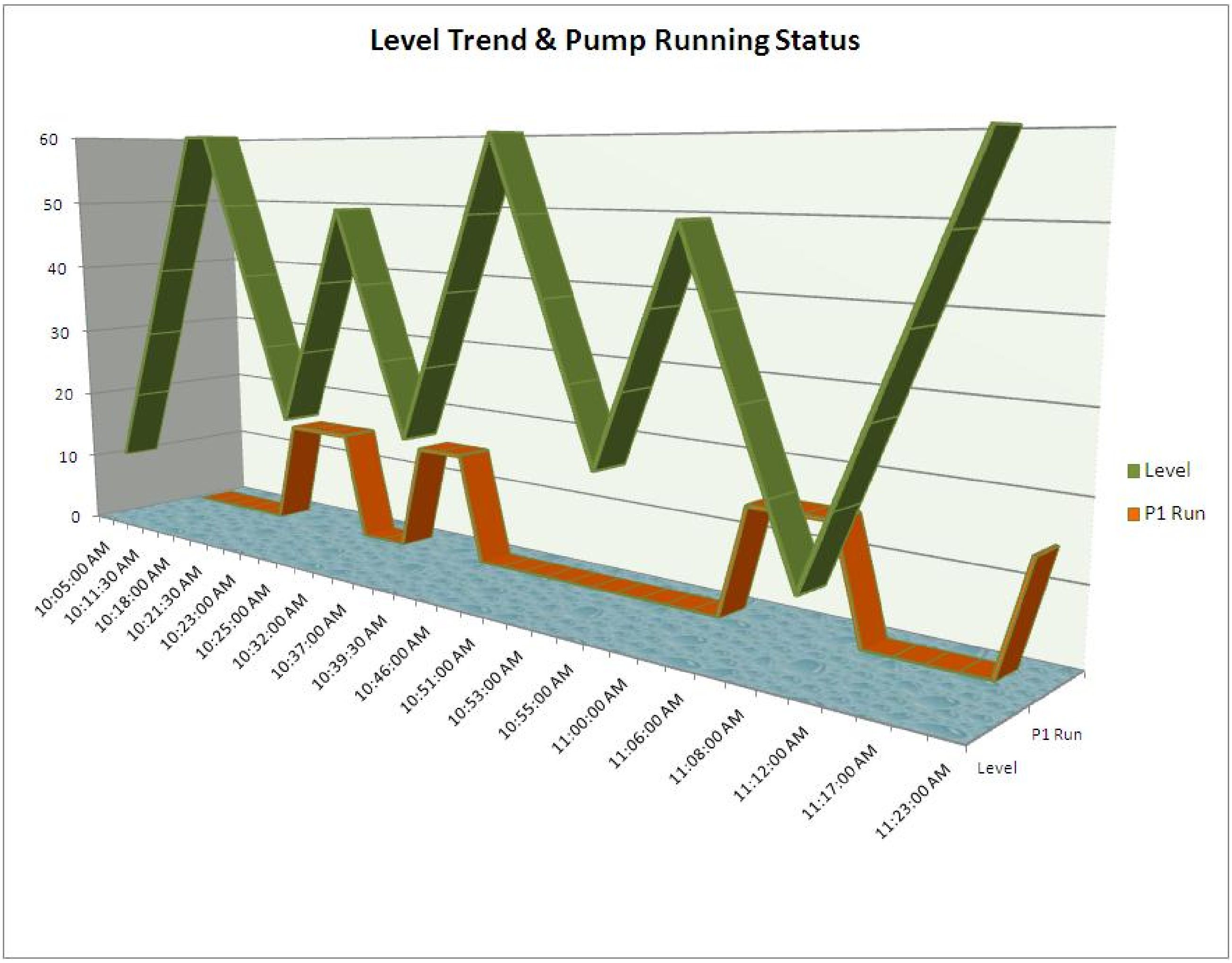
Benefits of a Modern SCADA Protocol

DNP3 vs Modbus

November 2007



Benefits of using modern protocols for SCADA

A protocol is simply the rules that govern the data transfer between parts of a control system. Different protocols have varying strengths and weaknesses.

