Modbus and DNP3:

Comparing Communication Efficiencies

Control Microsystems White Paper **September 2009**

The purpose of this paper is to compare the efficiency of two SCADA protocols: Modbus and DNP3. By utilizing DNP3 it is possible to significantly reduce bandwidth on your communication channels, allow more devices to be added to your system (i.e. scalability), and add new functionality to devices, such as time stamping.

CONTROL MICROSYSTEMS
SCADA products...
for the distance

Author: David Bevin

Introduction

The purpose of this paper is to compare the efficiency of two SCADA protocols: Modbus and DNP3. We will take a hypothetical device and calculate the number of bytes required on the communication channel to retrieve data for a specific number of points.

This paper does not consider vendor-specific commands that may be used to collect historic data in a more efficient manner. This ensures a fair comparison that would work equally well with a product from any vendor.

This paper will show that by utilizing DNP3 it is possible to:

- Significantly reduce bandwidth on your communication channels,
- Allow more devices to be added to your system (i.e. scalability), and
- Add new functionality to devices, such as time stamping.

The Device

The hypothetical device has:

- 32 digital inputs, and
- 16 analog inputs.

Requirements

The customer has the following requirements for the device:

1. The host must log the digital and analog changes with a timestamp accurate to the nearest 10 seconds.
2. Digital changes need to be reported to the user within one minute of the occurrence.
3. Analog changes need to be reported to the user within 10 minutes of the occurrence.

Assumptions

For the purposes of this working example, it has been assumed that:

1. There will be a total of 128 digital input changes every hour.
2. There will be a total of 80 analog input changes every hour. A change could be when the analog reaches a pre-defined limit (set-point) or when the input changes significantly (i.e. 5% of full scale).
3. No communication packets are dropped (i.e. perfect communications).

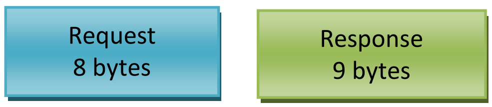
Modbus – Method 1

The system is initially configured with 32 status registers for the 32 digital inputs, and 16 input registers for the 16 analog inputs.

To meet requirement 1, the polling interval must be set to 10 seconds. Requirements 2 and 3 are also met because the polling interval is set to 10 seconds.

Status Registers

The 32 status registers will be retrieved using Modbus function code 2.



This gives a total of 17 bytes every 10 seconds.

Input Registers

The 16 input registers will be retrieved using Modbus function code 4.

This gives a total of 45 bytes every 10 seconds.

Total

Combining the sizes for status registers and input registers gives a total of 62 (i.e. 45 + 17) bytes every 10 seconds. Calculating this over a period of one day gives:

62 bytes per poll x 6 polls every minute x 60 minutes x 24 hours = 535,680 bytes

Therefore 535,680 bytes of data on the channel are required to meet the requirements.

Modbus – Method 2

The system is modified to pack the 32 digital inputs into 2 input registers (16-bit) that immediately follow the analog inputs. This gives a total of 18 input registers.

To meet requirement 1, the polling interval must be set to 10 seconds. Requirements 2 and 3 are also met because the polling interval is set to 10 seconds.

Input Registers

The 18 input registers will be retrieved using Modbus function code 4.

This gives a total of 49 bytes every 10 seconds.

Total

Calculating the number of bytes on the channel over a period of one day gives:

49 bytes per poll x 6 polls every minute x 60 minutes x 24 hours = 423,360 bytes

Therefore 423,360 bytes of data on the channel are required to meet the requirements.

DNP3

The system has 32 digital input points and 16 analog input points. Analog points will be stored as 16-bit integers. Digital inputs will be transmitted as “packed” points (i.e. without flags) in static objects.

DNP3 supports timestamps for events. Therefore, requirement 1 will be met (and exceeded).

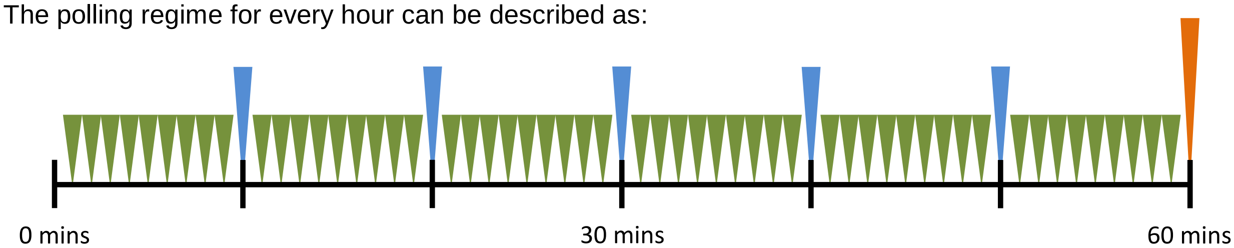
All 32 digital input point events have been configured to report in Class 1. To meet requirement 2, the Class 1 polling interval must be one minute.

All 16 analog input point events have been configured to report in Class 2. To meet requirement 3, the Class 2 polling interval must be 10 minutes.

To maintain integrity of data, an Integrity poll will be scheduled every hour.

Polling Regime

The polling regime for every hour can be described as:



Polls for Class 1 events are represented by the shortest (green) arrows, which occur every minute.

Combined Class 1 and Class 2 event polls are represented by the mid-sized (blue) arrows, which occur every 10 minutes.

The Integrity poll is represented by the largest (orange) arrow, which occurs every hour.

Integrity Poll

It is assumed that the response to the Integrity poll contains only static data (i.e. no events are waiting to be collected).

This gives a total of 100 bytes for an Integrity poll.

