Asynchronous sampling of an active non-synchronised time-to-digital converter

A. Mifsud, I. Sedgwick and N. Guerrini

Published version information

Citation: A Mifsud, I Sedgwick and N Guerrini. "Asynchronous sampling of an active non-synchronised time-to-digital converter." Electronics Letters (2019).

DOI: 10.1049/el.2019.0825

Asynchronous sampling of an active non-synchronised time-to-digital converter

A. Mifsud, I. Sedgwick and N. Guerrini

An architecture for a non-synchronised Time-to-Digital Converter (TDC) is presented. It makes use of a ring oscillator and a coarse counter for an increased dynamic range. The aim of this work is to enable multiple samples of the converter's state to be acquired asynchronously during one run without having to reset it. Such architecture is useful in applications where multiple timing values are required from a single circuit. In this scenario such architecture suffers from a timing violation when both the sampling and counter clock edges happen concurrently, thus sampling the incorrect state. A solution is proposed, and an example of this implementation is also given.

Introduction: A Time-to-Digital Converter (TDC) is used in a variety of applications such as laser range finders [1], space science instruments [2], and single photon time of flight applications [3]. Most TDCs, however, can be classified into one of two groups being: (a) those whose timing resolution is based on the minimum gate delay of the process [1], and (b) those achieving sub-gate timing resolution [2, 4].

In group (a), there are converters based on the ring oscillator architecture where an odd number of inverters is cascaded (with last output connected to first input), and made to oscillate when enabled. Shown in Fig. 1*a*, this approach uses the state of the ring oscillator to obtain the fine bits of the TDC output. A coarse counter (such as a ripple counter) is also added, thus increasing the dynamic range of the TDC. In such architecture, when the ring oscillator is enabled, its state will change depending on the gate-delay of each stage. When the oscillator overflows, the coarse counter will then increment by 1 as illustrated in Fig. 1*b*. This is repeated until the ring oscillator is disabled.

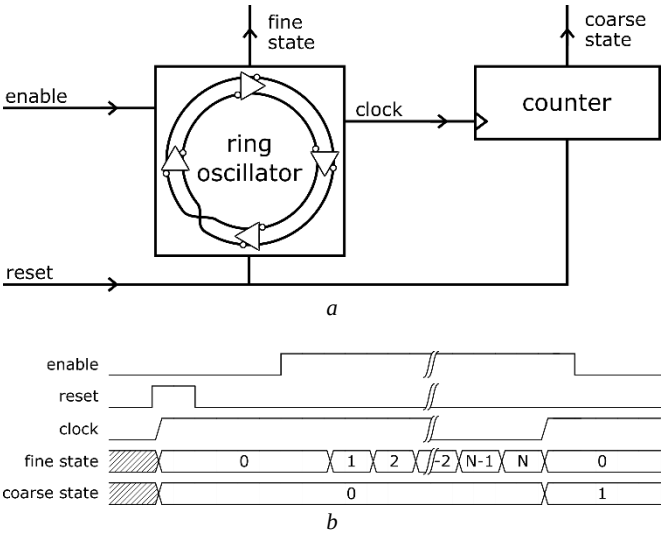

Fig. 1 *The TDC architecture considered in this work which makes use of a ring oscillator and a coarse counter for increased dynamic range.*
a Block diagram of such architecture.
b Timing diagram, where N is equal to R-1, and R is the total number of states given by the ring oscillator.

Sampling the Fine and Coarse states: Conventionally, a TDC is used to synthesise a digital value based on the time that has elapsed between two edges/signals. However, in applications where there are multiple events of interest during one TDC run, its state is sampled multiple times while it is active, just like a split time stopwatch. Such applications include mass spectrometry, where multiple ions are separated, and accelerated towards a detector. As there are multiple ions with different mass-to-charge ratios, a TDC can be triggered/sampled multiple times in one experiment.

In the architecture illustrated in Fig. 1*a*, the oscillator is asynchronous, as it is not synchronised with any other clock in the system. Thus the clock edge used to sample the states of the TDC can happen at any point in time (relative to the oscillator signals). Therefore it is important to ensure that whenever the state of the TDC is sampled, the stored state is always correct (+/- 1 oscillator state). For the ring oscillator, this is doable because it is a deterministic system, i.e. all the states are defined (even transitional states in case of a current-controlled oscillator), and can therefore be encoded correctly. The same can be said for the coarse counter. However, the interface between the ring oscillator and the coarse counter proves to be an issue when considering the timing of the sampling edge. As the counter makes use of an edge to change its state, should both edges (counter clock edge and sample edge) happen concurrently in time, or within the propagation delay of the coarse counter, the counter's stored state will be incorrect as illustrated in Fig. 2. This is due to the propagation delay of the coarse counter being non-zero. In this case, the stored state would be incorrect by R states which is a significant error (typically R is greater than 7 to give enough time to the coarse counter to settle – at least 7 states). As an example, for a timing resolution of 1ns and R equal to 8, the error would be equal to 8ns. It is therefore important to ensure that this does not happen.

