2010 First International Conference on Pervasive Computing, Signal Processing and Applications

The reading/ writing SD card system based on FPGA

Zhenlin LU
College of Information Science and Engineering,
Northeastern University,
Shenyang, China
e-mail: lzl_jb228@sina.com.cn

Jingjiao LI
College of Information Science and Engineering
Northeastern University,
Shenyang, China
e-mail: lijingjiao@ise.neu.edu.cn

Yao Zhang
College of Information Science and Engineering
Northeastern University,
Shenyang, China
e-mail: zhy20052395@sina.com

***Abstract*—the application based on FPGA chip are more and more popular especially for speech processing and image processing. At the same time, the application needs more memory except for on-chip memory to store additional data. To meet the requirements of external memory for FPGA-based systems, the writing/reading SD card system based on FPGA is designed and implemented. In this thesis, after analyzing the control principle of SD cards and SD card systems, the structure of 1-bit SD card transfer mode system is designed. The hardware design and implementation is completed on the QuartusII platform. The software including SD card driver and the corresponding control program is developed on the NiosII platform. At last, the system is tested and the result shows that the performance of the entire system can run stably and reliably, reach to the effect anticipative.**

Keywords: QuartusII; NiosII; 1-bit SD mode

I. INTRODUCTION

As advancement of FPGA chips rising and the advantages of high speed and parallel processing, the application based on FPGA chips used widely, especially for speech processing and image processing applications which call for the low-cost and micro-power demands. At the same time, during speech processing or image processing. It is important to find a storage device to store the large data files which are composite of voice data or image data [1].

SD Card (Secure Digital Memory Card) is a flash memory device based on a new generation of semiconductor memory devices. SD card with high memory capacity, fast data transfer rates, great flexibility and good mobile security. Considering of system size, power and storage capacity, it is easy to find out there are great advantages to choose SD card as the storage device for those systems which are based on the FPGA chip.

In this thesis, we adopted the 1-bit SD card transfer mode as a model to complete the image reading/writing [2]. The user can read a picture from the SD card or write a picture to the SD card by pushing different button.

The remaining of the paper is organized as follows. The principle of SD is briefly described in Section II. Design and Implementation of the reading/ writing SD card system is explained in section III. Experimental results are presented in Section IV. Conclusions are drawn in the last Section.

II. PRINCIPLE OF SD

A. transmission modes

The SD card support three transmission modes: SPI mode (separate serial input and serial output), 1-bit SD mode (separate command and data channels, a unique transmission format), 4-bit SD mode (use extra pins and some reset pin. to support 4-wide parallel transmission) [3,4]. SD mode has a faster access speed. The data transmission rate of 4-bit SD transfer modes can reach 0-100M/S; the data transmission rate of 1-bit SD mode is up to 0-25 M/S. Although the 4-bit SD transmission mode has a higher transfer rate, its complicated structure and timing make the system more difficult to develop. As the transmission rate required for the case is not very high, in this paper, 1-bit SD mode is chosen as a transmission mode [5].

B. SD Bus Protocol

SD communication principle is simple, and it works in master-slave mode. This mode often requires a master and multiple slave devices, using the lines from 3-6. They are CMD (line of control), CLK (clock line), and DAT0 ~ 3 (data line). Table1 shows the specific function of pin。

Table 1 pin function of SD mode

name	function
CLK	clock signal from host to SD card
CMD	Command signal from host to SD card or respond signal form SD card to host
DAT0~3	write data signals from host to SD card or read data signals from SD card to host

C. SD file system

The structure of FAT16 file system in SD card contains four parts. it is the Partition Boot Record(PBR), File Allocation Table(FAT), File Directory Table and Data Section

Usually, Partition Boot Record contains four elements: BPB(short for BIOS Parameter record Block), the disk flag record book, partition boot record code area and the end flag 55AA.

In the PBR, the BIOS parameter record block in which recorded some parameters is most important. The position of FAT, file directory table and data area (which sector in SD card) can be calculated through these parameters. Following the partition boot record is FAT area. FAT table records the

978-0-7695-4180-8/10 $26.00 © 2010 IEEE
DOI 10.1109/PCSPA.2010.107

link information of files which store between clusters. This is the chain store of files. Following the FAT table is the file directory table FDT (File Directory Table), which takes over 32 fixed sectors; each sector can hold 16 register items. The contents of the registration contain file name, file attributes, modification time, file length, etc Following FDT is the data area which takes up most of the disk space and is used to store file.

