

# A simple Field Programmable Gate Array (FPGA) based high precision low-jitter delay generator

Cite as: Rev. Sci. Instrum. 92, 024701 (2021); doi: 10.1063/5.0030341

Submitted: 23 September 2020 • Accepted: 2 January 2021 •

Published Online: 1 February 2021



View Online



Export Citation



CrossMark

Zeyang Chen,<sup>1</sup> Xiaohan Wang,<sup>1</sup> Ziwen Zhou,<sup>1</sup> Ramiro Moro,<sup>1</sup>  and Lei Ma<sup>1,2,a)</sup> 

## AFFILIATIONS

<sup>1</sup>Tianjin International Center for Nanoparticles and Nanosystems, Tianjin University, 92 Weijin Road, Nankai District, Tianjin 300072, China

<sup>2</sup>State Key Laboratory of Precision Measuring Technology and Instruments, Tianjin University, 92 Weijin Road, Nankai District, Tianjin 300072, China

<sup>a)</sup>Author to whom correspondence should be addressed: lei.ma@tju.edu.cn

## ABSTRACT

Pulse delay generators are ubiquitous in laboratories to coordinate and control the timing between different devices in applications that include lasers, mass spectrometers, and other scientific instruments. The most important required characteristics are precision, to control time exactly, and low-jitter, to minimize uncertainty in experiments. Here, we introduce a new design of a high precision and low-jitter digital delay generator based on a Field Programmable Gate Array (FPGA). The final delay is composed of steps of 4.2 ns (coarse delay) with fine steps of 16 ps (fine delay). The coarse delay is generated by a 240 MHz pulse sequence from the FPGA with a 50 MHz clock. An embedded time-to-digital conversion unit is used to measure the interval between the external trigger and the clock signal, which, together with an integrated delay generator, is used to realize the fine delay. Jitter compensation is achieved through a measurement-and-feedback module. A computer interface is designed to control the system through a Nios II processor. Measurements confirm a time resolution of  $16 \pm 2$  ps with jitter below  $450 \pm 20$  ps (at  $24^\circ\text{C}$ ) with a maximum delay of 1 s. The whole system is simple in structure and low in cost.

Published under license by AIP Publishing. <https://doi.org/10.1063/5.0030341>

## I. INTRODUCTION

A digital delay generator (DDG) is a device used to tune the timing between different devices or events, which is widely used in laser systems,<sup>1,2</sup> radar echo signal simulating systems,<sup>3,4</sup> Positron Emission Tomography (PET),<sup>5</sup> Automatic Test Equipment (ATE),<sup>6</sup> time-of-flight mass spectrometers,<sup>7-9</sup> and other wide spectra of scientific instruments.

The operational principle of a DDG is when an external signal triggers the device, an output signal is generated after a digitally set delay. It has been intensively explored in the past several decades, reaching sub-picosecond resolution and jitter.<sup>6,10-13</sup> One recently reported architecture, based on the subset sum delay line (SSDL), achieved a delay accuracy of 76 ps and a dynamic range of 32 ns.<sup>14</sup> Another design, based on the carry chain of Stratix III Field Programmable Gate Array (FPGA), demonstrated a device with a delay accuracy of 38.6 ps.<sup>15</sup> However, with the development of modern experimental requirements, there is urge of a DDG with even higher resolution of picosecond scale, larger time span of seconds, having multiple channels, low cost, and low jitter.<sup>6</sup> The common ways to realize this are the counter method<sup>16</sup> and the delay chip

method.<sup>17</sup> The counter method can provide a wide time range, but its accuracy is limited by the sampling rate, while the delay chip method offers much higher precision; however, it has a far limited range. Aiming for those aforementioned needs, taking advantage of both technologies, the design presented here achieves several tens of picosecond resolution, 1 s of delay range, and more than ten full channels at extremely low cost.

## II. DESIGN

The FPGA chip used here is the EP4CE6E22C8. It has an embedded time-to-digital conversion (TDC) module, two pulse generators, and a Nios II processor with the function of jitter compensation as shown inside the dashed-dotted line in Fig. 1. The total time delay of the DDG is composed of a coarse delay and a fine delay. The coarse delay is realized by using the counter method whose resolution is determined by the sampling rate. The 240 MHz clock signal is generated by multiplication of an external 50 MHz from a crystal oscillator, which is the maximum frequency that the system can run. To generate the fine delay, the coarse delay step is subdivided by the 8-bit control word of the integrated delay generator AD9501, giving

a resolution of  $4.2 \text{ ns}/255 = 16.5 \text{ ps}$ .<sup>18</sup> The block diagram of the system is shown in Fig. 1.

After the external trigger enters the system, it splits into two channels: the coarse delay is realized by one channel and the jitter is measured in the other by the TDC. The TDC measures the interval between the trigger and the clock signal and then outputs to the delay compensation module. Through the software interface, the personal computer (PC) sends the user delay setting to a Nios II processor, which will be allotted into coarse delay and fine delay and implemented via a CP2102<sup>24</sup> serial port bridge chip. The processor sends the delay information to the jitter compensation module through the parallel input output (PIO) IP core, which is responsible for processing the delay setting and the measured interval, and further yields the compensated delay consisting of coarse delay and fine delay. The compensated coarse delay will be directly sent to its corresponding module, while the compensated fine delay value is converted into 8 bits, and sent to the AD9501 to synthesize the fine delay. Finally, the delayed signal is output by the AD9501 chip.