This document looks at the ideas behind good protocols for the users of SCADA / telemetry systems. This is an area often neglected by users when first investing in a SCADA system, as they are often happy to get any data from remote stations. However, it is important to give protocol choice proper consideration.

DNP3 provides distinct advantages over Modbus and other protocols, even for the average user. Users will usually conclude that the advantages of DNP3 outweigh the additional initial investment.

If you don’t want to lose valuable data, or carry out extensive detective work to assess whether trends and reports are demonstrating system problems, station problems – or communication problems, then you need a protocol that reliably reports all changes from the field.

Data from the field as trended on the SCADA front end

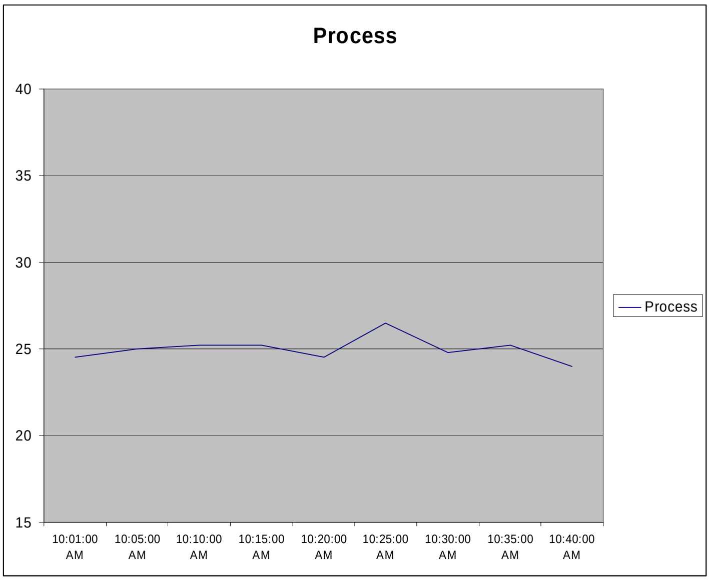


Figure 1 – Example 1

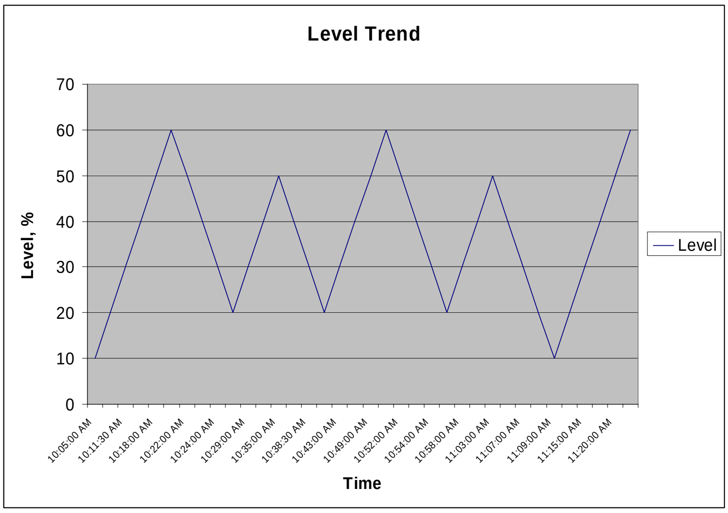


Figure 2 - Example 2

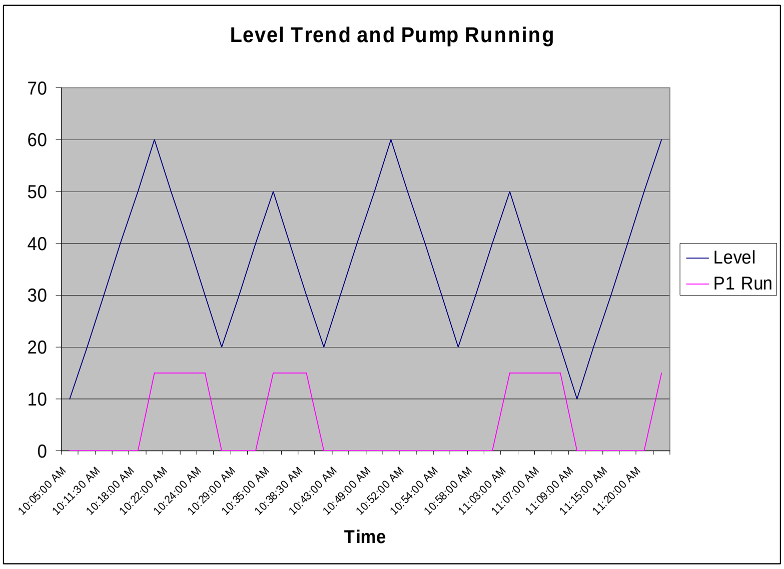
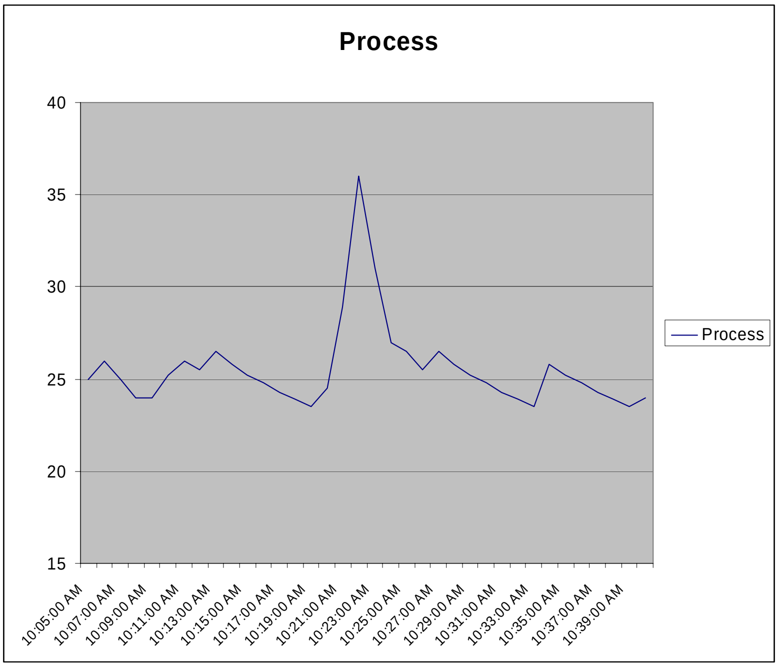


Figure 3 - Example 3

In example 1, the user might be happy with the view of the field data shown. In the other two examples, the user might start to look for the source of the problem.

Example 2 shows that the start and stop levels are not always the same, though they were expected to be. Is the control circuit working effectively? Are there delays which are in operation at certain times? In example 3, the user will be trying to determine how the level dropped while no pump was working.

In examples 2 and 3 the user might in the end consider SCADA communications as being the real problem – either the comms architecture or temporary radio problems.



Example 1 – The real data, compared with the sample above.

Example 1, the user might never realize that in fact the process parameter varied significantly. This might be an indication of serious problems that need to be rectified.

In each case, the problem has been caused by communications from the field device to the SCADA system. Note that field devices which communicate back to a SCADA system are known as **RTUs** (remote telemetry unit). Sometimes these RTUs can simply be sending I/O; in other instances the RTU might combine the communications device with some functionality, such as a dedicated pump controller.

Example 1 – Problems of Modbus for remote communications

Example 1 is a typical example of the problems that are encountered when the protocol from the field is Modbus. In a typical Modbus setup, the master telemetry unit (**MTU**) which is connected to the SCADA polls each device in the field in turn. At the time a remote device is scanned it simply provides the **current** value of each register scanned.

So example 1 shows a device that has been scanned every 5 minutes. Improvements can be made by increasing the scan speed – putting in faster radios or running more radio channels. But this doesn’t really address the main issue – collecting data that has changed.

The user might guess how often field devices or certain registers need to be scanned, but then again, that presupposes that they fully understand the dynamics of their system in every scenario. The SCADA system might **prevent** them from learning what is really happening, rather than **educating** them about the system.

