DD
EUROPEAN ORGANIZATION FOR NUCLEAR RESEARCH

CERN LIBRARIES, GENEVA

CERN-ECP-94-25

CERN/ECP 94-25 ✓
6 December 1994

see 9504

**AN INTEGRATED CMOS 0.15 ns DIGITAL TIMING GENERATOR FOR TDC's
AND CLOCK DISTRIBUTION SYSTEMS**

J. Christiansen
CERN, Geneva, Switzerland

Abstract

This paper describes the architecture and performance of a new high resolution timing generator used as a building block for Time to Digital Converters (TDC) and clock alignment functions. The timing generator is implemented as an array of delay locked loops. This architecture enables a timing generator with sub-gate delay resolution to be implemented in a standard digital CMOS process. The TDC function is implemented by storing the state of the timing generator signals in an asynchronous pipeline buffer when a hit signal is asserted. The clock alignment function is obtained by selecting one of the timing generator signals as an output clock. The proposed timing generator has been mapped into a 1.0 μm CMOS process and a RMS error of the time taps of 48 ps has been measured with a bin size of 0.15 ns. Used as a TDC device a RMS error of 76 ps has been obtained.

A short overview of the basic principles of major TDC and timing generator architectures is given to compare the merits of the proposed scheme to other alternatives.

*Presented at the IEEE Nuclear Science Symposium, Norfolk, Virginia, USA,
1-4 November 1994.*

1. Introduction

Precise timing measurements are required within most particle detectors. This may be: Drift time of ionized tracks in a gas (drift chamber, wire chamber), Propagation time of a signal on a wire to identify the origin of the signal along the wire (Z position measurement in wire chambers) or Time Of Flight (TOF) of a particle. For drift time measurements a time resolution of the order of 1 ns is normally sufficient. For TOF detectors a RMS error down to 10 ps may be desirable.

Highly integrated multi-channel TDC's for drift time measurements have been demonstrated and are already used in several experiments [1,2,3].

Very high resolution TDC's have been constructed at the board level and single channel IC's based on the current integration principle exists. These suffers from very low integration levels and are too expensive for future experiments with many thousands of channels.

TDC devices should not only be looked at from the point of view of resolution. Dynamic range, conversion speed, calibration procedure, buffering and read-out interface must also be taken into account. For very high rate experiments like LHC an extended dynamic range may be required to act as a time stamp of the measurement (event identification). With the increasing number of channels in future experiments these functions, previously implemented with external discreet logic, must now be a part of the TDC device itself. To integrate all these functions into one single IC, at a reasonable cost, technologies supporting large amounts of digital logic and memories must be used.

The mainstream IC technology today is sub-micron CMOS which can contain very big amounts of digital functions, but it is not optimal to make high resolution TDC's. The ever increasing integration level of CMOS processes though have very promising prospects if new TDC conversion techniques are considered.

2. TDC converter principles

TDC's have traditionally been divided into two types. The current integration TDC being a high resolution analog TDC, and the digital counter TDC being low resolution. Recently alternative TDC architectures have emerged which are somewhere in between these two extremes.

2.1 Current integration

The traditional analog high resolution TDC consists of a constant current generator used to charge a capacitor, whose voltage is sampled when a hit occurs. The analog voltage is then converted to a digital value by an ADC.

- Advantages: Very good time resolution.
Low power consumption.
- Disadvantages: Limited dynamic range.
Low integration level.
Requires good analog process.
Sensitive analog design.
Limited digital functions possible.

2.2 Counter

The counter based digital TDC consists of a Gray code counter running at high speed, which value is sampled when a hit occurs.

- Advantages: Easy to design.
Large dynamic range.
Large integration level.
Enables integration of digital functions.
- Disadvantages: Limited resolution.
High speed process required.
GHz clock required.
High power consumption.

