Application of EDA technology and μ Clinux operating system in the earthquake precursor instruments*

Anhua He^{1,*} Gang Zhao¹ Na Xue² Jun Wang³
Bailin Guo¹ Miaoxi Guo¹ and Zizhong Fu¹

¹ *Institute of Crustal Dynamics, China Earthquake Administration, Beijing 100085, China*

² *Earthquake Administration of Tianjin Municipality, Tianjin 300201, China*

³ *China Earthquake Networks Center, Beijing 100036, China*

Abstract Using Altera's Quartus II, Nios II IDE and Sopc Builder development tools, the proton precession magnetometer principle host hardware platform is designed in a cyclone II series FPGA chip (EP2C35). The proton precession magnetometer principle host core circuit's single-chip system-logic design is achieved by building and configuring the Nios II soft-core processor, developing the IO interface and sensor control circuits, programming some hardware units' VHDL code, for example the equal precision cymometer and the DPLL. Through researching the embedded operating system configuration technology and building the NIOS II soft-core processor's μ Clinux cross-compile environment, the μ Clinux system is transplanted to the NIOS II environment. Another important task is writing the device drivers' and user programs' code. Through these work, the design realize the host function and achieve the expected target.

Key words: EDA; μ Clinux; proton precession magnetometer; device driver; DPLL; equal precision cymometer

CLC number: P315.62

Document code: A

1 Introduction

Electronic Design Automation (EDA) is the important development frontier and core in the field of electronic science. Based on the computer technology, according to the HDL documents, EDA automatically complete the compile, simplification, segmentation, synthetic, optimization, placement routing, simulation, adaptation, program, download nine steps. Compared with the traditional design method, EDA shows great superiority in designing digital system such as taking the HDL as input; possessing independent intellectual property, system modeling and simulation function, development technology standardization, IP cores exploitability, high speed performance and so on (Pan and Huang, 2002). It is suitable for top-down design of high-efficiency large-scale system.

Linux is open source software, stable, mature and

practical operating system. The source-code base tailoring method, in other words, the kernel can be cut according to actual need. μ Clinux (Micro-Control-Linux) is the result of the Linux gradually application in the embedded equipments.

System On Programmable Chip (SOPC) is put forward by Altera Corporation that is a nimble and highly effective SOC solution. It is composed of processor, memory, I/O port, CDR and so on functional modules on a programmable component. This processor, NIOS II CPU, is the most popular and configurable soft-core processor (Altera, 2008).

The proton precession magnetometer is used to measure the total intensity of geomagnetic field's absolute or component value. In the direction of the magnetic field, a Larmor precession will be produced when a spin is rolled in the geomagnetic field with spin magnetic moment and spin angular momentum. The relationship of the precession frequency and the geomagnetic field's intensity is (Department of Monitoring and Forecasting, China Seismological Bureau, 2002):

* Received 7 August 2008; accepted in revised form 1 November 2008; published 10 February 2009.

* Corresponding author. e-mail: hua98611@163.com

$$F = 2\pi \frac{f_p}{\gamma_p} = 23.487215 \times f_p, \quad (1)$$

Where F is total intensity of geomagnetic field (nT); f_p is proton precession frequency (Hz); $\gamma_p = 0.26751513 \times 10^8$ rad/T·s represents gyro magnetic ratio.

According to this formula, the total intensity of geomagnetic field and the proton precession frequency have direct proportional relation, which is the measuring principle of the proton precession magnetometer.

This article's mainline is researching the proton precession magnetometer principle host. Seismic precursory instrument will be dramatically improved the

intelligent degree, reduced the production and maintenance cost, facilitate its replacement and so on by using the mature EDA technology and embedded Linux.

2 Hardware design

The hardware source mainly includes a CPU (NIOS II with 4K on-chip RAM), storage devices' interface, peripheral interface (such as display, buttons, network card, serial port and so on), and custom devices' interface. These interfaces connect with CPU through Avalon bus. Figure 1 shows the architecture of system hardware.

Figure 1 System hardware architecture.

The hardware configuration information and operating system image files are all stored in an external flash. The instrument start includes two steps: the first step is copying the hardware information from flash to FPGA; the second step is starting the bootstrap program. The operating system image file is decompressed and copied to the address 0 of SDRAM after the bootstrap program. In the end, the Linux system will be succeed in starting from the address 0 of SDRAM (Liu, 2005).