Compare the real data with the sampled data once again:

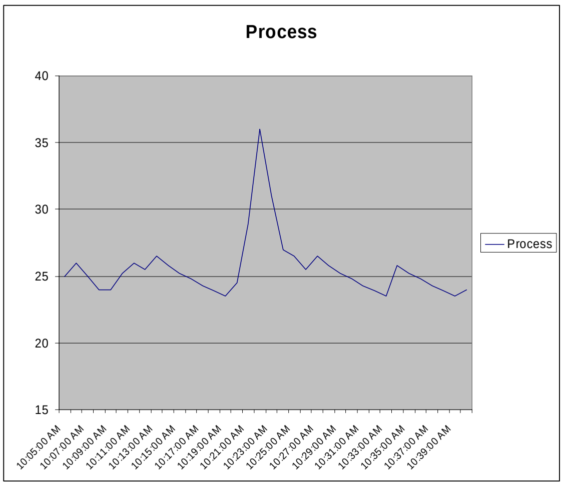


Figure 4 – Real data

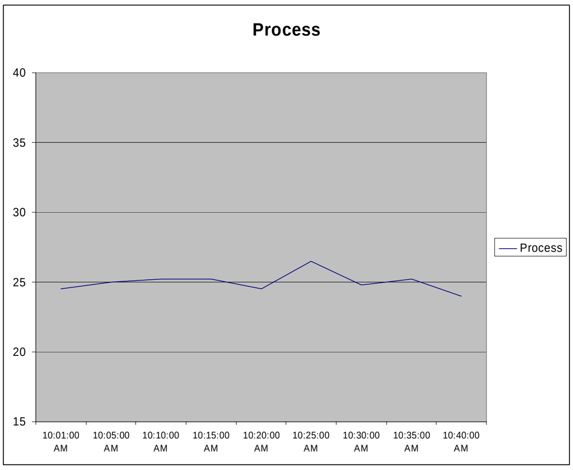


Figure 5 – Trend when Modbus protocol used

Example 2 – Sending data on change

So a protocol which simply polls periodically to capture the current snapshot is not a good protocol. How about a protocol which sends data on change?

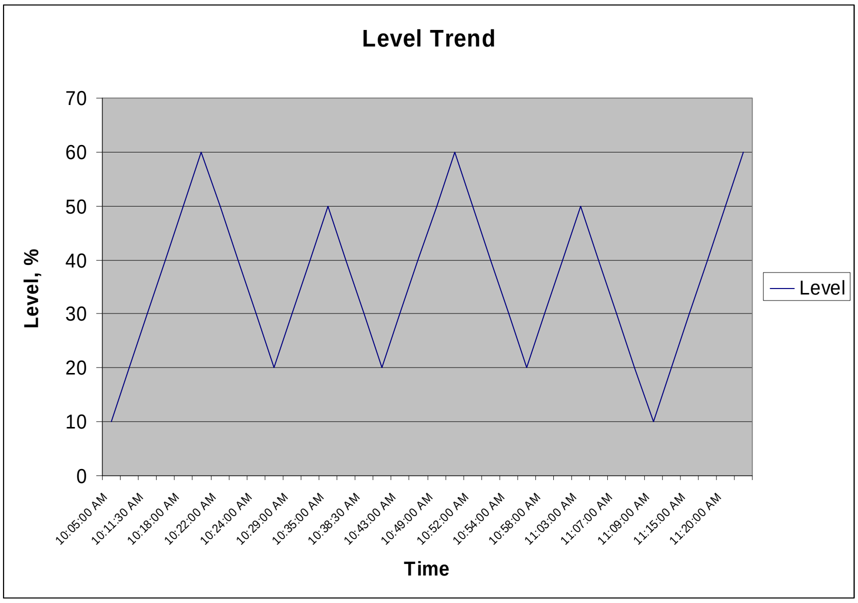


Figure 6 – Example 2

Example 2 shows a problem that still occurred with a protocol of this nature. The problem here is that the highest and lowest level are only present for maybe 10 seconds. The particular protocol (a manufacturer's protocol) initiates a data transmission when the level changes. However, if the radio network is temporarily congested – and in a point to multi-point radio network this may be true a lot of the time – then the device waits until a "clear to send" is indicated by the radio. If it takes more than 10 seconds before the radio channel becomes clear for that device, then it will send the **current** value. So by the time the channel is clear, the value has changed again and only the new value is sent.

Example 3 – Comms failing

The other problem that can occur with protocols that send data on change is what happens when the radio – or communications device – fails? Example 3 occurred because some occasional radio problems were being encountered. But suppose a radio fails, and it takes 8 hours before it is replaced. What happens when the field device next communicates with the SCADA? It only sends the current snapshot, so all the events and changes that happened for the last 8 hours are lost.