2.3 Chain of delay elements

The time resolution of the count based TDC can be improved significantly by using the gate delay as the basic time unit. The gate delay in a modern CMOS process is ~100 ps at typical process conditions. Basic CMOS gates are inverting and two gates (inverters) are therefore normally used as the delay element. The fact that CMOS processes worst case may be a factor two slower, reduces the possible time resolution to ~400 ps. CMOS processes are very temperature sensitive so frequent calibrations with a time reference is required.

The dynamic range can be extended with a coarse time counter. The merging of the fine time measurement from the delay chain and the counter is however not simple because the coarse count is related to the clocking frequency and the fine time is related to the gate delay.

- Advantages: Good time resolution.
Large integration level possible.
Simple design.
Enables integration of digital functions.
Low power consumption.
- Disadvantages: Frequent calibrations required.
Large dynamic range awkward.

2.4 Delay locked loop.

The delay chain based TDC can be improved by using a scheme where the delay elements continuously are adjusted in relation to a time reference (the clock). This is done by including the delay chain in a closed control loop as shown in Fig.1. A similar control scheme has for many years been used in Phase Locked Loops (PLL). In case of the delay chain the scheme is today well known as a Delay Locked Loop (DLL). When the delay chain is locked to the clock, the dynamic range can easily be increased by a coarse time counter.

Fig. 1 : Delay Locked Loop (DLL).

The DLL scheme is of relatively recent date but it is already used extensively in high performance microprocessors, telecom devices and TDC's.

- Advantages: Good time resolution.
Large integration level possible.
Enables integration of digital functions.
Low power consumption.
Self calibrating.
Large dynamic range.
- Disadvantages: Careful design of DLL required.

3. Improving resolution of DLL's

Despite the advantages of the DLL, the time resolution is still limited to the basic delay of two inverters in the selected process. Several different schemes to improve the resolution have therefore been proposed.

3.1 Interpolation between delay stages.

The signals propagated in the delay chain are not ideal digital signals with zero rise and fall time. A side effect of regulating the delays of the delay elements is that the signal edges are slowed down. The slopes of the edges are closely related to the delay of the delay element so the signal slope is nearly constant when the DLL is in lock. By performing an analog sum of two neighbour time taps from the delay chain a time interpolation between the two is obtained [6]. One of the problems in this scheme is to match the delay through the summing circuitry to the delays from the direct time taps.

A similar scheme is to save the analog voltages of all the delay elements when a hit occurs. The time interpolation can then be performed by a coarse quantisation of the stored voltages. An alternative approach is to use the stored voltages as inputs to a linear weighted filter which output is then converted by an ADC [7].

3.2 Delay chain difference

A very good time resolution can be obtained using the delay difference between two delay chains. The implementation is often made such that a start signal is propagated in one chain and a stop signal is propagated in the other [4]. This scheme is difficult to expand to large dynamic ranges and very long delay chains are normally required.

3.3 Array of delay locked loops

The concept of using two delay chains can be expanded to use an array of DLL's. We propose to use several DLL's of the same configuration with a small phase difference between them. The problem here is to generate the small phase shift. A delay smaller than the delay unit in the DLL's can not be made directly (otherwise that would have been used as the delay unit). It is however possible to generate with high precision a delay of one delay unit plus a fraction of a delay unit. This is done by an additional DLL, with fewer delay elements in the delay chain, locked to the same time reference as shown in Fig. 2. Because of the symmetry of such an array it can be made to look like it is phase shifted by a fraction of the delay unit only.

Fig. 2 : DLL array driven by additional DLL.

The delay steps in the locked delay chains are:

$$T_N = T_{clk}/N, T_M = T_{clk}/M$$

The time bin size equals the delay difference:

$$T_{bin} = T_M - T_N = T_{clk}/M - T_{clk}/N$$

If the required time bin size is expressed as a fraction $1/F$ of the basic delay unit in DLL-N an expression between N , M and F can be found.

$$T_{bin} = T_N/F = (T_{clk}/N)/F = T_{clk}/M - T_{clk}/N$$

$$M = N * F / (F+1)$$

Where N , F and M are integers.

