Development of the Electronic Circuit in High Frequency SLR Based on FPGA

Chong CHEN[1,2], Cunbo FAN[2], Zhenwei LI[2], Xue DONG[2,3], You ZHAO[1,2]

[1]National Astronomical Observatories, CAS, 100012, China
[2]Changchun Observatory, NAOC, CAS, 130117, China
[3]Graduate University of Chinese Academy of Sciences, Beijing

Abstract

Increase of the laser firing frequency will significant improve the performance of Satellite Laser Ranging (SLR) system. To meet the requirement of high frequency SLR, an implementation of control circuit in single FPGA chip was designed and developed in this paper. SOPC (System On Programmable Chip) system was proposed to solve these problems. To realize the system, a control circuit custom component was designed and simulated. Then, the component was integrated into a SOPC system. Cooperated with software, the circuit has the ability to control the SLR system running at high frequency. Finally, the system was simulated in the Quartus software and NIOS IDE provided by Altera and implemented in an Altera EP1S10 development kit.

Key words: High Frequency Satellite Laser Ranging, Rang Gate, Firing Control

1. Introduction

The running frequency of tradition satellite laser ranging (SLR) system was limited by the control circuit and other components of system; the frequency was determined by the following equation: F=C/2R. Here the constant C is the speed of light and R is the distance between satellite and station. We can find out that the running frequency will suffer a significant decrease when observing high orbit targets. When it concerned to synchronous satellite, the system only can complete 4.17 ranging per second. Also, with the application of event timer, some station keep its system running at the constant frequency of 10 Hz in spite of the target distance, but the quantity of date is still low. Increasing the running frequency of the system is an efficient method to solve this problem. The trend of new generation SLR system is developing the system able to running at the frequency of Kilo-Hz, called KHz SLR system. Not only increase the quantity of observation data, the benefits of KHz SLR also include the improvement of Normal Point (NP) precision. To increase the running frequency of the SLR system, several improvements need to be conduct to the current system, and one of the most important parts of these improvements is the redesign of control circuit. The function of the control circuit in SLR system is to generate the Rang Gate (RG) signal for time filter and the fire signal for laser. Time filter is a specific filter widely used in SLR observation, which means to generate a RG signal to control the photon receiver (SPAD in our station). With the application of time filter, the SPAD will only active just several nano-second before the return photon arrive, the noise in other time will not influence the SPAD. For the reason of complexity, the currant design method is no longer useful for new generation control circuit. In this paper, we present a method to integrate KHz SLR control circuit into single FPGA chip. Based on the introduction of SLR system in Changchun Observatory, we discussed and described the details about the design method and experiment

result of this implementation, which is meaningful for Changchun Observatory to update its system available to conduct KHz observation.

2. Main problem faced by the control circuit of high frequency SLR

Different from currant SLR control circuit, to meet the requirement of high frequency SLR, the control circuit should solve the following two problems:
(1). Overlap of start and return impulse: between the start and return of one impulse, there are amounts of new impulse on its flight, which means that the flight time of these impulses was overlapped at some level. Each impulse needs a RG signal, the number of RG data buffered in the circuit can be calculate trough the following equation: N= 2fR/C. Here f is the frequency of laser, R is the range between satellite and station, C is the speed of light. Above equation indicates that, if the system running at the frequency of 2KHZ, then the number overlapped impulse can reach to more then 480 when observer synchronous satellite. If we use counters, as currant control circuit did, to generate the RGs, more then 480 counters will be implemented, which is unpractical. In this paper, we use FIFO to buffer the return time of each impulse and comparer to generate the RG signal.
(2). Backscatter: laser impulse will be scattered by the atmosphere, some photons will be reflected back and receive by the telescope, and this phenomenon was named backscatter. Normally, the backscatter will last about 100 us after the laser pulse was send out. When backscatter happens, the noise will disable the receive equipment (SPAD), which will greatly decrease the efficiency of the SLR system. To solve this problem, the control circuit should generate the fire signal appropriately to avoid the return photons.