Summary of the examples

The examples above should indicate why the protocol between the field and the SCADA should be carefully chosen. If you don't want to lose valuable data, or have to carry out extensive detective work to assess whether trends and reports are demonstrating system problems, station problems – or communication problems, then you need a protocol which reliably reports all changes from the field.

There are some manufacturer's protocols around which carry out these requirements, but these should be avoided otherwise you might find you are locked into one supplier for the next 5 or 10 years.

Why DNP3

DNP3 is an open protocol which is supported by a user's group consisting of end-users and suppliers. The protocol was developed to solve the problems that resulted from the use of various protocols like Modbus.

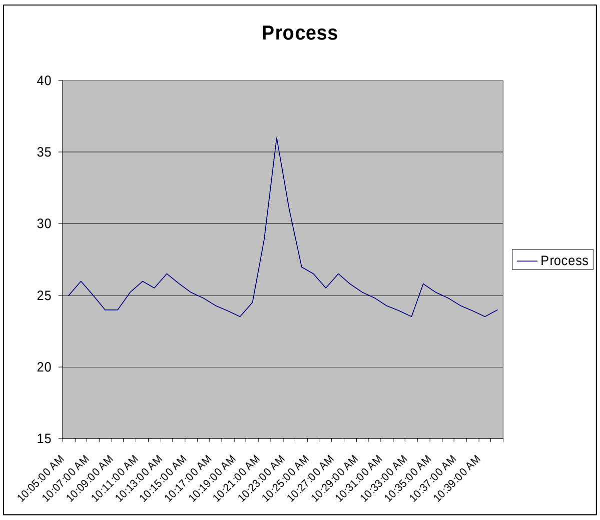


Figure 7 – Real data

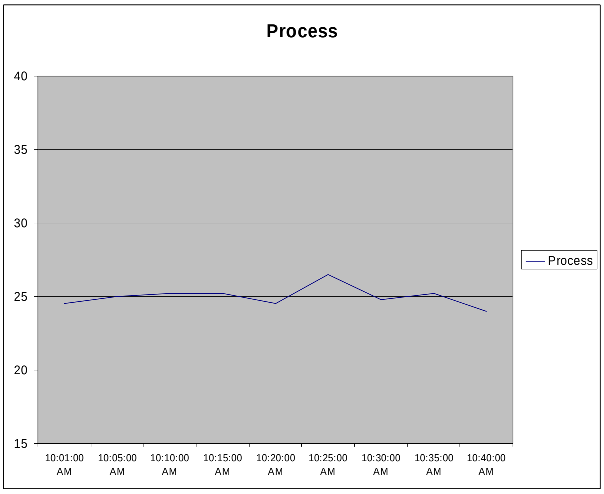


Figure 8 – Trend when Modbus protocol used

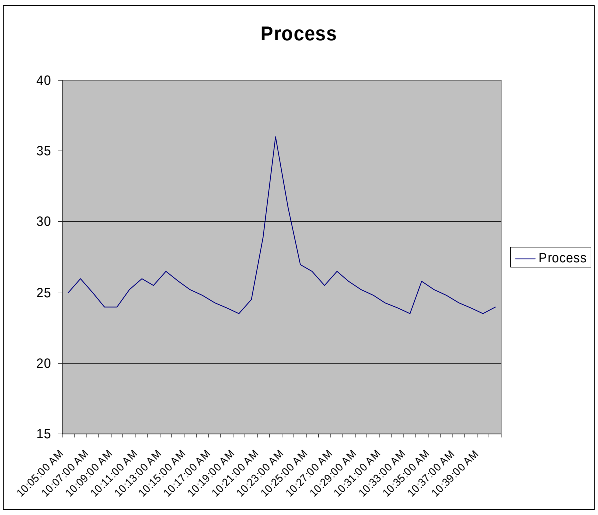


Figure 9 – Trend when DNP3 protocol used

How does DNP3 work?

In brief, DNP3 stores all changes with the date/time stamp of when they occurred. When the RTU communicates with the SCADA (and this can be initiated in a variety of ways) it sends all data which has changed. So even if a radio has been out for 8 hours, it doesn't matter. When communications are restored, all of the data, along with each original date/time stamp will be sent.

