Design of ACFM System on Chip Based on Nios ii

Shunke Ye[1, a], Shangkun Ren[1, b], Peng Wei[1, c], Liuci Zhou[1, d]

[1]Key Laboratory of Nondestructive Testing of Ministry of Education, Nanchang Hangkong University, Nanchang, 330063, China

[a]365246766@qq.com, [b]renshangkun@yeah.net, [c]1064322931@qq.com, [d]1060088564@qq.com

Key Words: ACFM; Nios ii; FPGA; SOPC

Abstract: ACFM is a new kind of non-destructive testing in electromagnetism.Fast speed, high accuracy, no much requirement for testing environment make it more and more popular in NDT.This paper presents an embedded solution of System-On-Chip for ACFM based on soft-core processor Nios ii which can integrate many functions such as DDS module, A/D module, user-defined instruction, encoding of collected data into a FPGA chip.Using Qsys of Quartus ii, the System-On-Chip for ACFM based on Nios ii is finally designed through hardware/software co-development which includes the design of modules for system, the construction of system and the programming on system.

Introduction

As large development of micro-electronics, the SOPC based on FPGA and Nios ii provide advanced technology and extensive application for market of which competition is fierce, because of its flexible design, tailorability and programmability[1].Furthermore, ACFM is a real-time non-destructive testing technology, making more and more popular in NDT because of its fast detection for surface and near-surface cracks in conductor[2].

In order to make full use of the real-time and fast detection of ACFM, this paper presents an embedded solution which can integrate modules of ACFM into the Nios ii system on FPGA chip.This solution can not only lowers the cost, complexity, size and power dissipation of ACFM instrument, but also makes the ACFM instrument more intelligent.Meanwhile, hardware/software co-development based on SOPC of Nios ii can effectively reduce the period of design, which has significant meaning for exploitation of ACFM instrument[3].This paper presents the hardware/software co-development in ACFM system based on Nios ii through the design of modules for system, the construction of system, the programming on system.

The design of modules for system

ACFM detects by stimulating a alternating magnetic field which will produce induced current in conductor's surface.Because induced current will distort encountering cracks in surface or near-surface, it is necessary to collect and analyze this distortion current which can measure the size of cracks.According to the processing of ACFM testing, it needs to design two module for ACFM system on chip:

Signal Generator Module

This module use the DDS to generate a sine waveform current which will stimulate an

alternating magnetic field by conducting through a coil.According to the theory of DDS, this module needs a D/A chip on hardware and Verilog programming on software.Programming upon the FPGA chip by Verilog language realize a signal generator module which can control the frequency of signal.The Verilog module is as Program 1.1.

```
dds_sin      u0_dds_sin(
                    .clk(clk),              //clock input
                    .rst_n(rst_n),          //async reset , active low
                    .en_dds(en_dds),        //enable control, active high
                    .chmod(chmod),          //enable frequency change, active high
                    .Fword(fword),          //frequency step word input

                    .DAC_CLK(DAC_CLK),      //D/A clock
                    .q(DAC_DATA),           //data output from ROM D/A input
                    .clk_n(rectangle)       //a rectangle waveform
                    );
```

Program 1.1 Signal Generator Module

In Program 1.1, the port clk is the clock input.The port rst_n is the asynchronous reset input.The port chmod is used to enable the control of the frequency.The port Fword is to input the new frequency step word.The port DAC_CLK and the port q is utilized to connect the D/A chip.And the last port clk_n is to generate a rectangle waveform for switch capacity filtering in ACFM system.

A/D Collection Module

After that the alternating magnetic field produced by Signal Generator Module induce current in conductor, it needs to change those current into digital signal for further analysis which can judge whether it exists a crack, what kinds of this crack and what is the size of the crack.The FPGA chip having lots of ports is easy to drive a multichannel A/D chip to collect data.The Verilog module for driving a multichannel A/D is as Program 1.2.

```
AD7606      u0_AD7606
//----------- Ports Declarations ----------------------------------------------
(
//clock and reset signals
.fpga_clk_i(clk),           //system clock
.reset_n_i(rst_n),          //active low reset signal

//IP control and data interface
.wr_data_n_i(da_wr),            // active low signal to initiate a data write to the ADC
.data_i(ad7606_data_i),         // channel[7:5], os[4:2], standby[1], range[0]
.data_o(ad7606_data_o),         // data read from the ADC
.data_rd_ready_o(ad7606_rd),    // when set to high the data read from the ADC is available
on the data_o bus

//AD7606 control and data interface
.adc_db_i(ADC_DATA),            // ADC parallel data bus
```

```
.adc_busy_i(ADC_BUSY),          // ADC BUSY signal
.adc_os_o(ADC_OS),              // ADC OVERSAMPLING signals
.adc_range_o(ADC_RANGE),        // ADC RANGE signal
.adc_cs_n_o(ADC_CS),            // ADC CS signal
.adc_rd_n_o(ADC_RD),            // ADC RD signal
.adc_reset_o(ADC_RST),          // ADC RESET signal
.adc_convst_o(ADC_CONVST),      // ADC CONVST signal
.channel_read(ad7606_ch)        //ADC CHANNAL
);
```

