

DESIGN OF EPICS IOC BASED ON RAIN1000Z1 ZYNQ MODULE*

Tao Xue, Hongming Li, Guanghua Gong, Jianmin Li, Tsinghua University, Beijing, China

Abstract

ZYNQ is the new architecture of FPGA with dual high performance ARM Cortex-A9 processors from Xilinx. A new module with Giga Bit Ethernet interface based on the ZYNQ XC7Z010 is development for the High Purity Germanium Detectors' data acquisition in the CJPL (China JinPing under-ground Lab) experiment, which is named as RAIN1000Z1. Base on the nice RAIN1000Z1 hardware platform, EPICS is porting on the ARM Cortex-A9 processor with embedded Linux and an Input Output Controller is implemented on the RAIN1000Z1 module. Due to the combine of processor and logic and new silicon technology of ZYNQ, embedded Linux with TCP/IP sockets and real time high throughput logic based on VHDL are running in a single chip with small module hardware size, lower power and higher performance. We will introduce how to porting the EPICS IOC application on the ZYNQ based on embedded Linux and give a demo of IO control and RS232 communication.

INTRODUCTION OF RAIN1000Z1

ZYNQ [1] is developed by Xilinx for high performance real time processing and embedded application. In the chip of ZYNQ, dual core of ARM Cortex-A9 processor with rich peripherals is the half of the chip, the other half part is the FPGA. People can develop the embedded application in the processor with C programming, such as embedded Linux and uC/OS or VxWorks real time OS can be running on the Cortxe-A9 dual core processors. In the other part of FPGA, people can develop the real time processing logic or DSP function with VHDL/Verilog programming. It is amazing that combine the two parts in one chip, the total footprint is more compact, board space is saved and the power is saved. At the same time, the assistive circuit, such as the on board power circuit, supervisor circuit and debug circuit is simpler.

With this feature of ZYNQ, we develop a readout module for the High Purity Germanium Detectors' data acquisition in the CJPL (China JinPing under-ground Lab) experiment, which is named as RAIN1000Z1 [2]. Figure 1 shows the system architecture of the RAIN1000Z1 module.

The RAIN1000Z1 module is based on the XC7Z010 chip of ZYNQ series, in the CLG225 footprint, which is the smallest package of ZYNQ series. A 256Mbytes DDR3 SDRAM running at 1066MHz and 1000Mbps Ethernet PHY chip 88E1512 from Marvell Inc. are embedded in the RAIN1000Z1 module. Also the power circuit and Flash for firmware saving in QSPI interface are

all integrated in the module. Figure 2 is the top view of the RAIN1000Z1 module.

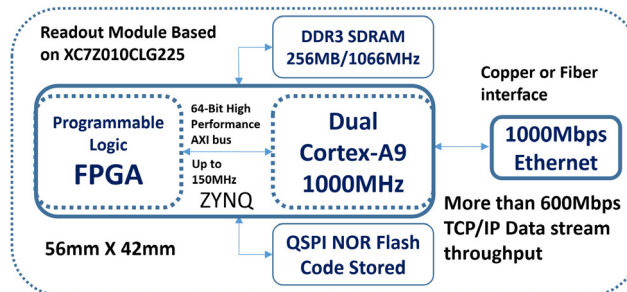


Figure 1: Architecture of the RAIN1000Z1 module.

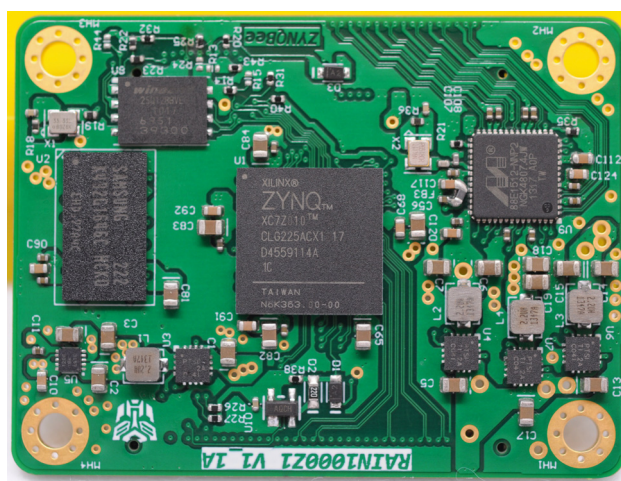


Figure 2: Top view of the RAIN1000Z1 module.

China JinPing underground Laboratory (CJPL) is a deepest underground in the world and provides a very good environment for direct observation of dark matter.

The CDEX experiment will going to direct detect the interaction of WIMP with nucleon in CJPL with high sensitivity in low mass region. Both CJPL and CDEX have got much more progress in recent two years. The CDEX use point contact germanium semi-conduct detector PCGe which has less than 300eV threshold.

The RAIN1000Z1 is used in the readout for HPGe detectors for the CDEX (China Dark Matter Experiment) in CJPL, we can easily get more than 600Mbps TCP/IP data throughput [2].

In the RAIN1000Z1 module, the embedded Linux is running. Currently, we use the version of 3.19 from Xilinx support git hub. Figure 3 shows the booting log of embedded Linux.

Some GPIO and one UARTlite (use the IP core from Xilinx in the Vivado development environment) are im-

*The authors are with Key Laboratory of Particle & Radiation Imaging, Department of Engineering Physics, Tsinghua University, Beijing, China. 100084 (email: gbe.tao.xue@gmail.com).

plemented in the FPGA part of ZYNQ, and the associated drivers are developed for the embedded Linux.

