Design of Extensible Data Acquisition System Based on the Zynq Platform

Jiawu Fan [a], Qingping Wang [b], Deping Zhang [c] and Naichang Yuan [d]

School of National University of Defense Technology, Changsha 410000, China

[a] FANjiawu217@163.com, [b]995012251@qq.com, [c]1025012261@qq.com, [d]1340596440@ccc.com

Keywords: extensible, data acquisition system, ADS8568, Zynq Platform.

Abstract. Currently, Data acquisition system is widely used in rader data fields. Demands are increasing on the rader data acquisition system with the development of rader technology. Frequently system upgrading not only increase the design time and cost, but also increase the burden on the designers, and sometimes even minor changes may cause the redesign of the entire system. To solve the implementation bottleneck problems, we used Xilinx's new Vivado Design Suite to design extensible data acquisition system with a full use of the system integration capabilities of "All Programmable" devices based on the Zynq platform [1].

1. Introduction

Data acquisition systems are widely used in the field of radar signal acquisition. Currently, with the development of modern electronic technology, particularly the development of FPGA and DSP technology, we need to improve and upgrade the data acquisition system constantly in order to respond to the needs of practical application. However, currently designers face many problems during development. For example, the problems exist in the aspect of system integration are how to reuse IP, how to integrate algorithms and RTL-level IP and how to mix DSP, connectivity, logic, and so on. As for realization, the problems exist in chip planning and layering, multi-field and a lot of physical optimization, multiple "design" and "timing" convergence, knock-on effects of post-engineering change orders (ECO) and design changes, and so on. Especially after ECO, when we change a local circuit, it may have "the butterfly effect" that will change the entire design timing, or make it need to re-timing convergence.

To solve the integration and implementation bottleneck problems, Zynq extensible processing platform uses the latest series of 28nm programmable technology which is used in the new generation Xilinx FPGA (Artix-7 and KINTEX-7 FPGA). Programmable logic can be configured by the user, and connected together through the "interconnection" module, which provides user-defined arbitrary logic function to extend performance and functionality of the processing system.

The Zynq platform architecture enables implementation of custom logic in the PL and custom software in the PS. It allows for the realization of unique and differentiated system functions. The integration of the PS with the PL allows levels of performance that two-chip solutions (e.g., an ASSP with an FPGA) cannot match due to their limited I/O bandwidth, latency, and power budgets [2]. A high level block diagram is shown in Figure 1.

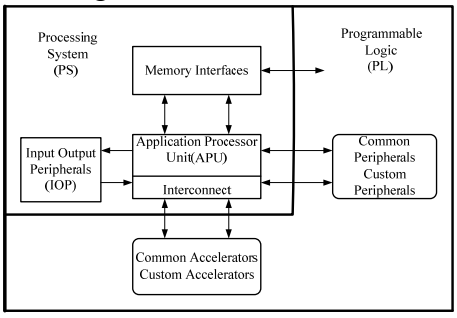

Fig. 1 Zynq block design

2. Principles of the design

2.1 Extensible of the Zynq Platform.

If you are designing an embedded processing system and you need lower system cost, sufficient performance, greater flexibility, competitive differentiation, faster time to market. Then you need to know about the Zynq Extensible Processing Platform.

The Zynq platform of Extensible Processing Platform (EPP) products consists of an SoC style integrated processing system (PS) and programmable logic (PL), providing an extensible and flexible SoC solution on a single die.

The PS integrates two ARM Cortex-A9 MPCore application processors, AMBA interconnect, internal memories, external memory interfaces, and on-chip standard peripherals including USB, Ethernet, SPI, SD/SDIO, I2C, CAN, UART, and GPIO. The PS is built using standard ASIC SoC design techniques and is designed such that it can run independently of the PL and boots at power-up or reset. This enables the Zynq platform EPP to achieve power, performance and cost characteristics of an SoC while retaining the flexibility offered by an FPGA. The PS, although embedded, is designed to be flexible and enables the user to configure aspects of the PS to meet their specific requirements.