Program 1.2 A/D Collection Module

Because of large number of ports, the functions of each port can refer to notes in Program 1.2.Owing to the fast running speed in FPGA, it is easy to realize the multichannel signal acquisition at the same time.After programming on system, the ACFM system on chip can collect datum periodically by this module and encode the datum as frame which will be sent to upper computer for waveform display.

The construction of system

For the construction of ACFM system on chip, it needs to use the SOPC tools of Altera, Qsys.Open Qsys, and then add the IP core needed such as CPU, SystemID, clock, timer, sdram, flash, serial port, PLL and the ports for signal generator module and A/D collection module, which is shown in Fig.1.

Fig.1 the list of IP cores in system

Fig.1 shows the list of IP cores and the distribution of IRQ and Address for each IP.After connecting the ports of those IP cores, we can generate a software core of Nios ii based on Verilog language as Program 2.1.

```
ACFM_Qsys u0 (
.clk_clk            (<connected-to-clk_clk>),            //        clk.clk
.sdram_addr         (<connected-to-sdram_addr>),         //      sdram.addr
.sdram_ba           (<connected-to-sdram_ba>),           //           .ba
.sdram_cas_n        (<connected-to-sdram_cas_n>),        //           .cas_n
.sdram_cke          (<connected-to-sdram_cke>),          //           .cke
.sdram_cs_n         (<connected-to-sdram_cs_n>),         //           .cs_n
.sdram_dq           (<connected-to-sdram_dq>),           //           .dq
.sdram_dqm          (<connected-to-sdram_dqm>),          //           .dqm
.sdram_ras_n        (<connected-to-sdram_ras_n>),        //           .ras_n
.sdram_we_n         (<connected-to-sdram_we_n>),         //           .we_n
.uart_rxd           (<connected-to-uart_rxd>),           //       uart.rxd
.uart_txd           (<connected-to-uart_txd>),           //           .txd
.ad7606_wr_export       (<connected-to-ad7606_wr_export>),       //ad7606_wr.export
.ad7606_data_i_export   (<connected-to-ad7606_data_i_export>),   //ad7606_data_i.export
.ad7606_rd_export       (<connected-to-ad7606_rd_export>),       //ad7606_rd.export
.ad7606_data_o_export   (<connected-to-ad7606_data_o_export>),   //ad7606_data_o.export
.ad7606_ch_export       (<connected-to-ad7606_ch_export>),       //ad7606_ch.export
.pll_areset_export      (<connected-to-pll_areset_export>),      //pll_areset.export
.pll_locked_export      (<connected-to-pll_locked_export>),      //pll_locked.export
.pll_phasedone_export   (<connected-to-pll_phasedone_export>),   //pll_phasedone.export
.sdram_clk_clk          (<connected-to-sdram_clk_clk>),          //sdram_clk.clk
.rst_n_reset_n          (<connected-to-rst_n_reset_n>),          //rst_n.reset_n
.en_dds_export          (<connected-to-en_dds_export>),          //en_dds.export
.chmod_export           (<connected-to-chmod_export>),           //chmod.export
.Fword_export           (<connected-to-Fword_export>),           // Fword.export
);
```

Program 2.1 the Nios ii module

From Program 2.1, we can see that the ports of the module is corresponding to the list of IP cores in Fig.1.After generating this Nios ii module, the next step is to connect this module with the signal generator module and the A/D collection module designed before.Then the programming in software is finished.

