

# APPLICATION OF MODBUS-TCP IN TPS CONTROL SYSTEM

Y. K. Chen, Jenny Chen, Y. S. Cheng, C. Y. Wu, P. C. Chiu, C. H. Kuo, K. H. Hu, K. T. Hsu  
NSRRC, Hsinchu 30076, Taiwan

## Abstract

Modbus-TCP is a widely used in industry for a long time and slow-control devices in accelerator control system recently. Modbus protocol over Ethernet has advantages for non real-time applications due to its maturity. The TPS (Taiwan Photon Source) project will have some Modbus-TCP enabled devices which are distributed in utility facilities and accelerator system. The accelerator control environment of TPS project is an EPICS based system. Modbus-TCP might be adopted for some subsystems. There are several possible Modbus-TCP available devices: the prototype power supply for magnet field mapping application, vacuum system local controller, beam line front-end controller, and some monitoring devices. In this paper, we will summarize preparation efforts to accommodate the Modbus-TCP support in the TPS control system.

## INTRODUCTION

Modbus-TCP [2] (also called Modbus-TCP/IP) is simply the Modbus RTU protocol with the TCP interface running on Ethernet. It uses TCP/IP and Ethernet to carry the data of Modbus messages between compatible controllers. In other words, the message of Modbus-TCP is encapsulated with an Ethernet TCP/IP frame. The communication of Modbus-TCP is a client server model between devices. A Modbus device may support a client and/or a server Modbus interface. The interface includes input discrete, output discrete (coil), input registers and output registers functions [5].

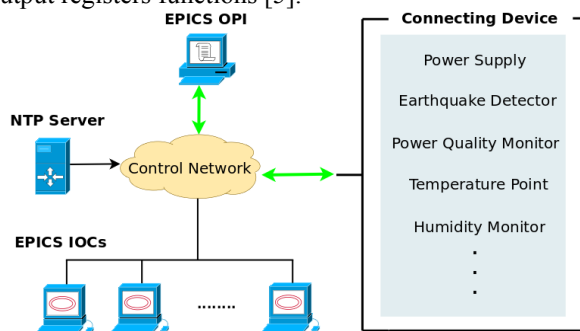


Figure 1: The structure of Modbus control system.

The structure of control system is shown in Fig 1. There are several components in this structure, such as power supply, earthquake detector, NTP server, EPICS IOC and OPI (Operator Interface) with Graphical User Interface (GUI).

In addition, the communication protocols TCP and RTU (Remote Terminal Unit) are both supported. These could be applied for various connection methods such as direct connection or broadcasting. In order to capture the

event data accurately, RTC (Real-Time Clock) with NTP (Network Time Protocol) is used to apply the event time stamp. So far as the above functionalities are concerned, it is the general infrastructure of the Modbus control system. In the following paragraph, we will briefly introduce each component orderly.

The Hezemeyer power supply (PS) is controlled by “IOCASTE-Control System” which is compact and has better integration of different elements of a close loop. It supports several standard protocols such as Profibus RS-485, Modbus RS-485 and Modbus TCP/IP, but Modbus-TCP is selected to simplify the communication medium to Ethernet.

Vibration has become an important issue which should be considered to integrate into the accelerator control system in recent years. The earthquake detector is an advanced seismic switch, including tri-axial MEMS accelerometer and powerful 16 bit 80 MHz CPU. The most important feature is the STA/LTA earthquake detecting algorithm which has been embedded in the instrument. In contrast with the traditional earthquake detecting algorithm, it applies the numerical technique and outputs the data stream (vector calculation, earthquake intensity and zero drifting compensation... etc) in real-time.

At GUI part, both of MATLAB and EDM are chosen for the user interfaces. In order to process the complex computing and capture the event data automatically, MATLAB supports not only monitor but analyze the earthquake information from the detector. For the GUI of a PS, EDM supports enough widgets to send commands and monitor the status from itself. In GUI section, more details for these two interfaces will be introduced.

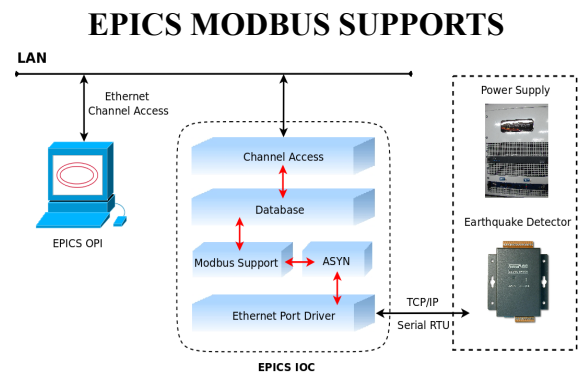


Figure 2: The block diagram of Modbus support in EPICS IOC.

By using the EPICS framework [1][7], Fig 2 describes the relationship between the EPICS IOC and other devices. EPICS IOC communicates with the devices through TCP/IP or serial RTU. Then GUI client console

grabs the data via EPICS channel access and display on the screen. In EPICS IOC block, we focus on the interaction between Modbus support and ASYN module.