An unfortunate characteristic of the DLL array is that it is not possible to make an array with 2^L (L = integer) number of time bins. This means that a coarse time counter can not simply be appended to extend the dynamic range. The coarse time count and the fine time measurement must be merged via a special encoder to get a measurement in normal binary encoding.

4. Delay locked loop

The DLL is a closed control loop where the delay of the delay chain is continuously compared to a time reference and adjusted accordingly. The behaviour of such a control loop is very dependant on the control scheme used. For the DLL it is important to obtain a very small static phase error. This is achieved by having an integrating loop filter which keeps the static phase error to zero [12]. Several configurations of the DLL have been evaluated and the best solution has been found to be the so called bang-bang configuration consisting of a phase detector and a charge pump. In this scheme a fixed amount of charge is added or removed from a filter capacitor during a complete clock cycle as illustrated in Fig. 3.

Fig. 3 : Closed loop behaviour of bang-bang configuration.

4.1 Delay element

The voltage controlled delay element can be implemented as two current starved inverters as shown in Fig. 4. Current starving on both the N and the P side requires two control voltages which may be generated by a simple current mirror.

Fig. 4 : Delay element with current mirror

Fig. 5 : Improved current mirror circuit

The simple configuration of the current starved inverter has a very non-linear delay characteristic. The loop gain of the DLL therefore varies according to where on the curve it has to lock (varies with process). This has been improved using the current mirror configuration shown in Fig. 5.

4.2 Phase detector

The phase detector measures the phase difference between the time reference and the delay chain. In a bang-bang configuration an "ideal" flip-flop can perform the required task. Real implementations of normal flip-flops though have setup and hold time requirements, and their exact sampling time of the data input is badly defined. However a symmetrical flip-flop implementation with cross-coupled RS-latches [8] have been found to accurately measure phase differences down to 20 ps in a 1 μ m CMOS process (see fig. 6).

Fig. 6 : Phase detector with cross-coupled RS-latches.

4.3 Charge pump.

The charge pump converts the phase measurement from the

phase detector into a voltage controlling the delay chain. This is done by adding/removing charge to/from a capacitor via two constant current sources. The voltage change seen on the filter capacitor when charging/discharging during one clock cycle should be as small as possible to limit the jitter in the loop. The capacitance of an on-chip capacitor is rather limited so small current levels are required. Clock feed-through and charge sharing from current switches in traditional charge pump configurations may exceed the currents from the current generators. The precision of the currents is not of great importance and the charge pump configuration shown in Fig. 7 has been found to work very well.

Fig. 7 : Charge pump.

The filter capacitor is a very sensitive node where coupling of noise sources should be prevented at all cost. A filter capacitor consisting of a large PMOS transistor implemented in a N-well has been chosen. The N-well is connected directly to ground whereby the gate capacitance works in accumulation mode and is nearly voltage independent. Connecting the N-well to ground also prevents any noise coupling from VDD.

The optimal current levels in the charge pump depend on the amount of noise coupled into the filter capacitor. This is difficult to estimate for a chip with many digital signals running at high rates. The current levels have been made programmable so they can be chosen when the chips are produced. Two levels of guard rings have been used around the charge pump circuit and filter capacitor, to keep the substrate coupled noise to an absolute minimum.

5. Timing errors in DLL array

The obtainable precision of DLL's relies on the matching of the delay elements. For a single DLL it is normally not difficult to obtain the required level of matching. When using several DLL's the matching properties of the delay elements become critical. All delay elements must have identical layout. Identical devices on the same chip though have some mis-match related to small process variations over the chip area. The smaller the transistor, the worse the matching becomes.