D. *SD command format*

All the SD Memory Card commands are 6 bytes long, including 1 byte command code, 4 bytes of command parameters and a checksum byte. Even using command without parameter, the host also has to send 4 bytes parameters when it sends commands to SD card. In this case, the parameter can be any value, and it will be automatically ignored by SD card. Command format is given in Table2

Table 2 SD card command format

The 1st byte			The 2nd ~5th bytes	The 6th byte	
7	6	5~0	31~0	7~1	0
0	1	Command code	Command parameters	Check code	1

The commands used in the design are shown in Table 3.

Table 3 SD card related command

command	Function
CMD0	Reset and Mode Selection
CMD2	Request all cards connected to the system send the values of its CID register back
CMD3	Request card send the relevant address (RCA) back
CMD7	Set the card enter programming state
CMD16	Setting the length of the read/write block. Reading and writing on the SD are block-based. The length of block must be set through the command before reading and writing
CMD24	Single-block write command
CMD25	Multiple-block write command
CMD32	Set erase block start
CMD33	Set erase end start
CMD38	Erase command

III. DESIGN AND IMPLEMENTATION

A. *System Structure*

This system used DE2-70 board to deign and develop the read-write SD Card system. The board carried EP2C70F896C6 chip which belongs to Cyclone II series.

The System includes the following modules as showed in Figure 1: NIOSII Processor, Data memory module, HCI module (LCD display modules, DIP switch module), Display interface module, SD card control module, JTAG module. The Figure 2 shows the hardware implement of the system on the QuartusII platform[6,7,8].

SD card interface module: connect the SD card , Store image information and achieve SD card read-write driver.

Data memory module: SDRAM is used to provide storage area for the system operation and cache the image data.

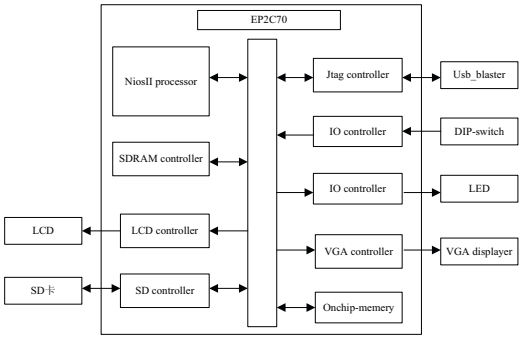

Fig.1 Figure of the structure of system

HCI Module: the LCD display module is used to display the information during SD card read-write processing.

JTAG module: it is used to download the program and debug online.

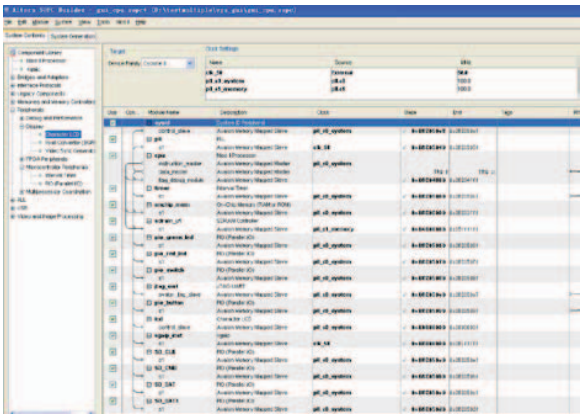

Fig 2. Figure of the structure of hardware

B. *SD Card initialization*

After powering on, the control logic initialized the SD card. According to the SD card protocol, CMD0 is firstly and then the SD cards will be set in idle state. After waiting at least 74 CLK clock cycles, we continue to send CMD55 + ACMD41 command which is used to detect the card status, if the highest bit (busy signal) of response returned is 0, it stands for the power / restart operations of SD cards has not be yet completed, we need to continue to send CMD55 + ACMD41 command until the power / restart operations of SD cards is completed and enter into READY status. Sending CMD2 in this state, this command makes the connected SD card send message back and then enter into indent status. In the indent state, control logic send CMD3 to ask the SD card to return the self-address RCA (Relative Card Address). When all above steps are completed successfully, the control logic sends CMD7 command, which will transform SD card from stby (standard) state into the tran (transmission) state. In the end, Sending CMD16 command to set the data transfers module size 512 bytes. Finally, SD card initialization process has been completed. The flow chart is showed in Figure 3.