3. The hardware block-figure and the software flowchart of the control circuit

Basically, there are two schemes for implement the circuit. The fist scheme is design a circuit without microcontroller, all the prediction and control instruction is operated by the host computer. The advantage of this method is that the hardware design will be much easier. The drawback of this scheme is that we should transplant the whole software to a real-time operation system from the currant windows platform, because the time remained for computer to complete the prediction is limited. Take a 2 KHz system for example, in the worst situation, the time between two laser pulses is 400 us, in this short time computer should not only give the prediction result but also complete the operation of system control and ranging data process/display. The scheme employed in this paper for the circuit implementation is to construct a embed system, which includes a micro processor, to execute the prediction operation. This scheme releases the burden of host computer. Therefore, the circuit can access specific resource to complete the calculation with real-time processing requirement. The currant software can be used for high frequency observation with little modify.

Figure 1 is the block figure of whole control circuit system, the host computer send the Ephemeris to the embedded system in FPGA chip, because the transmission will be completed before the observation, there are no strict requirement for the speed, we can transmit it through RS-232 serial communication. A SOPC system was integrated into the FPGA chip, which was composited by several IP components provided by Altera, and a custom component designed by us. All this components was bridged trough Avalon-bus.

For the reason that FPGA chip will loss its configuration information within power off, we use a FLASH ROM to storage the data to configure the device. For the limitation of inner

storage space in FPGA, we add SDRAM as system memory. Also, we design a custom component using VHDL, it communicate with other components through Avalon-bus. The components input signal includes 10 MHz, PPS and start pulse; most of the control signals act as the custom components output. Because the voltage level of TTL and CMOS was incompatible with each other, we need the voltage level transformation chip to guarantee the signal to be correctly identified.