Afterward, to realize the software/hardware co-development, it is necessary to build a module based on FPGA chip to connect the chips in hardware such as D/A chip, A/D chip, crystal, sdram, serial port and reset port.This module is as Program 2.2.

```
module ACFM        (
                     //clk
                     input                clk,
                     input                rst_n,

                     //input
                     input                uart_rx,        //UART
```

```
input      [15:0]        ADC_DATA,
input                    ADC_BUSY,          //AD7606

//output
output                   uart_tx,           //UART

output   [12:0]          sdram_addr,
output   [1:0]           sdram_ba,
output                   sdram_cas_n,
output                   sdram_cke,
output                   sdram_cs_n,
output                   sdram_udqm,
output                   sdram_ldqm,
output                   sdram_ras_n,
output                   sdram_we_n,
output                   sdram_clk,         //sdram

output    [2:0]          ADC_OS,
output                   ADC_RANGE,
output                   ADC_CS,
output                   ADC_RD,
output                   ADC_RST,
output                   ADC_CONVST,        //AD7606

output                   DAC_CLK,
output   [7:0]           DAC_DATA,
output                   rectangle,         //DDS

//inout
inout          [15:0]    sdram_dq
);
```

Program 2.2 the top module

And then, open the Pin Planner of Quartus ii to distribute the port defined in Program 2.2 to physical ports in FPGA.So far, the construction of ACFM system on chip has finished, which realizes software/hardware co-development by connecting the ports of each modules and chips.

The programming on system

After the construction of system, we need to program for ACFM system based on Nois ii.The programming is to realize the functions as follow: collecting data periodically by A/D collection module, encoding the data to send to upper-computer, changing the frequency of alternating magnetic field by signal generator module, designing an instruction set for users to operate this system.

With the help of the software Eclipse For Nios ii, we can build a project like C based on the file (.sopc) produced by the construction of system.Eclipse will help to produce the file system.h based on system.In the file system.h, we can find the base addresses of IP cores.Using these addresses, we can use IP cores like C programming.To be different from SCM, we can freely add the IP core or the functions as we need.From which we can see that the Nios ii is the most general and flexible software processor in the world.The key C programming for ACFM system on chip is as Program 3.1.

```
if ((receive_buffer[receive_count-1]=='s')&&(receive_buffer[receive_count-4]=='c'))
//change the frequency of alternating magnetic field
{
char mchar, lchar;
mchar = ConvertHexChar(receive_buffer[receive_count-3]);
lchar = ConvertHexChar(receive_buffer[receive_count-2]);
freq = mchar*16 + lchar;
model_set = 6;
receive_count=0;
}
```

Program 3.1 the instruction to change frequency

This Program 3.1 is used to receive instructions from the upper-computer by serial ports.After receiving instruction, extract the parameter from the instruction to change the frequency by signal generator module.The format of instructions is c01s to c10s, which means the signal generator module can change the frequency of alternating magnetic field from 1kHz to 10kHz.

In the next, we design a instruction set for users to operate the system.The format of instructions is m01l to m03l.The instruction m01l means to command the system to start testing, m02l means to stop testing, m03l means to reset testing.The key C programming to realize this function is as Program 3.2.

```
void uart_ISR(void* nirq_isr_context)
{
    while(!(UART->STATUS.BITS.RRDY));
    receive_buffer[receive_count++] = UART->RXDATA.BITS.RECEIVE_DATA;
    if ((receive_buffer[receive_count-1]=='l')&&(receive_buffer[receive_count-4]=='m'))

    {
        model_set = receive_buffer[receive_count-2]-48;
        receive_count=0;
    }
    else
    {
        model_set = 5;
    }
}
```

Program 3.2 the instruction to operate the ACFM system

Program 3.2 realizes the function that upper-computer operates the ACFM system by sending instructions through serial port, which makes ACFM instrument more intelligent.Last but no least, in order to send data to upper-computer, it is necessary to design a frame for data.The specific program is as Program 3.3.

```
data_uart[0] = 0xff;
data_uart[1] = 0xff;
data_uart[2] = 0xff;
data_uart[7] = 0x00;
data_uart[8] = 0x00;
while (RD)
{
if(CH == 1)
{
data_uart[3] = (uchar)(data_o & 0x00ff);
data_uart[4] = (uchar)((data_o & 0xff00)>>8);
}

else if (CH == 2)
{
data_uart[5] = (uchar)(data_o & 0x00ff);
data_uart[6] = (uchar)((data_o & 0xff00)>>8);
uart_send_string(9, data_uart);
}
}
```

Program 3.3 Frame for data transmission

Program 3.3 is used to collect signals from two ways by A/D collection module.The signals collected are changed to binary and encoded to a frame whose format is FF FF FF XX XX XX XX 00 00.And then send these encoded frames to upper-computer for waveform display.