Calculating the number of bytes required for Integrity polls on the channel over a period of one day gives:

100 bytes per poll x 24 hours = 2400 bytes

Analog Events

It is assumed that 16 analog events will be reported in each of the five combined Class 1 and Class 2 event polls. This gives a total of 80 analog events per hour, which meets assumption 2.

It is assumed that the device reports the analog events in 16-bit objects with timestamp (Group 32 Variation 4).

This gives a total of 256 bytes per poll.

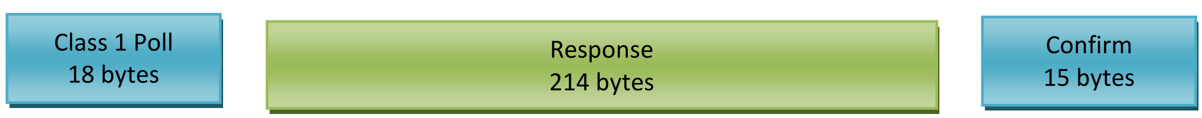
Calculating the number of bytes on the channel over a period of one day gives:

256 bytes per poll x 5 polls every hour x 24 hours = 30,720 bytes

Digital Events

It is assumed that 32 digital events will be returned in four of the Class 1 event polls, giving a total of 128 digital events every hour, which meets assumption 1.

It is assumed that the device reports the digital events with relative timestamp (Group 2 Variation 3) objects.



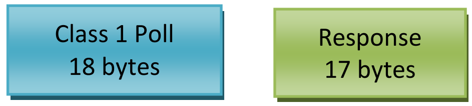
This gives a total of 247 bytes per poll.

Calculating the number of bytes on the channel over a period of one day gives:

247 bytes per poll x 4 polls every hour x 24 hours = 23,712 bytes

Empty (No Events) Polls

Because all of the digital events were returned in four polls, there will be 50 Class 1 event polls that do not contain events every hour.



This gives a total of 35 bytes per poll.

Calculating the number of bytes on the channel over a period of one day gives:

35 bytes per poll x 50 polls every hour x 24 hours = 42,000 bytes

Total

To calculate the total number of bytes on the channel over a period of one day we need to add together the calculations for the Integrity polls, analog event polls, digital event polls and the empty polls.

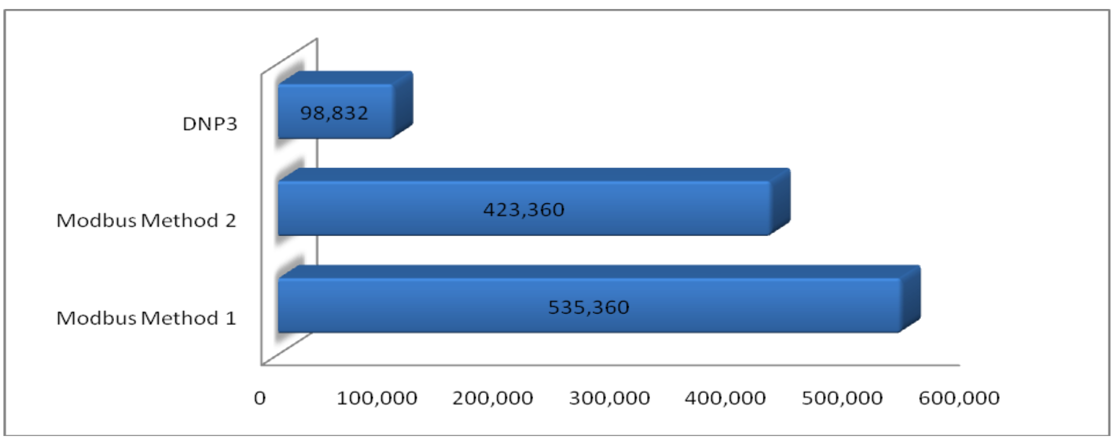
Calculating the number of bytes on the channel over a period of one day gives:

2400 bytes (Integrity polls) + 30,720 bytes (analog event polls) + 23,712 bytes (digital event polls) + 42,000 bytes (empty Class 1 event polls) = 98,832 bytes

Therefore 98,832 bytes of data on the channel are required to meet the requirements.

Summary

The following graph shows the number of bytes on the channel per day for the three methods described in this document.



In this example, DNP3 uses 23% of the bandwidth of the most efficient Modbus polling regime.

It should be noted that if the number of analog or digital events is less than specified, then the number of bytes required for DNP3 would be reduced but the number of bytes required for Modbus would remain the same.

In real-world terms, this is a significant reduction of bandwidth on the communication channel that could allow a larger number of devices to be used in the network without replacing the communication infrastructure. It also allows the spare bandwidth to be reliably used for remote maintenance of the devices (e.g. reconfiguration or firmware upgrades).

DNP3 offers many features that are not available in Modbus, including time-stamping of events at the remote device. These events, consisting of data and timestamp, are recorded in the device even if communications to the remote device fails. The events can then be collected at a later time, thereby ensuring that no data is lost.

About the Author

David Bevin has over 10 years of experience in the SCADA industry and has developed numerous protocol drivers including Modbus, DNP3 and IEC60870. David is a Lead Engineering Specialist for Control Microsystems and has worked as a developer for the ClearSCADA host platform and SCADAPack controllers.

Contact Us

Name: Sales Support

Address: Control Microsystems
Corporate Headquarters
48 Steacie Drive
Ottawa, Ontario, Canada,
K2K 2A9

E-mail: salessupport@controlmicrosystems.com

Tel: 1-888-CMSCADA (1-888-267-2232)
1-613-591-1943 ext. 254

Fax: 1-613-591-1022