The PL is the "extensible" part of the Zynq platform devices and is based on Xilinx's 7-Series FPGA technology. Users can add their custom logic using Xilinx ISE tools or Vivado design suite into the PL using standard ASIC/FPGA design techniques. User's designs are synthesized and mapped to the basic building blocks of the PL such as look-up-tables, block RAM, and DSP slices and then placed and routed using Xilinx tools. The PL also consists of integrated blocks including system monitoring with ADCs, security blocks, and in selected devices PCIe and high speed serial transceivers [2].

2.2 Extensible of the Data Acquistion System.

Data acquisition system is mainly composed of the ADC and other related interfaces. In this design, we chose ADS8568 to collect data. ADS8568 is a 16-bit 8-channel simultaneous sampling analog to digital converter (ADC) launched by TI company. These channels are divided into four pairs, which allows the data rate of up to 510 ksps synchronous high-speed signal's acquisition.

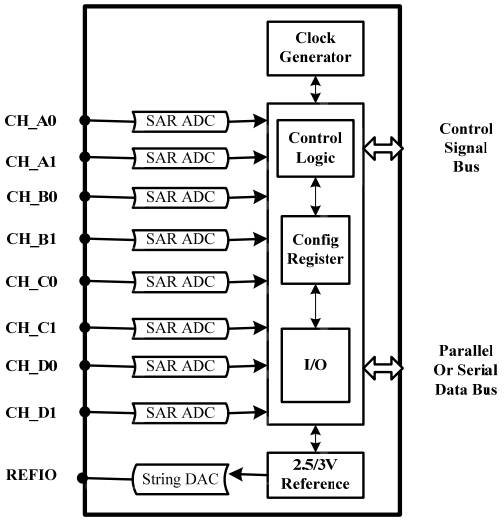

Fig.2 Schematics of ADS8568

ADS8568 has a Configuration Register, and can be switched to software mode or hardware mode through the corresponding pin level of control. When it turns to software mode, ADC writes data to configure the internal registers via a bidirectional data pins, and then control the ADC. When it turns to hardware mode, external signals are used to control the functions of ADC. The Configuration Register settings can only be changed in software mode and are not affected when switching to hardware mode thereafter. In this design, we used the default settings of the Configuration Register [4].

In short, ADS8568 has a wealth of features, including an optional serial and parallel input and convenient mode switching hardware, a programmable and buffer the internal reference voltage and a low-power automatic sleep mode. These features are likely to provide an alternative solution to encountered problem in engineering design. They are conducive to the practical application and can facilitate the user to expand and reduce massive changes in hardware circuit caused by local changes.

3. Design and Implementation

On Zynq platform, we used Xilinx's Vivado design suite to complete the design of program and hardware debugging. Vivado design suite integrates design environment that includes highly integrated design environment and new generation tools of different levels from system to IC, which is based on a shared extensible data model and a common debugging environment. It is also an open environment based on Advanced Microcontroller Bus Architecture (AMBA) AXI4 interconnect specification, IP-XACTIP package metadata, Tool Command Language (TCL), Synopsys System Constraints (SDC) and others that help to design according to customer's demand and industry-standard [3].

For the software design, ADC control program was designed to meet the job requirements according to the datasheet of ADS8568. In this design, we chose an eight-channel parallel sampling mode. The sampling rate can be controlled by the PC and it could be up to 510 ksps. ADS8568 control program was encapsulated into IP in Vivado software and then added into IP Catalog for subsequent design.