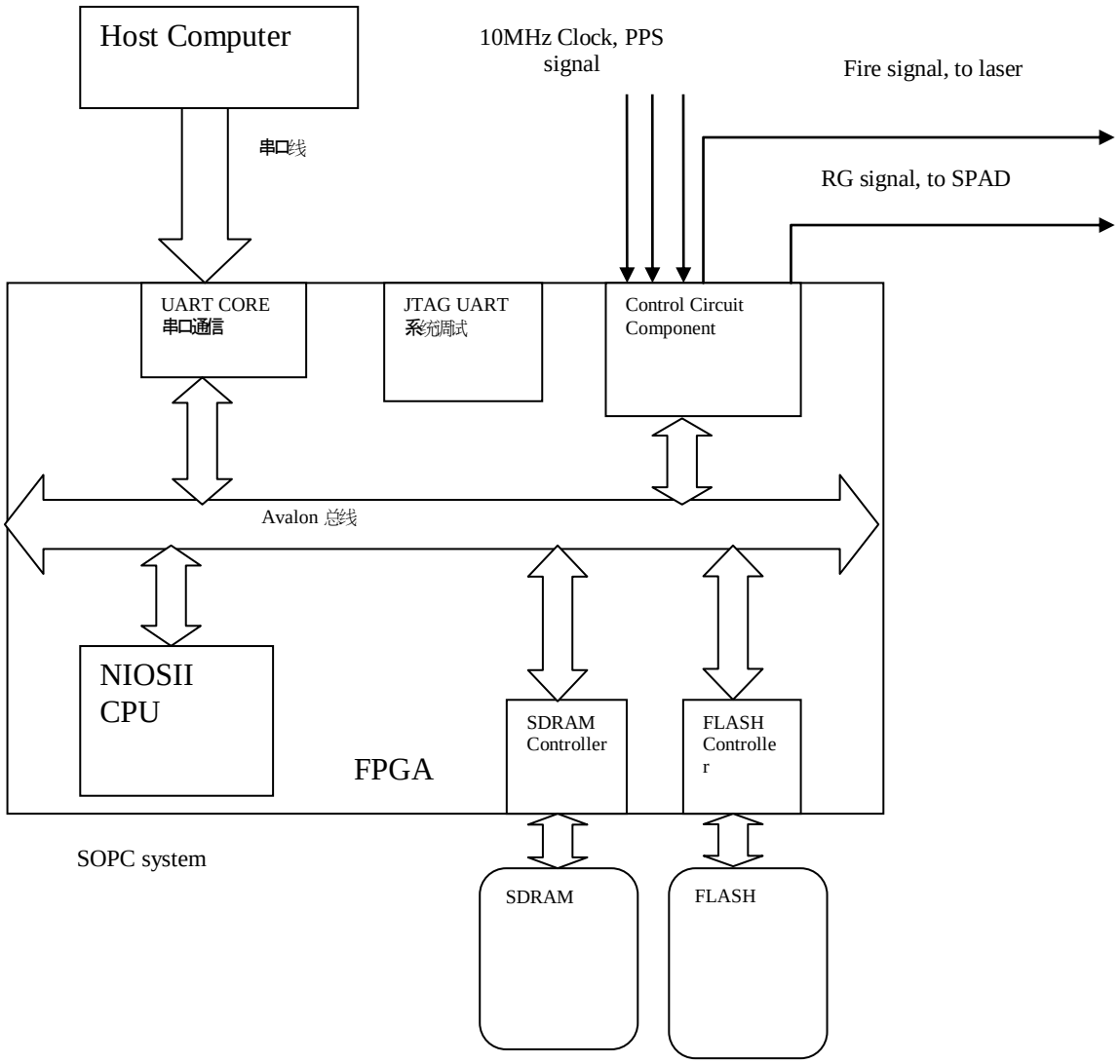

Figure 1. Hardware block diagram

3.1 Flow-Chart of software

Firstly, for the reason that interpolation is used to calculate the return time, we should consider the time sync between UTC (which was used in ephemeris) and the inner clock of control circuit. So, after complete the transmission of ephemeris, the host computer read the currant second of UTC from GPS and transmits it to the circuit, then the circuit begins to count in the next second.

In host computer, the software operation related to the control circuit is very simple; firstly the computer transmits the ephemeris to the circuit, and then transmits the currant second data.

Figure 2 presents the flow-chart of control circuit, it can be described as follow: firstly read the RS-232 UART port, storage the ephemeris to the inner and outside memory, after the transmission is completed, active the "running registration" of custom components, then the components begin to fire the laser, at the same time, the NIOS CPU begin to check the statue bit of this component to determine whether the start pulse occurred, if yes, CPU will read the registered happen time of start pulse, calculate the return time and write it to the component. If the observer want to change the target, reset the CPU, system will waiting for new transmission of ephemeris.