A mis-match sensitivity analysis of the delay cell circuit reveals that the most sensitive devices in the delay cell are the current starving transistors. These transistors are not in the direct delay path of the signal propagating through the delay element and have been implemented as non minimum length devices to improve matching.

The fact that the input signal of the delay chain equals the time reference and that the output signal is locked to the same time reference gives the following variance of the DLL caused by mis-match:

$$\sigma_{dll} = \sigma_{elem}(N/2)^{1/2}$$

In the DLL array the improved time resolution is obtained by extracting the time difference of two DLL's.

$$\sigma_{\text{diff}} = \sigma_{\text{elem}} N^{1/2}$$

This does not include the effects of the phase shifting DLL. If the delay elements in DLL-M have the same variance as the delay elements in DLL-N the total variance of the timing generator can be found to be:

$$\sigma_{\text{array}} = (\sigma_{\text{diff}}^2 + (F-1)\sigma_{\text{elem}}^2)^{1/2} = \sigma_{\text{elem}}(N + F - 1)^{1/2}$$

In the literature the mis-match of MOS devices on the same chip has been found to be in the range of 0.1% to 0.2% in strong inversion and up to 4% in weak inversion for the transistor sizes used [5]. For typical process parameters the current starving transistor will work in a region between these two extremes. Based on the simulated working point of the transistors a current deviation of 1% is taken as a realistic value.

Other sources of timing errors are jitter in the DLL and the phase error of the phase detector.

6. TDC registers

A time measurement is obtained by storing the state of the DLL array when a hit signal is asserted. After the measurement has been latched it must be decoded and transferred into a buffer waiting for read-out. Sharing of this buffer by many channels may result in a significant area saving in a TDC device. Sharing a common resource though implies that a channel may have to wait some time to write its data. New hits arriving during this time would then be rejected.

The rejection risk can be reduced dramatically by implementing several hit registers per channel. We have adopted an asynchronous pipelined buffering scheme [9,10] where no time is wasted in synchronisation (see fig. 8). The area overhead of the additional registers is very small when using dynamic storage elements. The last storage element in the pipeline is made as a normal regenerative latch to resolve possible metastable states from the asynchronous sampling of the DLL time taps.

Fig. 8 : Asynchronous pipeline buffer.

7. Evaluation chip

An evaluation chip in a 1 μm CMOS process has been produced to measure the precision of the proposed DLL array. Four DLL-N's of 25 delay elements are skewed by a DLL-M with 20 delay elements as seen in figure 9.

Fig. 9 : Block diagram of evaluation chip.

4 TDC channels with two pipeline buffer registers have been included to verify the correct function of the asynchronous buffer scheme.

A multiplexer to select any of the time taps from the array has been included in order to evaluate the timing generator as a programmable clock phase adjustment in a timing and control system for LHC [11]. A sixth DLL is provided in order to compensate for delay variations in an external clock distribution network and the interface circuits of the chip itself.

The timing variance from the mismatching of delay elements in this configuration (σ_{array}) is expected to be 42 ps. In addition a jitter component of ~ 15 ps (peak-peak = 44 ps) has been seen in the circuit simulations. If these components are added quadratic a RMS timing error of 45 ps can be expected. The array running at 65MHz has a time bin size of 154 ps which gives an intrinsic RMS error from the binning of 44 ps.

7.1 Measurements

The time distribution of the time taps and the jitter have been measured directly via the time tap multiplexer using a 6 GHz bandwidth sampling oscilloscope (Tektronix TDS 820). The time position of the 100 time taps can be seen in Fig. 10. The error is shown as a histogram in Fig. 11.

Fig. 10 : Time taps running at 65 MHz.

Fig. 11 : Error of time taps.

The RMS jitter at the last time tap (max. jitter) has been measured to be 23 ps RMS (103 ps peak-peak). This is slightly worse than expected and is probably caused by substrate noise being coupled into the filter capacitor. This indicates that the RMS error of a TDC using the DLL array should be better than $(23\text{ps}^2 + 48\text{ps}^2 + 44\text{ps}^2)^{1/2} = 69\text{ ps}$ (23 ps = jitter, 48 ps = measured error of time taps, 44 ps = binning).