As Zynq AXI-DMA bus uses a 32-bit transfer to improve the work performance and resource utilization of Zynq, we expanded the ADS8568 output data to 32 bits by combining two adjacent 16-bit output data. At the same time, due to the AXI-DMA transfer rate is far greater than the sampling rate of AD, sustained reading and writing of AD sampling will increase power consumption to a large extent, resulting in a tremendous waste of resources. In this design, firstly, we saved ADC sampling data as cache into fifo (fifo depth set to 1024), then, when the fifo has storied 512 data, we sent a half-full signal to the Zynq processing system. Once the Zynq processing system received the control signal, it would make the PCI DMA fast read the 512 FIFO data to DDR3, which was used as the device's cache in order to increase the efficiency of reading and writing. Through opening the PCI DMA way to directly moved the collected data to the PC memory, and also could host computer to move data directly without CPU intervention for grestly increasing the speed of data transmission. Figure 3 shows the basic flow diagram of data acquisition system.

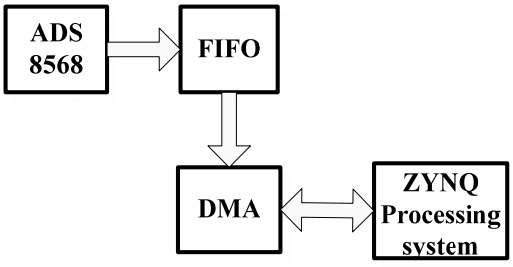

Fig.3 Data acquisition system basic flowchart

As for the hardware design, we separated the data acquisition board into two blocks, motherboard (MicroBoard on the Zynq platform) and baseboard (ADC data acquisition board). We connected the motherboard and baseboard through reserved interface and retained the basic interface of motherboard. The baseboard has some spare pins and connector locations aside for the subsequent improvement upgrades. By this means, we can facilitate the expansion later.

The physical map is shown in figure 4. The data acquisition board can be divided into the following main modules: A module is ADS8568 eight-channel analog signal input; B module is ADS8568 chip; C module is Zynq external clock signal input, using the chip cdclvd1204 to select optional 100MHZ oscillator or external input clock signal; D module is reserved pins for debugging and late improvement or upgrade; E module is the MicroBoard board on Zynq platform.

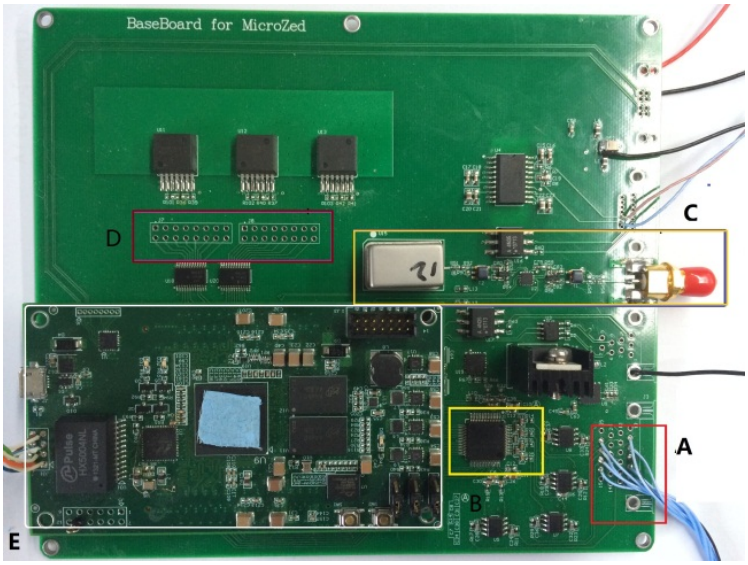

Fig.4 Physical map

For ADS8568 chip, while it is theoretically working in parallel, the descriptions of the main control signals list in table 1.

Table 1 Pin description

NAME	DESCRIPTION
CONVST	Conversion start signal.
BUSY	The BUSY signal indicates if a conversion is in progress.
nCS	Chip select input.
nRD	Read data input.
data_bus	The data output.