2.1 System architecture design by SOPC builder

SOPC builder provide a system platform for setting up one module-level and component-level definition (Altera, 2008). The component library contains a series of components, from the simple and fixed logic function block to complex and can parameterization, dynamic product subsystem component. The design mainly uses the NIOS II CPU module (standard kernel, 4K on-chip RAM), system clock, Avalon bus, and some interface modules. The system main frequency is 85 MHz.

2.2 Equal precision cymometer design by VHDL

In the design, the equal precision cymometer is mainly used to sample the probe signal and feedback the result to CPU for computation. This cymometer not only possess high precision (its measuring precision is decided by reference frequency), but also can maintain the constant measuring precision in the entire frequency region. The Figure 2 shows the measurement principle of cymometer.

Figure 2 The principle of equal precision cymometer. stand_data [31..0] is the result of standard signal counter (32-bit); test_data [31..0] is the result of test signal counter (32-bit).

In one measure cycle, it hasn't counter error relative to the test signal because the start and stop time of counters is triggered by the test signal. Due to the standard frequency counting integer (defined as N_s) difference a pulse at most in a cycle time; the theoretical error is $\leq 1/N_s$. Obviously, if the N_s is big enough it can guarantee the measuring precision because it merely related to N_s . The following formula shows the relationship between the standard signal and the test signal

$$\frac{N_s}{f_s} = \frac{N_x}{f_x} = t \quad (\text{The Q-side of D flip-flop}), \quad (2)$$

where f_s is the frequency of standard signal; N_s is the standard frequency counting integer; f_x is the frequency of test signal; N_x is the test frequency counting integer.

Its derivation formula:

$$f_x = \frac{f_s}{N_s} N_x, \quad (3)$$

In the formula, f_s is a known quantity, N_s and N_x can read from the counters, so the test signal frequency can

be calculated easily. Based on the above analysis, the equal precision cymometer is divided into a D flip-flop and two counters. Certainly, it is very easy to write these modules code in VHDL (Jiang and Liu, 2004).

2.3 DPLL

In the design, the center frequency of switched-capacitor filter in this system is provided by DPLL. Chinese geomagnetic total intensity, from the south to the north, changes from 41 000 nT to 60 000 nT. According to the principle formula, the filter center frequency change from 80 KHz to 320 KHz. The step is set at 5 KHz. DPLL composed of a digital phase detector (DPD), a digital loop filter (DLF), a digital control oscillator (DCO) and N -counter. DPD is a exclusive-OR gate detector; DLF is a mode-changed counter; DCO is a plus-minus pulse controller. Obviously, DPLL is divided into several simple modules and complete these VHDL code easily (Wang and Zhong, 2003).

The specific application in the design is shown in Figure 3.

Figure 3 The specific application of DPLL. reset_n: the signal of reset; dppll_switch: the signal of restoring the factory setting; dppll_param[7..0]: the input parameter of DPLL and it control the output frequency; dco_out: the output frequency; display: the auxiliary display. In order to achieve objective that the CPU can select center frequency, the dco_out is controlled by dppll_param[7..0] and its pin connect with the center frequency pin of filter in the external circuit.

2.4 Measurement, control and amplifier circuit

Figure 4 shows the hardware workflow (Department of Monitoring and Forecasting, China Seismological Bureau, 2002). The proton precession sensor is incented for 5 s after the IS command from CPU, otherwise it is reset state in normal times. Then, the sensor signal will be accessed to the signal amplifier after disconnect current. The signal is 2 s duration and micro-volt level sinusoidal signal. First, it is amplified by the capacitor harmonic distribution (the CAP [7..0] is used to select the capacitance), and then into the INA128 preamplifier. Second, it is selective amplified through the switched-capacitor filter which is controlled by CPU (Q[5..0] is decided to select center frequency). At last, it is plastic and eventually generates square feedback to system by OP07 and MAX953. After above

all steps, the signal's sample, compute, data storage and so on are finished in CPU.