This also gives the advantage of better use of bandwidth. Instead of capturing a snapshot of every register in every device as often as possible, you capture all changes.

So suppose you have a device with 50 parameters (perhaps some I/O and some derived parameters) and over a 5 minute period 49 haven't changed while 1 parameter has changed 4 times. With Modbus you will read 50 values, 49 unchanged, and one which has changed – but you will have missed the 3 other changes. When you use DNP3, you will get just the 4 changes from the one I/O point along with the date and time that each change occurred.

The other major advantage of DNP3 is that you can assign different events to different **classes** and treat each class differently. For example, you might assign a high or low level alarm to class 1, and all other events to class 2. Class 1 will be setup to notify the SCADA on change. Class 2 will be periodically polled by the SCADA for all changes.

This gives the best of both worlds – immediate notification of serious problems, without clogging up the bandwidth with every device trying to communicate every change when it takes place.

One last point to make about DNP3 is that it can be set up so that acknowledgements are required from SCADA. This ensures data integrity.

This works as follows: the RTU sends the most recent changes to SCADA; it waits for an acknowledgement from SCADA that this data was received ok, otherwise it resends.

Security

Many manufacturers who have their own protocols claim that they offer better security than an open protocol such as DNP3.

In fact, this claim is usually flawed, as it is easy to reverse engineer most protocols unless they specifically encrypt each message to a reasonable encryption standard.

In any case, the DNP3 user group has recently released its security specification. Organizations such as MultiTrode will be implementing DNP3 security very shortly.

Storing the data in a SCADA system

It is possible to have an effective communication system based on DNP3 so that all data is reliably transmitted to the MTU, and still not make use of it.

Suppose that the trending system or historian simply samples data on a periodic basis – e.g. a user-defined period such as every 5 minutes. Many SCADA systems operate this way. The problems that result are very similar to the ones outlined in the first section.

It is important to make sure that data capture at the SCADA is based on **change** and uses the **original date/time** stamp from the field.

The MultiTrode solution

MultiTrode is a leading supplier of control and monitoring equipment to the water and wastewater industry.

The MultiSmart pump station manager has a DNP3 RTU. (This is a fully compliant DNP3 level 2 RTU). The product is an out of the box pump station controller with an intuitive user interface for operator control and engineering changes. MultiSmart can integrate into any SCADA system, and it also has Modbus to integrate into any existing Modbus-based SCADA.

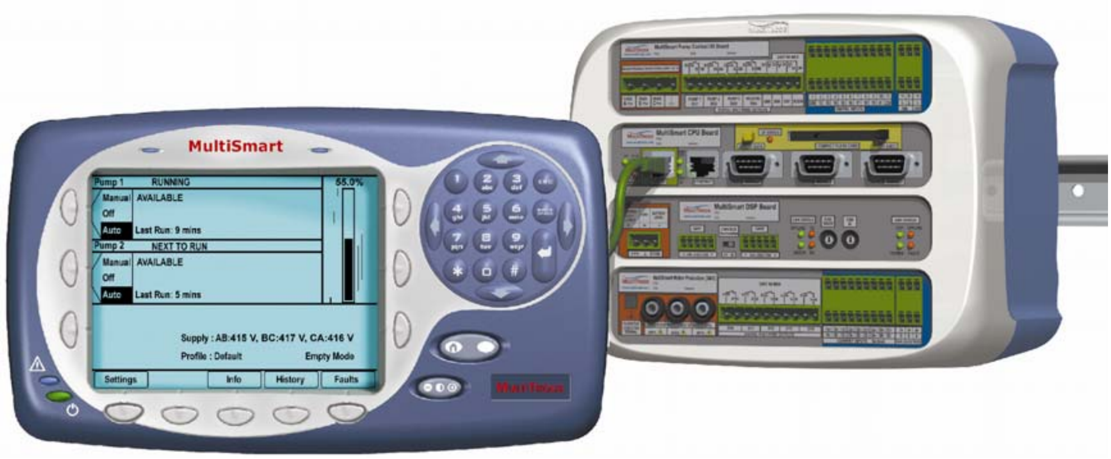


Figure 10 – MultiSmart Face Plate and I/O unit