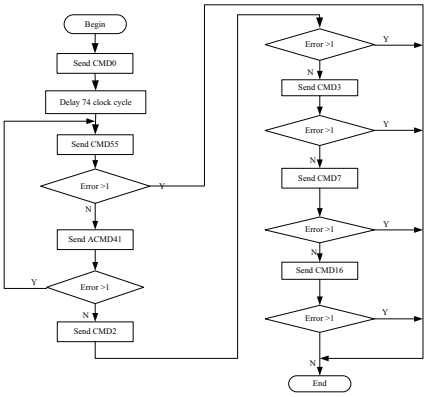

Fig 3 Flow chart of SD card initialization

C. SD card data reading

The command of reading data for the single block is CMD17. When the SD card receives a CMD17 command, it will in accordance with the data block address that is sent with the command to search for the data to be read. Look down the data line after successfully finding. When the system detects the data in the data line is low, it prepares to begin to read the data in the next clock, and the data size is 512 bytes. After receiving 512 bytes data, it followed by a 2 bytes checksum. Thus, the SD card single block of data reading is completed. As shown in the Figure 4.

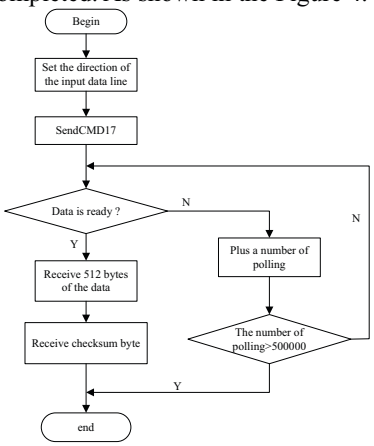

Fig 4 Flow chart of SD card read data

D. SD card data writing

It is relatively complex to write data into a SD card. The command of writing data for the single block is CMD24.When the SD card receives a CMD24 command; it will send a response token to the host and wait for the host's data block. If the answer of the R1 is 0, the host can send a block of 512 bytes. Then the host needs to send a 16-bit check code obtained by looking up the table in the system. After receiving the data block and the check code, the SD card will return a response token of a type to the data block. When the five-low number is 00101, the data block is written correctly. If the validation of the host's check code fails, the transmitted data will be lost and the course of the transmission will be terminated. The operation of the SD card data writing is shown in the figure 5.

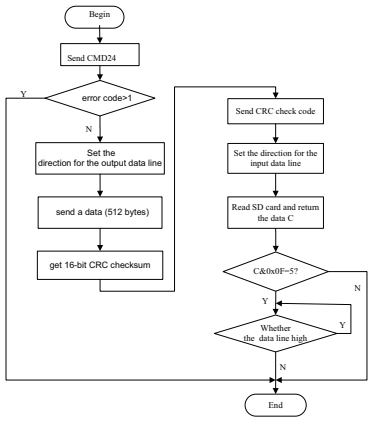

Fig 5 Flow chart of SD card data write

E. SD card erasing

The command CMD32 is used to send the starting address of erasing sector. Continue to send the ending address of erasing blocks after successfully sending the command and its command is CMD33. And similar to CMD32 the parameter is the ending address of erasing blocks. Send the erasing command CMD33 in the end. This command begins to execute the erasing operation. When the specified sector is erased completely, the SD card pulls the data transmission line. If the system reads the data line is high, it returns 1 that is successfully erased. The flow chart of SD card erasing is shown in the figure 6.