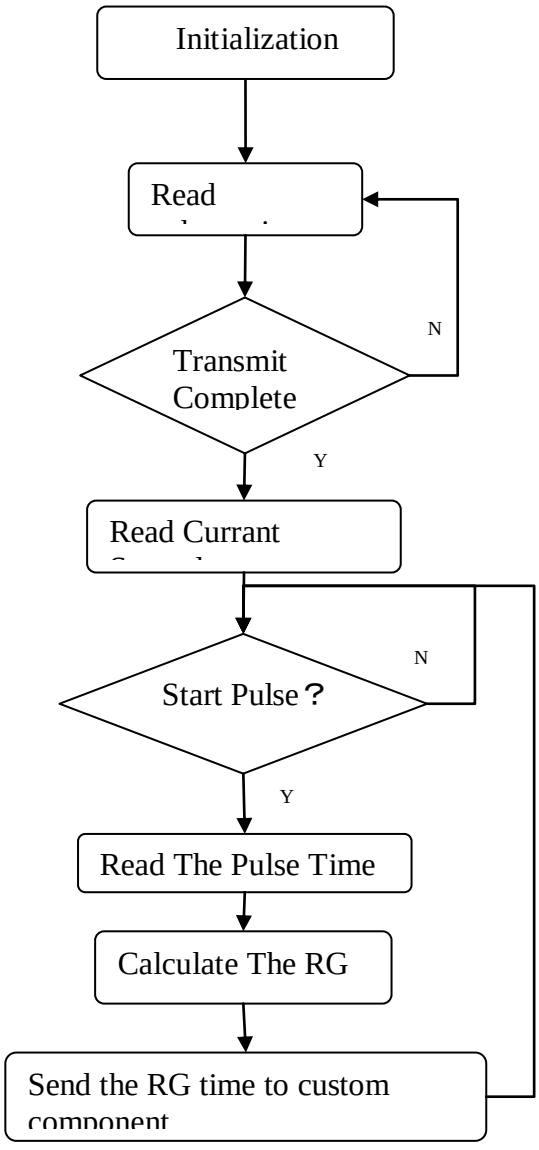

Figure 2. Software in embed system

4. Design, simulation and experiments of custom components

4.1. Fire module

Suppose the system running at the frequency of 1 KHz, we divide the 10 MHz clock with the parameter of 10,000 (we can change the divide parameter to let the system running at any frequency), get the 1 KHz signal for fire. But if we send this signal directly to laser, it is possible that the black scatter region will overlap with the return photons.

From the ephemeris, we know the time when the return photons will arrive, base on this assumption, there is a simple method to avoid backscatter: we generate two fire signals, on without any delay; the other was delayed with 100 us. Figure 3 describes the principle of this method, from the figure, we can see that every fire signal will cause a "backscatter region" of about 100 us, to avoid backscatter, we choose only one pulse as output signal, depend on whether the return photons will be interfered by the backscatter noise. To be more practical, if the return will happens in the fist pulse's backscatter region, pulse two will be send out, in the contrast condition, the first pulse will be send out.

This method requires a state-signal to indicate whether the return photons will be interfered by the backscatter. The circuit in the RG module will generate it and more details will be discussed in the following section of this paper. Figure 4 presents the block diagram of fire module; the function of each pin can be described as follow:

Clk: 10 MHz input clock.

Shift: Control signal, from RG module, indicate whether the return is in the backscatter region of pulse 1.
CLR: reset.
Main_pulse: fire signal.

The delay unit we implemented is a synchronize state-machine. After trig by a rise-edge, the counter of the unit will keep counting until the number reach a constant, and then the unit will output a pulse with 1-clock cycle width.

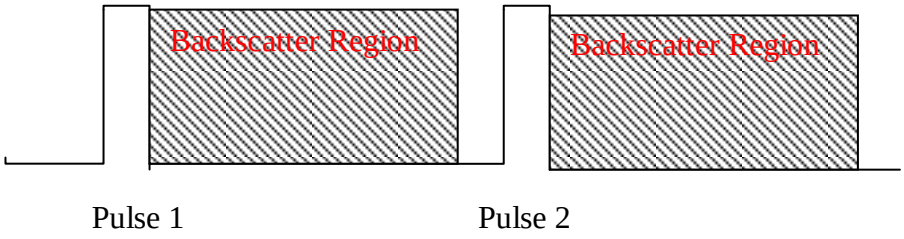

Figure 3. The principle of Avoid backscatter

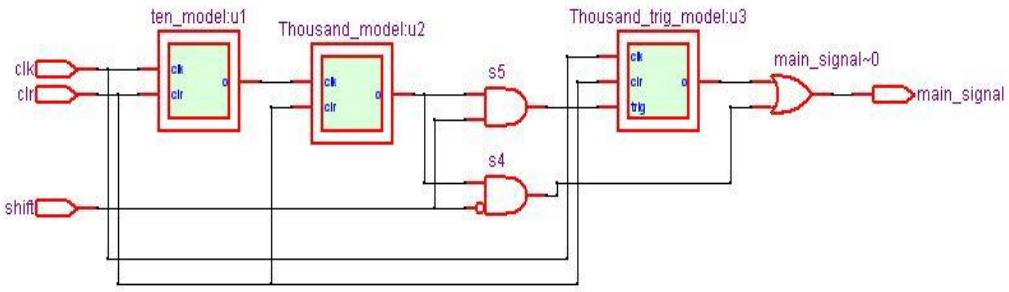

Figure 4. The block diagram of fire module

4.2. RG module

Figure 5 is the block diagram of RG module, to describe it more clearly; we simplified the diagram and neglect several minor signals. The "counter" unit is a multi-usage unit, it provide the system clock, meanwhile register the start pulse time. When the laser firing, the start pulse time was registered by "counter" and read by NIOSII CPU, after the return time is calculated, it will be send to FIFO. Contemporarily the FIFO already buffered numerous of RG time, the output of FIFO is the RG time for next return; this data will be compared with currant time. If the two numbers are equal, the compare will send out a RG signal, the signal also feedback to FIFO to update its output.