The analog inputs of each channel pair (CH_x0/1) are held with the rising edge of the corresponding CONVST_x signal in which "x" refers to A,B,C and D. The conversion automatically starts with the next rising edge of the conversion clock, e.g. ,CONVST_A is a master conversion start that resets the internal state machine and causes the data output to start with the result of channel A0. In this design, we need eight channels work at the same time, so we tied CONVST_A, CONVST_B, CONVST_C and CONVST_D together, and rename this signal "CONVST". BUSY signal goes high with a rising edge of any CONVST signal and goes low when the output data of the last channel pair are available in the respective output register [4]. The readout of the data can be initiated immediately after the falling edge of BUSY. When signal nCS and nRD are low at the same time, the parallel data output is enabled.

We setted the analog inputs of each channel the same sine wave. The sine wave's main parameters were setted as: the magnitude is 8vpp, frequency is 10KHZ, Y-axis offset is 0. Setted the sampling rate 500ksps, we could see the main signal waveform in the Vivado software integration waveform viewer, which is shown in Figure 5.

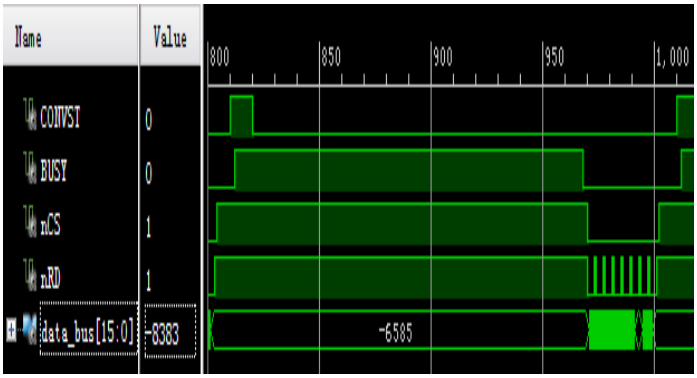

Fig.5 Main control signals

MicroBoard implements a 10/100/1000 Ethernet port for network connection using a Marvell 88E1512 PHY. MicroBoard exports the sampling datas which stored in DDR3 via Ethernet port to the computer, then extract the single-channel sampling data to recovery the signal analog waveform. The

result can be shown in Matlab as figure 6. The figure shows that the collection of data acquisition system is working properly.

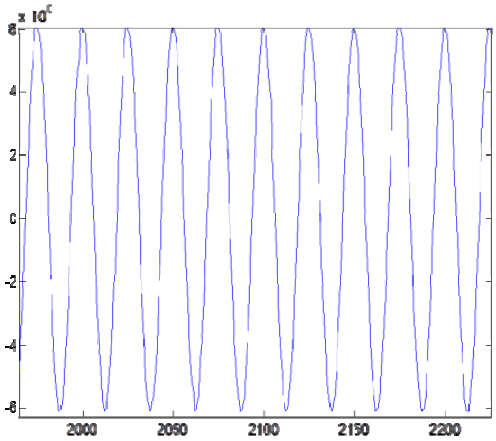

Fig.6 single channel data output of ADC

4. Summary

Based on the completion of the data acquisition system, extensibility of Zynq platform can be applied to design a variety of other functions to respond to the needs of specific applications. The design of extensible data acquisition system with a full use of the system integration capabilities of "All Programmable" devices based on the Zynq platform provides a wealth of possibilities for reducing design time and decreasing the cost for hardware and software's upgrading.

References

[1]. F. jingjing, "Realization of radar data acquisition system," academic dissertation of Nanjing university of science and technology, pp. 1-30, December 2009.

[2]. Xilinx UG804 (v1.1), "Zynq-7000 Extensible Processing Platform Summary," pp. 5-7, December 2011.

[3]. Xilinx, "Zynq-7000 EPP Software Developers Guide", pp.1-18, April 2012.

[4]. Texas Instruments, "12-, 14-, 16-bit, 8-channel synchronous sampling analog to digital converter", pp. 1-33, August 2011.