Nuclear Instruments and Methods in Physics Research A 832 (2016) 103–107

Contents lists available at ScienceDirect

Nuclear Instruments and Methods in Physics Research A

journal homepage: www.elsevier.com/locate/nima

Technical notes

Design of time interval generator based on hybrid counting method

Yuan Yao [a,b], Zhaoqi Wang [a], Houbing Lu [a,c], Lian Chen [a], Ge Jin [a,*]

[a] State Key Laboratory of Particle Detection and Electronics, University of Science and Technology of China, Hefei, Anhui 230026, China
[b] Institute of Plasma Physics, Chinese Academy of Sciences, Hefei 230031, China
[c] Hefei Electronic Engineering Institute, Hefei 230037, China

ARTICLE INFO

Article history:
Received 8 November 2015
Received in revised form
12 June 2016
Accepted 13 June 2016
Available online 16 June 2016

Keywords:
Time interval generator
Hybrid counting
Tap selector

ABSTRACT

Time Interval Generators (TIGs) are frequently used for the characterizations or timing operations of instruments in particle physics experiments. Though some "off-the-shelf" TIGs can be employed, the necessity of a custom test system or control system makes the TIGs, being implemented in a programmable device desirable. Nowadays, the feasibility of using Field Programmable Gate Arrays (FPGAs) to implement particle physics instrumentation has been validated in the design of Time-to-Digital Converters (TDCs) for precise time measurement. The FPGA-TDC technique is based on the architectures of Tapped Delay Line (TDL), whose delay cells are down to few tens of picosecond. In this case, FPGA-based TIGs with high delay step are preferable allowing the implementation of customized particle physics instrumentations and other utilities on the same FPGA device. A hybrid counting method for designing TIGs with both high resolution and wide range is presented in this paper. The combination of two different counting methods realizing an integratable TIG is described in detail. A specially designed multiplexer for tap selection is emphatically introduced. The special structure of the multiplexer is devised for minimizing the different additional delays caused by the unpredictable routings from different taps to the output. A Kintex-7 FPGA is used for the hybrid counting-based implementation of a TIG, providing a resolution up to 11 ps and an interval range up to 8 s.

1. Introduction

Time Interval Generator (TIG) produces time interval between the rising edges of pulses at different outputs. TIG is frequently used in particle physics experiments as high-precision trigger synchronizers, characterization of time-to-digital convertors (TDCs) or Automatic Test Equipment (ATE) instruments [1,2]. Time intervals can be simply generated via counting periods of a reference clock or directly by delay lines with different lengths. The counting method based on a reference clock has an advantage in dynamic range, but its delay resolution is limited by a single period of the clock. On the contrary, a delay line can provide high resolution, but the drawback is that the incoming rising edges will be dispersed while the length of the line increases. A compromise between the dynamic range and the delay resolution is necessary to satisfy the requirements of different detectors or experiments. A primal solution for solving this trade-off is the classic Nutt method [4]. Many developed solutions based on Nutt method are presented as follows. In 2012 and 2014, TIGs based on phase shifting method are respectively proposed in field programmable gate arrays (FPGA) device [1,3]. In 2015, a team implements the TIG in an integrated circuit chip, using an interpolation based on the capacitor charging method [2]. Moreover, some multi-level interpolation methods have been presented in recent years [2,5,6].

Customizability and no extra devices needed for other logical parts make the TIGs implemented on the FPGA very attractive. In this article we describe a simple but effective hybrid counting method combining with coarse and fine counting for FPGA-implementation. The coarse counting, standing for counting periods of a reference clock, is a direct way to generate wide range time interval, while the fine counting based on Tapped Delay Line (TDL) can produce precise interpolation within a single period of the reference clock. To combine the advantages of these two methods, the unequal length from taps to output, caused by the unpredictability of the routing algorithm, is inevitable in TDL-based technique. The tapping issue is dealt with a novel tap selector which makes use of a symmetrical structure to minimize the unpredictable signal delays.

2. Configuration of TDL-Based TIG

In general, a time interval can be separated into coarse and fine parts. The coarse part is constituted of several reference clock

* Corresponding author.
E-mail address: goldjin@ustc.edu.cn (G. Jin).

http://dx.doi.org/10.1016/j.nima.2016.06.063
0168-9002/

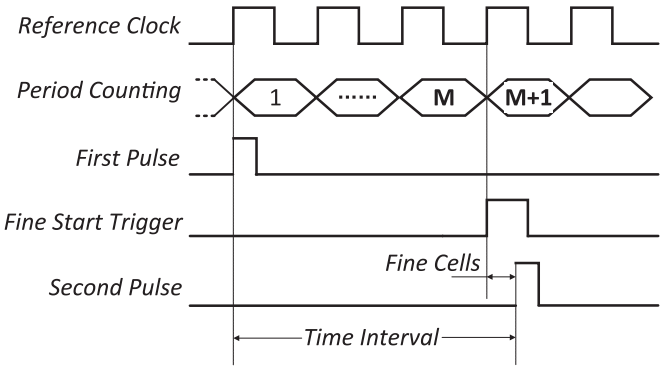

Fig. 1. The principle of interpolation-based TIG.

periods and the residue is the fine part. The coarse part can be manifested by M coarse clocks periods while the fine time can be likewise represented by N fine cells delays. Then the value of the time interval can be determined by the following formula:

$$T = M \cdot T_C + N \cdot T_F, \quad (1)$$

where T_C means the period of the coarse reference clock, T_F is the delay time of a fine cell. Variable M and N can work together to achieve adjustable time intervals. The delay time through all fine cells is longer than a single period of the reference, i.e., $N_{max} \cdot T_F \geq T_C$.

