

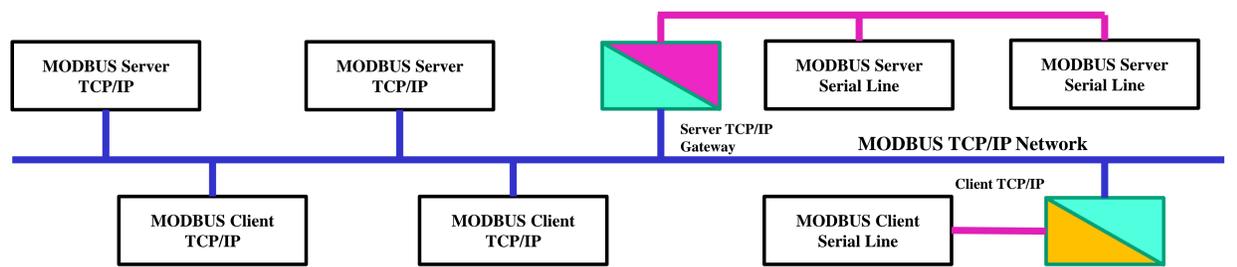
MODBUS APPLICATIONS AT JEFFERSON LAB*

J. Yan, C. Seaton, S. Philip, Thomas Jefferson National Accelerator Facility, Newport News, VA 23606, USA

Abstract

Modbus-TCP is the Modbus RTU protocol with the TCP interface running on Ethernet. In our applications, an XPort device utilizing Modbus-TCP is used to control remote devices and communicates with the accelerator control system (EPICS). Modbus software provides a layer between the standard EPICS asyn support and EPICS asyn for TCP/IP or serial port driver. The EPICS application for each specific Modbus device is developed and it can be deployed on a soft IOC. The configuration of XPort and Modbus-TCP is easy to setup and suitable for applications that do not require high-speed communications. Additionally, the use of Ethernet makes it quicker to develop instrumentation for remote deployment. An eight-channel 24-bit Data Acquisition (DAQ) system is used to test the hardware and software capabilities.

Modbus TCP/IP Communication Architecture



- * Modbus is a Client/Server Communication Model
- * The Modbus TCP messaging service provides a Client/Server communication between devices connected on an Ethernet TCP/IP network

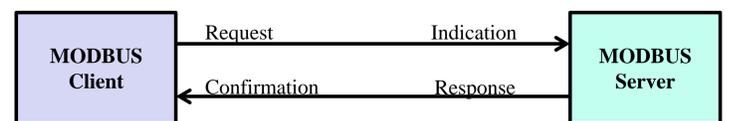
XPort and Eval Board



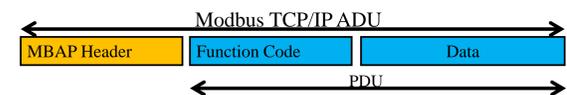
Key Features:

- Complete, Integrated Solution in an RJ45 Form Factor
- 10/100 Mbit Ethernet Connection
- Embedded Web Server
- Full TCP/IP Protocol Stack
- High-performance Processor
- Reliable, Proven Operation System
- Serial-to-10/100 Ethernet Conversion
- 300 to 921,600 Baud Rate Data Transfer
- Evaluation Board Provides a Simple, Quick, and Cost-effective Way to Develop Application
- Software DeviceInstaller to Configure the XPort

Modbus Messages



Modbus TCP/IP Request/Response

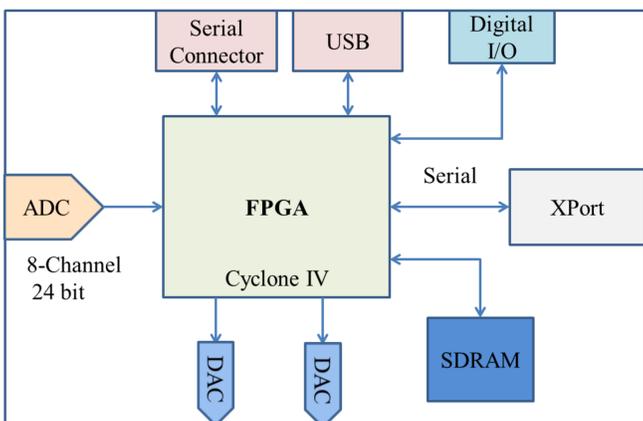


Application Data Unit (ADU) = Modbus Application Protocol (MBAP)+ Protocol Data Unit (PDU)

Reading Holding Register ADU:

Transaction ID	Protocol ID		Length		Unit ID		Function Code	Starting Address		Quantity of Registers	
Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8	Byte 9	Byte 10	Byte 11
Hi	Lo	Hi	Lo	Hi	Lo	1 byte	1 byte	Hi	Lo	Hi	Lo
0x01	0x00	0x00	0x00	x00	0x06	0x01	0x03	0x00	0x00	0x00	0x01

Modbus Data Acquisition Board



Features:

- A Cyclone IV FPGA
- A 8-Channel 24-bit ADC
- Two DACs
- SDRAM Memory for Data Buffer
- A Serial Interface
- Digital I/O Connector
- USB Connector
- XPort

EPICS Modbus Module

Layer 1: EPICS Asyn Device Support (asyn)

Layer 2: EPICS Asyn Port Driver

Layer 3: Asyn Interpose Interface

Layer 4: Asyn Port Driver

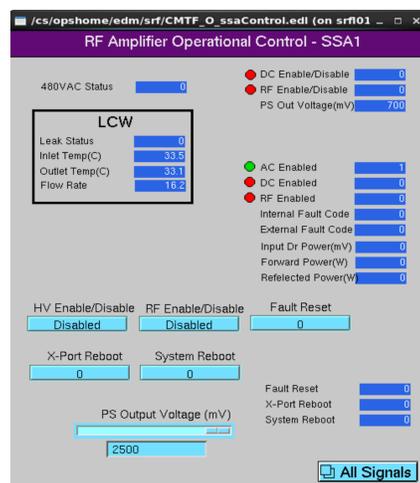
Procedure to Create a Modbus Port Driver:

- Run `drvAsynIPPortConfigure("ssa1", 129.57.200.226:502", 0,0,1)` to Create an asyn IP Port Driver "ssa1".
- Run `modbusInterposeconfig("ssa1", 0, 2000, 0)` to Configure the Interpose Driver.
- Call `drvModbusAsynConfigure("RF_In", "ssa1", 0, 3, 1,125, 0, 100, "RK")` to Create a Modbus Port "RF_In".
- Call `dbLoadTemplate("ssa1.substitutions")` to Load Database.

RF Amplifier Control Application

Modbus Protocol are Applied to Control the Cryomodule RF Amplifiers for the CMTF Project at Jefferson Lab:

- 8 Solid State RF Amplifiers
- High RF Power at 1.3 GHz
- Make Modbus Module Application
- Make SSA Modbus Application
- Run Soft IOC
- Load Database for all SSAs
- Create Operation Screens for all SSAs
- Function Code 0x06 to Write 6 Registers
- Function Code 0x03 to Read over 500 Registers



Conclusions

The Modbus data acquisition board, based on the XPort and Modbus TCP/IP protocol, has been prototyped. The board can be used in various applications that require remote communications. The Modbus driver support for Modbus protocol under EPICS is installed and applied for the cryomodule RF amplifier control system. The configuration of XPort, Modbus TCP/IP, and the EPICS Modbus package is easy to setup and suitable for applications that do not require high-speed communications.