EPICS PORTING

The EPICS (Experiment Physics and Industrial Control System) is a software environment used to develop and implement distributed control systems to operate devices such as particle accelerators, telescopes and other large experiments [3]. In the next generation of CJPL, which is named CJPL-2, there are more complex and space to develop many physics experiments, we want to use the EPICS as the distributed control system. Then we porting the EPICS on the RAIN1000Z1 and implement an IOC based on the RAIN100Z1 module.

The EPICS software system can running on the Linux OS, and there are many engineers, researchers install the EPICS on Linux [4]. These give a good reference for our application.

```
Booting Linux on physical CPU 0x0
Linux version 3.19.0-xilinx (root@ubuntu) (gcc version 4.9.1 (Sourcey CodeBench
Lite 2014.11-30) ) #11 SMP PREEMPT Mon Oct 12 09:01:35 PDT 2015
CPU: ARMv7 Processor [413fc090] revision 0 (ARMv7), cr=18c5387d
CPU: PIPT / VIPT nonaliasing data cache, VIPT aliasing instruction cache
Machine model: xlnx,zynq-7000
cma: Reserved 16 MiB at 0x0f000000
Memory policy: Data cache writealloc
PERCPU: Embedded 9 pages/cpu @4edd3000 s8128 r8192 d20544 u36864
Built 1 zonelists in Zone order, mobility grouping on. Total pages: 65024
Kernel command line: console=ttyPS0,115200 root=/dev/ram rw earlyprintk
PID hash table entries: 1024 (order: 0, 4096 bytes)
Dentry cache hash table entries: 32768 (order: 5, 131072 bytes)
Inode-cache hash table entries: 16384 (order: 4, 65536 bytes)
Memory: 231164K/262144K available (4557K kernel code, 268K rwdata, 1608K rodata,
216K init, 210K bss, 14596K reserved, 16384K cma-reserved, 0K highmem)
```

Figure 3: RAIN1000Z1 Linux booting.

With the embedded Linux, it is not difficult to install the EPICS on RAIN1000Z1 module. The EPICS is a set of software, there are two way to install it on the embedded Linux of RAIN1000Z1. The directly way is compile the EPICS with the gcc compile within the RAIN1000Z1 file system and the other way is cross compile with the arm cross gcc on the Linux (such as Ubuntu) running on a PC [5].

For directly compile in RAIN1000Z1, we need a rootfs with gcc for ZYNQ, and more time is needed, for example, maybe 50 minutes is needed, because the ARM Cortex-A9 is not fast than a CPU in PC. But this way is very simple, we do not need to change the setting of cross compile environment to match the EPICS's compile, although the cross compile is so fast in the PC. Then we use the directly compile in the RAIN1000Z1 module. And a more complete rootfs is built for use with Yocto, the gcc compiler and python support are included.

For the version of EPICS, we use a stable version of EPICS Base 3.14.12.5 (more other version is used, but not described in this paper).

Step 1, Download the EPICS Base (it can be download in PC or directly by RAIN1000Z1).

```
wget http://www.aps.anl.gov/epics/download/base/base3.14.12.5.tar.gz
```

```
tar xzf base3.14.12.5.tar.gz
```

Step 2, Building the EPICS Base. EPICS's build is directed by the EPICS_HOST_ARCH environment variable, the following command is used to set this variable in embedded Linux running on RAIN1000Z1.

```
export EPICS_HOST_ARCH=$(/home/epics/base base-3.14.12.5/startup/EpicsHost
Arch)
```

After setting, this command can check the value:

```
echo $EPICS_HOST_ARCH
```

The "linux-arm" echoed show that EPICS base will be built for Linux OS on ARM architecture which is the Cortex-A9 core in ZYNQ. Then we can build the EPICS Base with the following command:

```
cd ./base-3.14.12.5
```

```
make
```

In the platform of RAIN1000Z1, about 40 minutes is needed for building.

Step 3, Start the EPICS Base

After the building, it's possible to start an EPICS IOC, we also implement some GPIO function for IO control and a RS232 port for communication demo. Start EPICS as the following:

```
./bin/linux-arm/softIoc
```

And EPICS will start with a basic command line prompt:

```
epics>
```

type iocInit and we will get the following:

```
Starting iocInit
```

```
#####
```

```
## EPICS R3.14.12.5 $Date: Tue 2015-03-24 09:57:35 -0500$
```

```
## EPICS Base built Oct 12 2015
```

```
#####
```

```
iocRun: All initialization complete
```

```
epics>
```

And the EPICS Base has been built on embedded Linux of the RAIN1000Z1.

More test is done with the synApps, PyEpics and a GUI window is developed in Ubuntu for IO and RS232 test. The graphical user interface was developed using the EPICS CA client CSS (Control System Studio). And the user interface is execute on a remote PC. The function is running well and more peripherals can be added in the RAIN1000Z1 platform.

CONCLUSION

EPICS is porting on the embedded Linux within the ARM Cortex-A9 processor which is included in the RAIN1000Z1 module. It's suitable to develop more flexible interface and function with this platform. More software in the EPICS software set are under test on the RAIN1000Z1 module and the new result will be show.

REFERENCES

- [1] www.xilinx.com/zynq
- [2] Tao Xue et al., "Design of Giga Bit Ethernet Readout Module Based on ZYNQ for HPGe", Real Time Conference (RT), 2014 19th IEEE-NPSS Nara, Japan (2014).
- [3] www.aps.anl.gov/epics/
- [4] prjemian.github.io/epicspi/
- [5] wiki-ext.aps.anl.gov/epics/index.php