The running of system

Write program into FPGA chip, and then connect FPGA chip with the other chips.Use computer as upper-computer to connect with FPGA through serial port.Open the Serial Debugging Assistant to send “m01l” to FPGA.And then, upper-computer can receive the encoded data from the ACFM system as Fig.2.

```
FF FF FF 02 C8 25 01 00 00 FF FF FF 02 C8 27 01 00 00 FF FF FF 02 C8 2C 01 00 00
FF FF FF 02 C8 29 01 00 00 FF FF FF 02 C8 2B 01 00 00 FF FF FF 02 C8 35 01 00 00
FF FF FF 02 C8 2C 01 00 00 FF FF FF 02 C8 2A 01 00 00 FF FF FF 02 C8 30 01 00 00
FF FF FF 02 C8 19 01 00 00 FF FF FF 02 C8 11 01 00 00 FF FF FF 02 C8 25 01 00 00
FF FF FF 02 C8 35 01 00 00 FF FF FF 02 C8 40 01 00 00 FF FF FF 02 C8 42 01 00 00
FF FF FF 02 C8 38 01 00 00 FF FF FF 02 C8 39 01 00 00 FF FF FF 02 C8 3F 01 00 00
FF FF FF 02 C8 39 01 00 00 FF FF FF 02 C8 34 01 00 00 FF FF FF 02 C6 28 01 00 00
FF FF FF 02 AE 1C 01 00 00 FF FF FF 02 C7 1E 01 00 00 FF FF FF 02 C8 12 01 00 00
FF FF FF 02 95 12 01 00 00 FF FF FF 02 A5 1C 01 00 00 FF FF FF 02 C7 23 00 00 00
FF FF FF 02 C8 2A 01 00 00 FF FF FF 02 C8 27 01 00 00 FF FF FF 02 C8 26 01 00 00
FF FF FF 02 C8 1B 01 00 00 FF FF FF 02 C8 11 01 00 00 FF FF FF 02 C8 0E 01 00 00
FF FF FF 02 C8 1B 01 00 00 FF FF FF 02 C8 09 01 00 00 FF FF FF C8 C8 02 01 00 00
FF FF FF 02 C8 16 01 00 00 FF FF FF 02 C8 25 01 00 00 FF FF FF 02 C8 2A 01 00 00
FF FF FF 02 C8 2E 01 00 00 FF FF FF 02 C8 30 01 00 00 FF FF FF 02 26 01 00 00 00
FF FF FF 02 C8 3A 00 00 00 FF FF FF 02 C8 38 00 00 00 FF FF FF 02 C8 29 01 00 00
FF FF FF 02 C8 23 00 00 00 FF FF FF 02 C8 01 00 00 00 FF FF FF 02 A2 01 00 00 00
FF FF FF C8 33 01 00 00 00 FF FF FF 02 2C 01 00 00 00 FF FF FF 02 C8 01 00 00 00
```

Fig.2 Transmission of encoded data

Fig.2 shows that the encoded data transmits from FPGA to computer, which proves that the design of ACFM system on chip is successful and meets requirements of ACFM.

Conclusions

This paper presents the programming and construction of ACFM system on chip, displays the SOPC development based on FPGA.From the development process, we can see the advantages of software/hardware co-development:

1)Use less chips, but achieve more functions, which effectively reduces the cost, complexity, power dissipation and the size of ACFM instrument.

2)Customize the system on chip flexibly, and cut the dress according to ACFM's figure, which not only realizes the reserved functions, but make ACFM instrument more intelligent.

3)Use Altera's SOPC tools to develop system on chip, which not only simplifies the development processing, but reduced the development period.At the same time, it is beneficial to further upgrading of function, having significance for development of ACFM instrument.

Reference

[1] Jiejun Luo. The Research of SOPC Based on Nios ii. Harbin Institute of Technology. 2009. In Chinese.

[2] Shangkun Ren, Zhibin Zhu, Tianhua Lin, Kai Song, Ren Jilin.Design for the ACFM Sensor and the Signal Processing Based on Wavelet Denoise.2009 2nd International Congress on Image and Signal Processing (CISP'09), Pressed by IEEE.

[3] Zongshu Pan. The Design and Research of SOPC System Based on Nios ii. Wuhan University of Science and Technology. 2007. In Chinese