The output of FIFO also subscribed with the decimal part of the counter output, the result will be compared with the constant derived from the following equation: N= ζ /T. Here ζ is the "backscatter sustain time" and T is the period of clock. If R<N, the shift signal will be active.

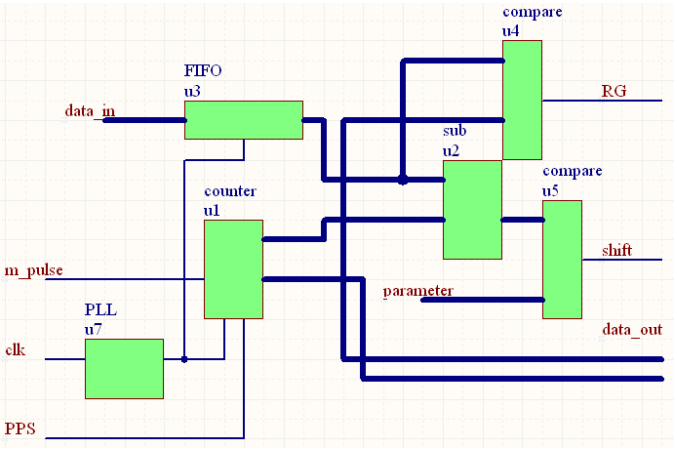

Figure 5. The block diagram of RG module

4.3 Implementation and simulation

All the above design was described by VHDL, synthesis and routed by Quartus II software provided by Altera cooperation, the target device is Stratix 1s10 FPGA. After complete the design flow, we simulated the above design using Quartus II simulator. Figure 6 and figure 7 is the simulation result.

The input waveform file includes input signal m_pulse, PPS (second) and CLK (clock for SOPC system), clk1 (10 MHz external clock), at first active period of wr, set the whole bits of control word to '1', enable the component to running. In every follow active period of wr, write RG time in the input port, in this simulation, we write 4 RG time, 1 ms, 2 ms, 3 ms and 4 ms. After a while, we assign several active periods to the rd signal, In more then one periods, the adr input is '0' (output the decimal part of start pulse time), and in other periods is '1' (output the remain part of the start pulse time). Chipselect signal is always high. Figure 6 is beginning of the simulation, we can see that the m_pulse signal was registered, and the rddate output is correct.

