main point, though, is

that this rate does *not* increase as the number of detector elements goes up (for the same number of hodoscope planes). The power spent in the buffered system has two parts: the readout, which scales with the useful physics event rate, which could be quite low; and the buffering, which is independent of the particle production but scales directly as the number of detector elements. Clearly there is some channel count past which the power cost of buffering exceeds the cost of running always-live ADC's, and this is almost certainly true for counts in the millions of elements.

Another advantage of the scheme shown here is the lack of need for an external clock for the ADC circuit. Space is tight in ASIC's [*Ed. Well, duh.*] and it can only help the general noise environment if we can avoid having high-frequency signals carried across the die and into every per-detector circuit. And since the ADC's all work in parallel the overall latency is very low, especially compared to any arrangement in which an ADC would be shared among several channels.

In conclusion, we have illustrated a basic design for an unclocked ADC circuit which consumes minimal power and would be appropriate for inclusion in an ASIC. It has several advantages for per-element use in very large channel count detectors. The design could probably serve as a useful starting point to inspire an actual engineer to approve a workable version as part of a detector ASIC system.