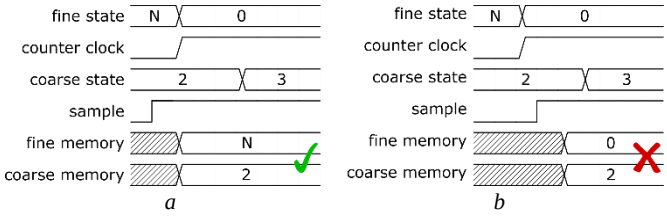

Fig. 2 *TDC operation considering two scenarios for the sampling edge.*
a In this scenario, the sampling edge occurs just before the ring oscillator overflows, leading to the correct states being stored in their corresponding memories.
b In this scenario, the sampling edge and the counter edge happen concurrently. As a result the state saved in the fine memory is incorrect by R states, where R is the total number of states given by the ring oscillator.

Solution: The solution varies depending on whether the sample signal is synchronous or asynchronous to a clock in a given system. For the former, the solution is to synchronise the TDC with a global clock. Provided that the sample edge is synchronous to this global clock, and that it does not happen during the active edge of the counter clock, then the stored state will always be correct. This solution has been implemented in [3, 5]. In these works, a delay line is used to measure the time between the edge of the sample signal and the following clock edge as depicted in Fig. 3. The coarse counter then measures the time between the first valid clock edge, and the disabling of the TDC. This is only possible because both the counter enable negative edge and the coarse counter are synchronised to the same clock. Thus, it is possible to replace disabling the TDC with sampling its state and the stored state would always be correct.

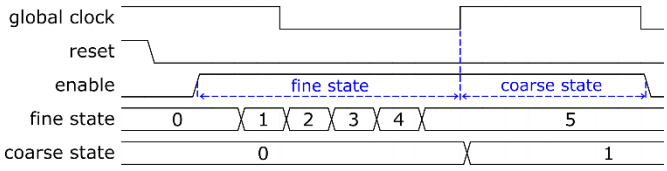

Fig. 3 *When the coarse counter is synchronised with the positive edge of the global clock, the TDC can be stopped or sampled at the negative edge of this same clock. This is only possible when the stop/sample signals are synchronous to a global clock.*

Contrarily, when the sample signal is asynchronous, synchronising the TDC does not solve the problem as the sample signal can still happen at the same time as the active edge of the coarse counter. In this case the sampling and counter edges have to be separated in time as presented in Fig. 4*a*.

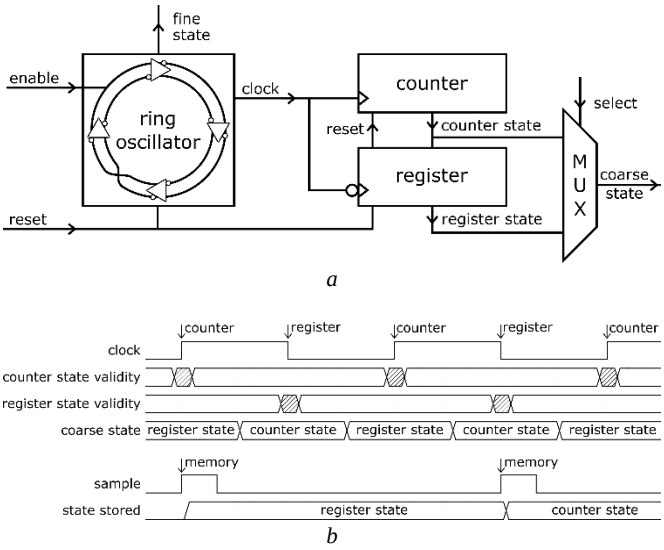

Fig. 4 *The solution for an asynchronous sample signal.*
a Block diagram, incorporating two new blocks – a register which is a memory element that is activated on the negative clock edge while the counter is activated on the positive edge. The multiplexer is then used to choose either the counter's state or the register's state.
b Timing diagram, showing the operation of the new architecture, including an example with the sample edge and one of the counter/register edges happening concurrently.

With this change in the TDC's architecture the state of the coarse counter is also sampled by a register on the opposite clock edge. In doing so, two outputs synthesised at different points in time have been generated. This makes it impossible for the sample edge to happen concurrently to both the counter and register clock edges. Thus, all that is left is a mechanism to multiplex between the counter and register outputs (hence the multiplexer). Multiplexing is done based on the location in time of the sample edge with respect to the counter and register clock edges. If the sample edge is close to the counter clock edge, the register output is stored and vice-versa as illustrated in Fig. 4*b*. This is implemented by the select signal of the multiplexer. The select signal is connected to an internal signal in the ring oscillator as one of these signals has the correct phase (compared to the coarse counter clock).