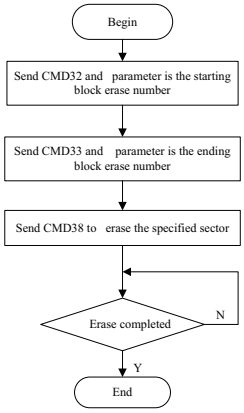

Fig 6 Flow chart of SD card erase

IV. SYSTEM TEST

The system is used for reading and writing on SD card. A test was performed on it after it has developed, and the system uses a picture file as test data. The test mainly checked the performance of three aspects as follows: first, recognizing the SD card, and alerting users through interface used for Human-computer interaction; second, reading the pictures from SD card, and displaying them in VGA monitor; thirdly, writing the pictures of sdram into SD card, and prompting users through Human-computer interaction interface. We can see the test results range from Fig8 to

Fig10. The whole system is shown as Fig 7, in which the picture file is been reading; and the Fig 8 shows NiosII console is displaying the condition of reading successfully, the Fig 9 shows NiosII console is displaying the condition of writing successfully.

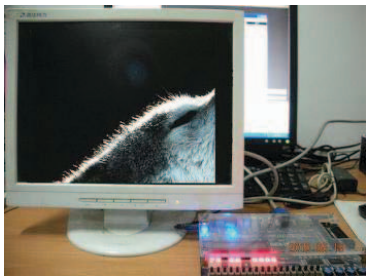

Fig 7 Figure of real system

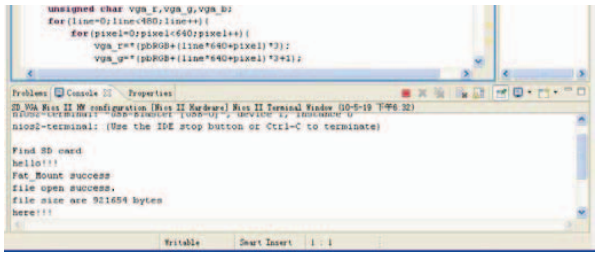

Fig 8 Figure of finishing reading picutre information

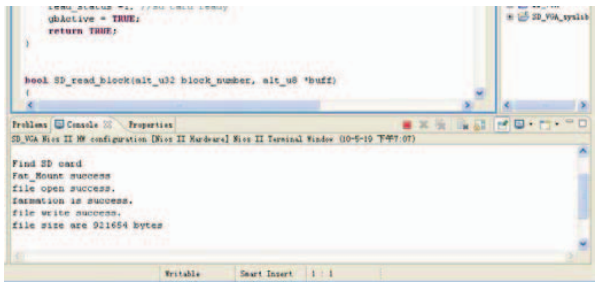

Fig 9 Figure of finishing writinging picutre information

V. CONCLUSION

Firstly, this paper gives a brief introduction to the relevant information about the SD card. Then based on FPGA for the SD memory card reader requirements, it designs FPGA-based SD card reader system architecture with the various functional modules for design and development. The system was functional tests. By the test results, you can see that the system can read and write data on SD card, good to meet the FPGA system needs for external storage devices. At the same time, this involved the SD card reader technology that can be applied to a variety of FPGA-based system. Therefore the use of this technology should also have more potential applications.

REFERENCES

[1] PANG Tao, SHI Zhong-ke, “Application of SD Card in Flying Trial Data Recorder,” Measurement & Control Technology, vol.28, pp.11-14. October 2009.

[2] Chen Guozhu, Liu Tao, LI Yuanzong, “Design of Image Sampling and Storage System Based on FPGA,” Mechanical Engineering and Automation, vol.4, pp.44-46, 2007.

[3] Wang qing, Liu xinning, “Design and Implementation of the Control Logic for SD Card Initialization and Data Storaging,” Computer Knowledge and Technology, vol.4, pp.990-991, April 2008.

[4] SI Hai, Liu Zhaodu, WANG Junzheng, LI Jingliang, “Design of Driving Behavior Reconstruction Storage System Based on SD Card,” Control and Automation, vol.25, pp. 275-276, 2009.

[5] Peng Di, SHI Juanjuan, “Design and Implementation of SD Card Storage Scheme Based on Platform of SH_Mobile,” Microcomputer Applications, vol.29, pp.88-92, 2008.

[6] Fang zhuo, Chen Zewen, peng chenglian, “User-defined Logic in SOPC Design,” Computer Engineering, vol.30, pp.12-15, 2004.

[7] Wang Jianxiao, SOPC Design Basis and Practice, 2006.

[8] Zhou ligong, SOPC embedded systems Essentials, 2006.