3 Operating system: μ Clinux

3.1 Building of cross-compiling environment about μ Clinux

The cross-compiling tool is nios2gcc.tar.bz2 source code package and adopt "git server" to manage it. "Git Pull" is used to get the last kernel patch after download and decompress the μ Clinux source code package. "Make menuconfig" mainly configures the kernel, such as serial communication protocol, supports CF card, and adds FAT file system and so on. "Make" is used to finish the source code compiler and eventually generate the image file which will be downloaded to flash device. After above steps, it has achieved the system migration.

Figure 4 Measurement, control and amplifier circuit. IR: the reset signal of sensor; IS: the test signal of sensor; CAP[7..0]:the select signal of harmonic distribution capacitor; Q[5..0]:the Q set signal of filter; CENTER_FRE: the center frequency of filter.

3.2 File system and data security

The root file system of μ Clinux is ROMFS which request much less memory relative to EXT2. Because the ROMFS does not support dynamic write, it can ensure the security of operating system and system data (Liu, 2000). In addition, the design also loads FAT which supports the CF card data. CF card is dynamic mounted in the design, in other words, it will be unloaded after finish access immediately.

Through above operations, the system and measure data will be protect very well.

4 Software design

The design mainly includes two kinds of software, the device driver and user program. The former runs in kernel space and the latter runs in user space.

Figure 5 is the software system structure diagram. The tiptop is application layer which includes serial communication, socket server and so on. The third layer is Linux kernel which includes process schedule, memory management, file system, device driver, and network subsystem. The system call interface is located between the two layers; it is the only legal channel that applications access the kernel. The bottom layer is device driver and it can be thought as the bridge between the hardware and operating system. Natural, the hardware is followed device driver.

Figure 5 Software system structure.

4.1 Driver

As a part of Linux kernel, device driver is a bridge which connects the user program with hardware. It can be compiled into the kernel source code and also can be compiled as a single module which insert core in run-time. In the design, all peripheral interfaces require own drivers and they are compiled into the kernel code (Corbet et al, 2005). It have about six peripherals: LCD display, LED, setting the filter Q value, setting the filter center frequency, setting the equal precision cymometer and controlling sensor. Their major equipments are defined in the “mydriverparam.h”.

```
# define MYLCD_MAJOR      250 // LCD display major equipment
# define MYLED_MAJOR      249 // LED major equipment
# define SET_Q_MAJOR      248 // setting the filter Q value
# define CENTERFREQUENCY_MAJOR  247 // setting the filter center frequency
# define TEST_MAJOR      246 // controlling sensor
# define CYMOMETER_MAJOR  245 // setting the equal precision cymometer
# define SETCAP_MAJOR     244 // capacitor selection
```

The “file-layer and driver-layer” interface is used in the design due to all of the custom hardware belong to the character equipments. For example, the driver-layer interface of LCD frame buffer is described below:

```
#define MYLCD_IOCTL_BASE 0xb1
```

```

#define MYLCD_CLEAR
_IO(MYLCD_IOCTL_BASE,0)
#define MYLCD_LINE
_IOW(MYLCD_IOCTL_BASE,1,unsigned long)
static struct file_operations my_fops = {
    .read = NULL,
    .write = mylcd_write, //Data for writing and
displaying
    .ioctl = mylcd_ioctl, //Control setting
    .open = mylcd_open, //open
    .release= mylcd_release,//close
};

```

If the application wants to operate the frame buffer device (/dev/mylcd) only call the function of file-layer. For example, file.open is called for opening /dev/mylcd, file.ioctl is used to control and the function of file.write is writing.

4.2 User program

The programming skill is mainly the Linux multi-thread. First, the multi-thread is a “thrifty” multi-task operation mode. In the Linux system (Stevens, 2005; Matthew and Stones, 2007), it must assign an independent address space for starting a new process. In addition, system must establish a large number of data-sheets to safeguard its code, stack and data. On the contrary, it request far less time and space for starting a new multi-thread process. Second, the inter-threads communication is more convenient than the inter-processes. The different processes have independent data space and they only can transfer data with each other. At the same time, the different threads share the same space and they can directly access the data with each other.

4.2.1 Serial communication thread (Li, 2002)

Serial Communication thread is mainly used to realize the Ninth Five-year Plan seismic precursory instrument communication standards (Department of Monitoring and Forecasting, China Seismological Bureau, 2003), such as responding the receive data command form PC serial port. In addition, the design adds some custom command according to its own device's demand, such as the parameter of filter Q value. The thread creates by the following code:

```

res = pthread_create(&com_thread,NULL,
com_function,(void *)message)

```

4.2.3 Measurement Thread

The measurement thread is used to complete sample, feedback, calculate, show and save the result.