The AD9501 works by comparing the output voltage of a ramp generator against a set DAC value, when they are equal, the delay pulse will be created. Initially, the ramping voltage is set to 0 (reset) by pulling up pin 4. Therefore, the coarse delay module outputs two pulses to the AD9501: one for triggering the fine delay and the other for resetting the ramp generator. The interval setting between these two pulses controls the width of the delay signal. The timing diagram of the module is shown in Fig. 2.

The highest delay resolution in the counter method is determined by the sampling clock frequency. In this design, the 240 MHz clock yields 4.2 ns unit of coarse delay. The time interval between the external trigger and the sampling clock (see Fig. 2) is randomly

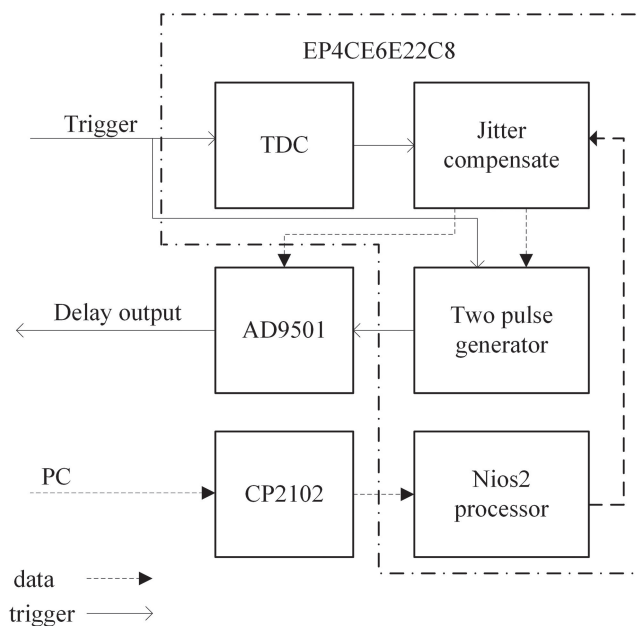
distributed from 0 to 1 clock period, which would lead to a jitter of 4.2 ns if no compensation was applied.

The jitter measurement and compensation module measures the time interval between the trigger and the rising edge of the next clock signal and eliminates the jitter in the output signal. The most commonly used technique for time interval measurement includes current integration,<sup>19–21</sup> counter,<sup>22,23</sup> and time interpolation methods. This last method is realized by inserting much shorter time delay units into the interval and then counting the number of these units until it fills up the full span of the interval. This can be easily implemented in a FPGA and practically not only can improve system integration but also can reduce its cost. The construction of an interpolation circuit is realized simply by using the adder carry chain inside the FPGA<sup>26–30</sup> shown in Fig. 3.

The adder has two n-bit inputs. One is set to all 1s, while the other has all the high (n – 1) bits at 0, leaving only the lowest bit for the external trigger signal. When the external trigger signal changes to 1, the addition operation is conducted from the lowest significant digit up. Because the lowest adder has its two inputs at 1, a carrier is generated and transferred to the next arithmetic unit for operation, where the inputs are 0 and 1, and a carrier is generated again, and so on. Hence, the pulse signal is transferred upwards in the add carry chain. When the clock signal arrives, the latch of the arithmetic unit locks the result of the adder. A value of 0 in an arithmetic unit indicates that the carrier was generated, and conversely, a value of 1 indicates that the signal did not reach that unit yet. Therefore, the latch locks a column of binary numbers “0000111111,” where the position of the 0–1 jump indicates the location of the transmitted signal. Since the time span of each carrier unit is known, the number of zeros in the binary column can be converted to time, which realizes the measurement of the interval between the trigger signal and the clock signal.<sup>31</sup>

The interval measured by the carry-chain needs to be compensated in the total delay by subtracting it from the set delay. First, the delay will be separated into two parts: one is the maximum integer number of coarse delay units that can be fitted in the interval and the other is the remainder, which is less than a coarse delay unit. These two parts are called coarse delay part and the fine delay part, respectively. The realization of the coarse delay is by using the counter method, which is the integer number of fixed unit time of 4.2 ns, as mentioned above; the fine delay is realized by controlling an AD9501 chip through its 8-bit control word and reaches a resolution of 4.2 ns/255. The average length of each arithmetic unit is 0.073 ns,<sup>25</sup> so the time precision of the AD9501 is better than the carry-chain unit. For example, if the measured interval is 0.073 ns,  $0.073 \text{ ns}/0.016 \text{ ns} \approx 5$ , so we can compensate the 0.073 ns with 5 units of fine delay that express that interval. After analyzing the delay requirements, the operation is as follows:

- (1) When the coarse delay is set to 0, only the carry-chain would be used to compensate the coarse delay jitter. The pulse delay will only come from the AD9501 with its fine delay.
- (2) When the coarse delay is nonzero, the coarse delay counter kicks-in. The carry-chain will measure the interval between trigger and clock signal, and there would be the following cases:
  - (i) If the measured interval “jitter” is less than or equal to the fine delay part, then the output coarse delay part will



**FIG. 1.** Diagram of the system architecture. The solid line represents the trigger signal path, and the dotted line represents the data path. Function blocks inside the dotted-dashed line are the parts of FPGA chip.