Modbus Support is a protocol application which is based on ASYN [4]. ASYN is a general purpose module which responds the low level driver for target devices. Moreover, the asynDriver could solve the synchronous problem by its blocking and non-blocking methods of the registered ports. The Modbus module unpacks the packet from Ethernet Port Driver then passes the data to Modbus Support block. After the Ethernet header is taken off (includes Ethernet, TCP and IP), the Modbus support extracts the data from Modbus ADU (Application Data Unit). According to the EPICS database definition, the data can be mapped into Boolean states or analog values. The CA client with GUI provides user interface and the channel status, command buttons, waveform charts, screen-shots and history archives.

### SOME MODBUS-TCP APPLICATIONS

As mentioned previously, Modbus-TCP protocol will be used for several subsystems of TPS project. In this paper, the control of the power supply and earthquake detector includes EPICS IOC and GUI. Both IOCs adopt asynDriver of EPICS to communicate with Modbus-TCP, but their methods are different due to the non-standard data format of the power supply. The detail will be explained in the following paragraph.

#### Modbus-TCP Power Supply

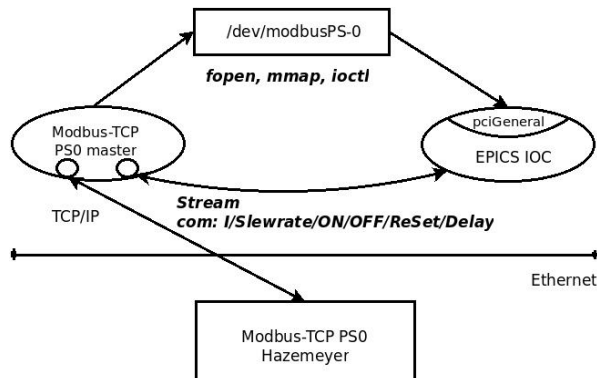


Figure 3: The flow diagram describes the relationship between Modbus-TCP and pciGeneral in EPICS IOC.

Several power supplies made by Hazemeyer HH were purchased for magnet field mapping application. The internal control of the PS includes a numerical regulation with 18bits resolution and interacts with Modbus-TCP. The data format of current and slew-rate control commanded remotely is IEEE-754 32-bit floating-point which is not a standard format of Modbus PDU (protocol data unit). It needs two transactions of holding register control sending by a master to perform a current or slew-rate setting which is another part of incompatible with the standard. It was the reason Modbus support [1] was not adopted to control the PS under EPICS like earthquake detectors. A multi-threaded program is instead developed

to handle the communication between EPICS and PS due to only one master is allowed to control it. One thread waits for current, slew-rate or time period of data update commands from EPICS via “stream” device support, and another thread sends data update request to the PS periodically. A mutex is created to manage the resource, TCP connection to the PS, between two threads. The data update thread flushes the received data into the memory created by a char device driver where can be accessed by EPICS IOC directly also. The process flow is shown in Fig 3.

A device support called pciGeneral base on asynDriver was developed to access cPCI cards directly from EPICS. The data interface to EPICS includes floating-point data type (asynFloat32) which is not used by PCI interface but used for data interchange between processes. The purpose of the pciGeneral is not only for accessing cPCI cards but also for inter-process communication. It is to support as more function as possible by one device support for EPICS. A simple kernel driver is installed to create a char device that may be accessed by file operation utilities of Linux OS. The interface of the char device includes memory map function that maps the kernel space memory into user space. It increases the performance of data access by EPICS compared to file operations of the char device. Since there is no hardware interrupt for this case, after the data receiving thread flushes data, it may trigger the kernel driver via ioctl to request OS to wake up the process which is waiting for the signal. The SCAN field of the PV records of EPICS database may be set to “I/O Intr” to update data synchronously.

For power supply, the EDM page displays the voltage and current value in real-time, and provides the current/slew-rate control fields with ON/OFF/RESET push buttons. The main GUI of earthquake detector was developed under MATLAB, it shows the scale intensity, UTC time, waveform chart and automatic mode (screenshot and save the received data).

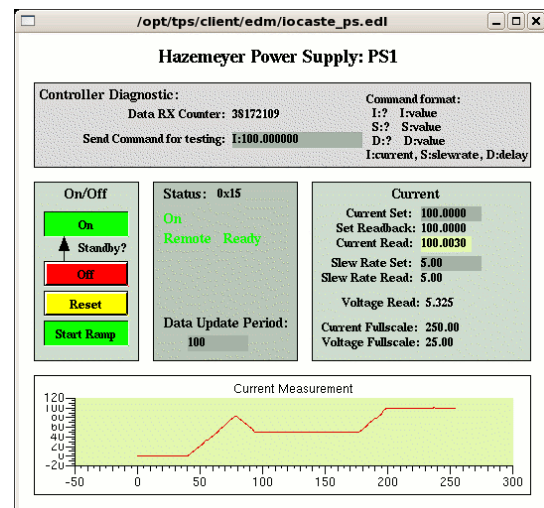


Figure 4: The EDM GUI of the power supply.