4.2.4 Socket Thread

Socket thread (Wang and Zhong, 2003) is mainly

used for achieving the Tenth Five-year Plan's rules. Through the socket thread, the equipment's data can directly save in the database. Moreover, this thread is responsible for checking the equipment's time with SNTP.

The above-mentioned threads work at the same process. They can use the global variables or file for data sharing. In order to saving memory and reducing the CF card mount times, the small-scale and common-use data use the global variable to share (Tan, 1993). In this way, it is not necessary that mount the CF card when remote access. Otherwise, the large-scale, no common-use or requiring higher security data will be shared in files (Stevens, 2005). First, it is not allowed to apply for a large block of memory because the embedded resource restrict. Second, it can ensure the data continuity even if the unusual procedure.

5 Summary

Due to the key circuits are VHDL documents and processor is the soft-core NIOS II CPU, the hardware maintenance changes into the software perfecting process, and accordingly it can extend the instrument's life cycle and is extreme convenience to its maintenance and upgrade. It has a lot of extended space of the equipment, for example, the whole hardware resource only accounts for less than half room of the FPGA chip, and even it can be configured into a multi-CPU high-intelligence instrument. The design will greatly enhance the stability of equipment and the security of data for using a stable embedded Linux operating system, taking ROMFS as system file system and dynamic data storage.

The study does not involve too much about the proton precession magnetometer sensor, but it achieves obvious improvement in intelligent-level, stability, easy maintenance of the equipment. In addition, EDA is a general technology that can realize the existing instruments' update after simple change. There are a number of further study after the design, for example, the signal high-power low-noise amplification problem, U-boot bootstrap, pass parameters from bootstrap to system and so on.

Acknowledgements This study is supported by Public Benefit Research Sector (No. H200708045); Basic R&D Operations Special (Nos. ZDJ2008-01 and J220780804). We would like to thank Professors Huizhong Chen and Bing Xue for their help to complete successfully this study.

References

- Altera (2008). *Introduction to the Quartus II Software*. Altera corporation, 1–247.
- Department of Monitoring and Forecasting, China Seismological Bureau (2002). *Earthquake Electromagnetic Observation Technology*. Seismological Press, Beijing, 75–110 (in Chinese).
- Department of Monitoring and Forecasting, China Seismological Bureau (2003). *Earthquake Precursor Observation Technology and Public Network*. Seismological Press, Beijing, 240–327 (in Chinese).
- Jiang X S and Liu D S (2004). *Hardware Description Language (VHDL) Course*. Xi'an Jiaotong University Press, Xi'an, 371–417 (in Chinese).
- Jonathan Corbet, Alessandro Rubini and Greg Kroah-Hartman (2005). *Linux device drivers*. O'Reilly Media, Inc USA, 46–285.
- Li X Y (2002). *Visual C++ Serial Communication Technology and Engineering Practice*. Posts & Telecom Press, Beijing, 1–427 (in Chinese).
- Liu Z R (2005). *Detailed Application Development Embedded Linux*. Machinery Industry Press, Beijing, 53–448 (in Chinese).
- Liu Z L (2000). *Linux/Unix Rapid Entry*. China Youth Press, Beijing, 90–133 (in Chinese).
- Matthew N and Stones R (2007). *Beginning Linux Programming*. Wrox Press, USA, 1–605.
- Pan S and Huang J Y (2002). *EDA Technologies and Practical Course*. Science Press, Beijing, 16–357 (in Chinese).
- Tan H Q (1993). *C Programming*. Tsinghua University Press, Beijing, 268–291 (in Chinese).
- Wang X P and Zhong J (2003). *Analysis of Network Communication and Application in Visual C++*. Posts & Telecom Press, Beijing, 22–124 (in Chinese).
- Wang X P (1995). *Integrated Phase Locked Loop Circuit*. Posts & Telecom Press, Beijing, 1–408 (in Chinese).
- Stevens W R (2005). *Advanced Programming in the UNIX(R) Environment*. Addison-Wesley Professional, USA, Indiana, 1–536.