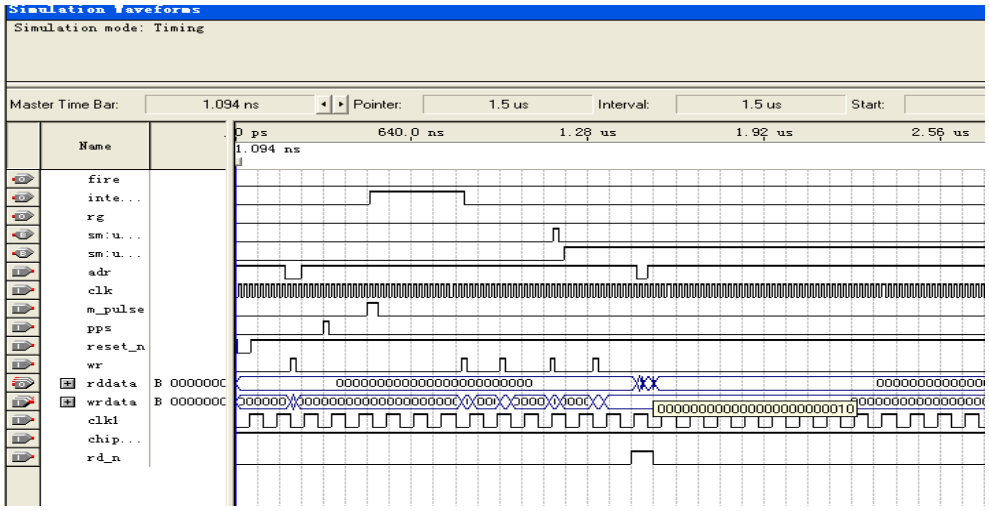

Figure 6. The beginning of simulation

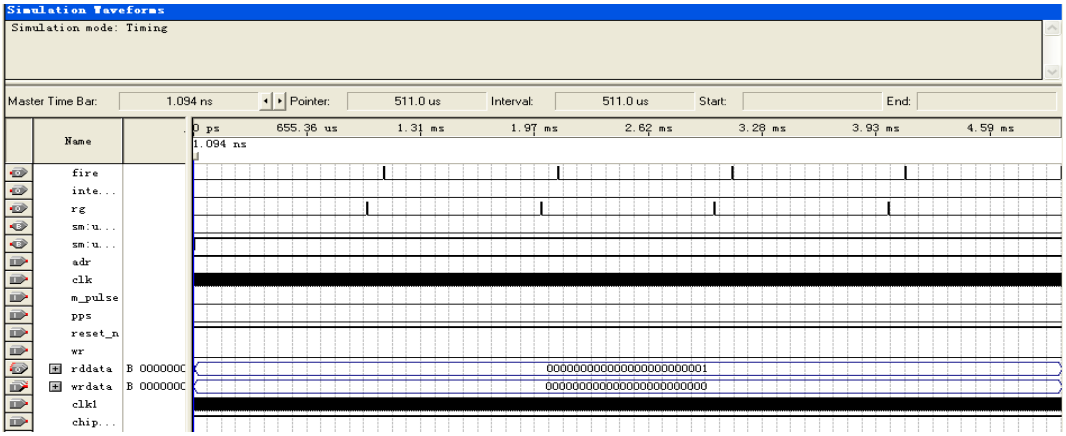

Figure 7. The whole simulation result in the 5 ms time scale.

Figure 7 is the whole simulation result in the 5 ms time scale. We can find out that the RG signal was generated correctly at 1 ms, 2 ms, 3 ms and 4 ms after PPS signal was active, all the fire signal avoid the backscatter region.

The above simulation only verify the performance of our design within 5 ms. With the application of no-sign subtracter, when the system time "cross the second", the circuit can still calculate the number of clock periods between currant time and the predicted arrive time of return photons, then generate the "sift" signal correctly.

5. Hardware implementation and precision

We realize the system in a 1S10 development board from Altera, the documents about this board can be find in Altera website. The board has a RS-232 UART connector and two 14-pin headers for 3.3/5.0 volt prototype connector. So we can build a daughter card with the connector to SPAD, laser and 10 MHz clock in one side and the 14-pin slot in the other side, then insert this card upon the develop board. The SPAD, laser and 10 MHz clock will link to FPGA board with this daughter card. The host PC and FPGA board will be connected with a serial communication cable.

6. Summary

Frequency increase will significantly improve the performance of SLR system, we design a new control circuit which can running at KHz frequency. By integrate a SOPC system in signal FPGA; the circuit can complete the prediction independently after the ephemeris was successfully transmitted. To guarantee the efficiency, the circuit can shift the laser-fire signal automatically to avoid backscatter. The circuit is experimented in a 1S10 develop board, the future work is to implement the circuit in a PCB board designed by ourselves and use it to enable the SLR system in Changchun Observatory running at KHz frequency.

References

1. Kirchner, G., *Rang Gate Generator at GRAZ*, Proceedings of the kHz SLR Meeting, Graz, Austria, October, 2004
2. Altera Corporation., *NiosII Software Development Handbook*.
3. Altera Corporation., *Stratix Device Handbook.*
4. Altera Corporation., *Quartus Handbook Volume 4, SOPC builder*.