Fig. 12 : TDC error histogram.

Measurements have been performed on the TDC using a high resolution timing generator (HP5359A). This timing generator has in itself a jitter of ~25 ps (105 ps peak-peak) so slightly worse than expected results should be seen. The measured error is shown as a histogram in Fig. 12. The RMS error of the TDC is 76 ps which is in good accordance with the direct measurements on the time taps and the jitter of the delay generator.

Statistical averaging techniques have also been used to cancel the jitter component from the delay generator. Unfortunately this also cancels jitter, binning and noise components in the TDC itself and should not be taken as a real performance figure of a TDC device. A RMS error of 43 ps have been measured in this way and this should be compared to the measured RMS error of the time taps (48 ps).

In the produced implementation it can be seen that the timing errors from mis-matching, binning and jitter are of the same order of magnitude and it is probably hard to reduce these significantly in the used technology.

8. Future projects

Several projects in the CERN micro electronics group are using DLL's as basic building blocks. A general purpose 32-channel 0.5 ns bin size TDC with first level trigger buffering and trigger matching, partly aimed at the muon detectors in LHC experiments, is going to be released next year. DLL's are also used in a clock extraction and programmable clock phasing ASIC for the LHC timing and control distribution system.

The use of a DLL array for a high resolution TDC for the

ALICE experiment [13] is being considered as a project for the near future. If implemented in a more modern 0.5 μm CMOS process, the RMS error can probably be reduced to ~ 25 ps. In such an implementation it is likely that a differential delay element must be used to reject power supply noise.

9. References

- [1] J. Christiansen et al., An Integrated 16-channel CMOS Time to Digital Converter, Nuclear Science Sym. 1993, pp 625 - 629.
- [2] Y. Arai et al., A CMOS four channel x 1K Time Memory LSI with 1ns/b Resolution, IEEE J. Solid-State Circuits, Vol. 27, No. 3, March 1992
- [3] S. Kleinfelder et al., The MTD132 - A new sub-nanosecond multi-hit CMOS Time to Digital Converter, IEEE Trans. on Nuclear Science, Vol. 38, No. 2, April 1991.
- [4] T. Rahkonen et al., The use of stabilized CMOS delay lines for the digitization of short time intervals. J. Solid-State Circuits, Vol. 28, No. 8, pp 887 - 894, August 1993.
- [5] M. Pelgrom et al., Matching properties of MOS transistors. IEEE J. Solid-State Circuits, Vol. 24, No. 5, pp 1433-1439, Oct. 1989.
- [6] T. Knotts et al., A 500 MHz Time Digitizer IC with 15.625ps Resolution, ISSCC 95, pp 58 - 59.
- [7] C. Neyer, R. Schulze, Developments of a fast time to analog converter, presented at ALICE TOF electronics meeting at CERN June 23 1994.
- [8] M. Johnson, E. Hudson, A Variable delay line for CPU co-processor synchronization, IEEE J. Solid-State Circuits, Vol. 23, No. 23, pp 1218 - 1223, Oct. 1988.
- [9] I. Sutherland, Micro-pipelines, Communications of the ACM, Vol. 32, No. 6, June 1989.
- [10] J. Christiansen et al., A Micro-Pipelined Zero Suppression, Trigger matching and Recalibration Integrated Circuit, Nuclear Science Sym. 1992, pp 477 - 479.
- [11] B. Taylor, Optical Timing, Trigger and Control Distribution for LHC Detectors, Nuclear Science Sym. 1993, pp. 777 - 781.
- [12] D. Wolaver, Phase Locked Loop Circuit Design, Prentice Hall.
- [13] Letter of intent for the ALICE experiment, CERN/LHCC/93-16.