The principle of the TIG is illustrated in Fig. 1. Once a coarse counter for counting periods of the reference clock has been enabled, the First Pulse signal will be created. After $(M-1)$ periods of the reference clock, the counting module will generate a Fine Start Trigger signal to start propagation through TDL. The Second Pulse signal will be created after the trigger signal propagates through N fine cells.

A simplified diagram of the TIG is depicted in Fig. 2. As mentioned above, a wide time range is achieved by counting M cycles (T_C) of the reference clock using a binary counter. The fine part is made of a TDL with the fine delay cells and the total line delay is longer than a reference clock. The fine time is achieved by choosing a fine cell as the Fine Start Trigger signal's entrance to the TDL.

3. Implementation of FPGA-Based TIG

In comparison to other methods based on the Nutt Method, the FPGA-implementation is more convenient. It not only decreases the usage of external integrated circuits, but also reduces the complexity and power dissipation of the instruments, especially in FPGA-based particle experimental instruments which has a continuous demand for more compactness.

3.1. Delay cell of TDL in FPGA

TDL, making use of several identical delay elements, can be appropriately designed using the cascade chain, the carry chain or the phase-locked loop (PLL) in FPGA. It is worth mentioning that PLL can generate same frequency signals at different phases. The phases increase progressively with equal interval which can be utilized as the delay element. In 2010, a team performed fine tuning in routing by using the programmable delay logic implemented by look-up table (LUT) structure in FPGA [7]. Carry chains in FPGAs have been extensively used to implement time measurement due to its fixed propagation time of carry cells [8]. Different delays can also be obtained by phase modulation in PLL [9]. As a requisite structure for rapid computations in most FPGAs, carry chain, consisting of delay elements of tens of picosecond [10], is chosen as the fine cell in our TIG.

The carry chain can be automatically created via the compiler when IP core, e.g., adder or counter, is implementing. Moreover, the chain cells are automatically linked in the same column by the compiler. Using low-level primitives can also directly invoke the carry chain. For example, in some Altera devices, CARRY_SUM primitive indicates that the wires should be placed on the fast carry chain logic. Our TIG is realized in a Xilinx Kintex-7 device, while the primitive for creating carry cells is CARRY4.

3.2. Tap selector

A typical idea to design the tap selector is to directly feed all taps into a multiplexer [11] or to utilize the wired-and or wired-or structure. Nevertheless, multiplexers are realized by Look-Up αe (LUT) in FPGA and the wired-structure does not exist in most FPGAs. The LUT-implemented low-level routings is unpredictable and will lead to unequal delays from different taps to the output of the tap selector. The skew is of the same order of the finest delay step of a carry chain (more than tens of picoseconds). Actually, the implementation of routings with equal length from different taps to the output is challenging not only in FPGAs, but also in ASICs.

For solving the problem caused by asymmetric routings of the multiplexer, a novel structure of the tap selector is devised. The Global Clock (GCLK) resources, provided by most FPGAs and mostly used for high fan-out and low-skew operations, can satisfy the crucial requirements of the tap selector. In Kintex-7 devices, a dedicated multiplexer (Carry-MUX) determines whether the source of signal spreading in the chain is the Fine Start Trigger signal or the port COUT of previous Carry Cell. As shown in Fig. 3, the Fine Start Trigger signal is fanned out by BUFG primitive to 50 Carry Cells. The tap selector assigns the source of a specified Carry-

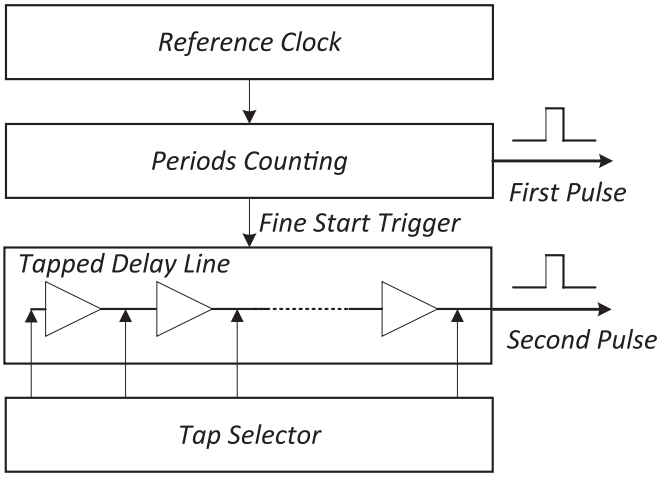

Fig. 2. Simplified diagram of TDL-based TIG.

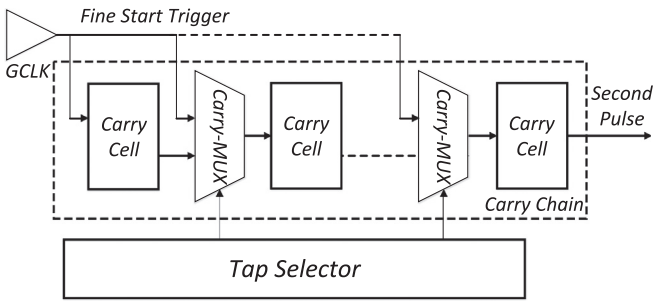

Fig. 3. Structure of TDL-based interpolation in FPGA.