The PS control panel shown in Fig 4 is created by EDM. There is an interrupt counter and a command field

in the diagnostic area, a text command can be sent to the multi-threaded process through a port of “stream support” of EPICS to test the process and the Modbus PS. The reply sent by the process shows the result of the control or the current status of the PS. The interrupt counter is the number of the data block received by the data upload thread and updated by the pciGeneral support under EPICS with interrupt manner.

About the ramp control, the PS needs a ramp command to charge current according to the current setting and the slew-rate. The ramp command can be sent right after the current setting by the database operation of EPICS according to the FLNK field of the current control PV.

### Modbus-TCP Earthquake Detector

According to the previous sections, EPICS Modbus support module [1] does not only include the definitions of the holding register read-back (3) and multiple registers output (16) function codes but also a set of DB template files. The *asynInt32* and *asynInt32Array* are used for analog and waveform records. Then the information of earthquake detector can be correctly extracted. To perform periodically data update, *poll delay* record determines the reading or writing delay time in polling threads. With pseudo-interrupt functions, the polling thread updates EPICS records when new data is available.

The GUI of earthquake detector provided by the manufacture is truly friendly that it combines the real-time plotting and history trends. However, the disadvantages are: it needs large amount of resources and cannot be used for correlated with other sub-systems.

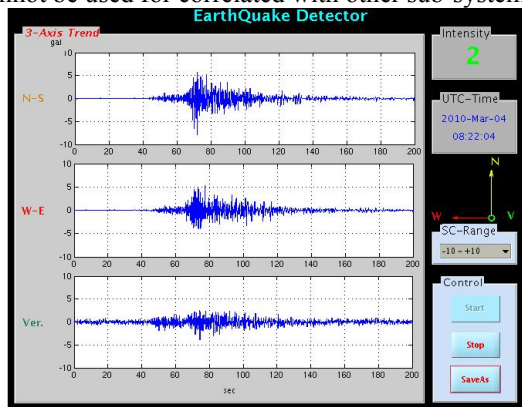


Figure 5: The popup MATLAB GUI when earthquake was detected.

A specific mode for the earthquake detector, automatic mode, is to save the system resources for archiving earthquake events. To store the data record at once, the following mechanism was developed. In monitor mode, incoming waveform of 3-axis will be inserted into three ring buffers individually. Once the earthquake or vibration detected, subroutines are called to log the current screenshot and waveform of 3-axis. Compared with archiving forever, automatic mode does not only decrease the read/write loops, but also reduce the disk space. Besides, the CPU load of the console computer

which runs MATLAB GUI is acceptable. The average CPU load is 2~2.5% when the scan rate is set to 50 Hz and the update rate is 1 Hz. Moreover, the size of archive file is less than 100 kB including screenshot and 3-axis waveform with time stamp.

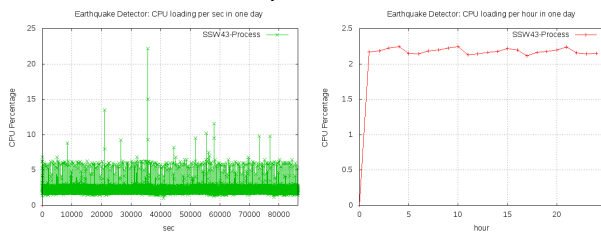


Figure 6: CPU loading of earthquake detector GUI.

In addition, Fig 5 shows the 3-axis waveforms of the earthquake which had been detected at 08:19:26 on March 4. Based on the information from Central Weather Bureau (CWB), the earthquake is happened at 08:18:53 in Kaohsiung. For post-mortem diagnostic, it becomes useful to combine the received data with BPM retrieval for analysis [6].

## SUMMARY

In this report, we had not only improved the connection performance of power supply under EPICS, but also provided a convenient and automatic monitor GUI for the earthquake detector. Nowadays, more and more instruments or devices support Ethernet as the field bus for control remotely, thus they can be closer to the control targets or signal sources to provide better signal qualities and services. The Modbus-TCP is easily to integrate several kinds of devices and it is license free. Besides, it becomes a connection option which would be adopted by some TPS subsystems, such as the vacuum and the machine operation interlock. These experiences are very helpful for constructing them in the future.

## REFERENCES

- [1] Driver Support for Modbus Protocol under EPICS – Release 1.3, Mark Rivers, September 19, 2008.
- [2] Introduction to Modbus TCP/IP, Acromag Technical Reference – Modbus TCP/IP, 2005.
- [3] EPICS Record Reference Manual – Release 3.13, Philip Stanely, Janet Anderson, Marty Kraimer, 1998.
- [4] asynDriver: Asynchronous Driver Support – Release 4.12, Marty Kraimer, Eric Norum, Mark Rivers, August 19, 2009.
- [5] Modbus Messaging On TCP/IP Implementation Guide Rev 1.0, Modbus Community, May 8, 2002.
- [6] C. H. Kuo., et al., “Diagnostics Update of the Taiwan Light Source”, Proceedings of the Beam Instrumentation Workshop 2010.
- [7] EPICS: Input / Output Controller Application Developer's Guide – Release 3.14.10, Martin R. Kraimer, Janet Anderson, Andrew Johnson, Eric Norum, Jeff Hill, Ralph Lange, Benjamin Franksen, January 5, 2009.