Implementation: Consider a system with a 4 stage ring oscillator, of which the first is an AND gate (connected to the enable signal), followed by 3 inverters giving a total of 8 states as depicted in Fig. 5*a*. For the counter to increment when the oscillator overflows, the clock for both the counter and the register is provided by F3 (fine bit 3, or output of stage 3), and therefore the select signal to the multiplexer is provided by F1 as presented in Fig. 5*b*. Halfway through the fine states of the ring oscillator (on the negative edge of F3), the register latches the counter state. Additionally the multiplexer selects the counter state when F1 is low, and the register state when high.

Thus whatever the timestamp of the sample signal, validity of the sampled data is always confirmed. In practice, this still needs a small change to work because if the signal arrives when the multiplexer is changing its output, there can be a timing violation due to the propagation delay of the multiplexer. This is solved by first sampling the state of the oscillator, and then, based on the value of F1 (sampled in memory), the multiplexer selects either the counter's or the register's state.

Another thing to note in the resulting sequence of the coarse state in Fig. 5*b* is that it increments when the fine state goes to 2 (and not when it overflows, i.e. 7 → 0). This is undesirable and can be solved by swapping the select and clock signals obtaining the output in Fig. 5*c*.

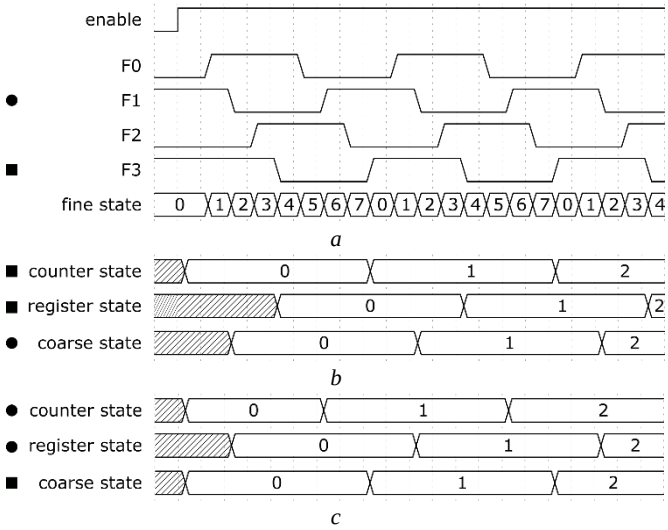

Fig. 5 *Implementation example of the proposed solution.*
a Timing diagram, showing the operation of the ring oscillator.
b Timing diagram, for the coarse state when clock is provided by F3, and select is provided by F1.
c Timing diagram, for the coarse state when clock is provided by F1, and select is provided by F3.

Conclusion: A new architecture for a TDC enabling multiple samples of its state to be acquired during one TDC run without having to reset and restart the converter itself has been presented. It ensures that the states of both the ring oscillator and the coarse counter are sampled correctly, whatever the time difference between the active coarse counter clock edge and the sampling edge. Additionally an implementation example for this solution was also provided.

A. Mifsud, I. Sedgwick and N. Guerrini (*CMOS Sensor Design Group, Science and Technology Facilities Council, Rutherford Appleton Laboratory, Harwell, OX11 0QX*)

E-mail: andrea.mifsud@stfc.ac.uk

References

1. Nissinen, I., Mantyniemi, A., Kostamovaara, J.: 'A CMOS time-to-digital converter based on a ring oscillator for a laser radar,' 29th European Solid-State Circuits Conference, Estoril, Portugal, September 2003, pp. 469-472, doi 10.1109/ESSCIRC.2003.1257174
2. Karadamoglou, K., Paschalidis, N. P., Sarris, E., Stamatopoulos, N., Kottaras, G., Paschalidis, V.: 'An 11-bit high-resolution and adjustable-range CMOS time-to-digital converter for space science instruments,' in *IEEE Journal of Solid-State Circuits*, vol. 39, no. 1, pp. 214-222, Jan. 2004, doi 10.1109/JSSC.2003.817263
3. Gersbach, M., Trimananda, R., Maruyama, Y., Fishburn, M., Stoppa, D., Richardson, J., Walker, R., Henderson, R., Charbon, E.: 'High frame-rate TCSPC-FLIM using a novel SPAD-based image sensor,' Proceedings of SPIE - The International Society for Optical Engineering, Single Photon Detectors and Imaging, San Diego, USA, August 2010, doi 10.1117/12.860769
4. Tisa, S., Lotito, A., Giudice, A., Zappa, F.: 'Monolithic time-to-digital converter with 20ps resolution,' 29th European Solid-State Circuits Conference, Estoril, Portugal, September 2003, pp. 465-468, doi 10.1109/ESSCIRC.2003.1257173
5. Gersbach, M., et al.: 'A parallel 32×32 time-to-digital converter array fabricated in a 130 nm imaging CMOS technology,' Proceedings of European Solid-State Circuits Conference, Athens, September 2009, pp. 196-199, doi 10.1109/ESSCIRC.2009.5326021