# Building Your Own “HMI SCADA”™ Process Control System

*John P. Bougneit & Craig A. Henry*

The focus of this paper is on the “HMI SCADA”™ Process Control System, which eliminates the monitoring of data by hand, through using standard software and hardware. The system allows one to incrementally implement it as needs and resources allow. This talk will describe the system and its implementation in detail.

**For more information contact:**
Craig A. Henry
J P Technology, Inc.
2286 Church St. Unit A
East Troy , WI 53120-0863
Phone: 262-642-7671
FAX: 262-642-7681
craigh@jptechinc.com

# Building Your Own "HMI SCADA" System on a Budget

## OVERVIEW

HMI has evolved from the acronym MMI that means *man machine interface*. This term has been around for decades and has recently changed to *human machine interface* for political correctness. Both MMI and HMI mean the same thing. They refer to the connection between a human and a machine.

HMI has three main interfaces with a human:

- Visual Cues – Usually provided by a monitor or light system.
- Audible Cues – Usually alarm buzzers or pagers.
- Control System – Usually buttons, keyboard, mouse, or pointing device.

SCADA is an acronym for ***s***_upervisory_ ***c***_ontrol_ ***a***_nd_ ***d***_ata_ ***a***_cquisition_. It is a computer system for gathering and analyzing real time data. SCADA systems are used to monitor and control a plant or equipment.

To implement an effective SCADA system there must be a console control computer called an RTU. RTU is an acronym for *Remote Telemetry Unit*. This is the master computer that links the human to the machine. In most cases it does double duty by running all data logging and supervisory software.

Most SCADA systems implement a distributed database. The database is made up of points. Each point is a single piece of information like a temperature reading or a motor's start/stop condition. These points are considered *hard* data points because they actually exist on some portion of the machine. *Soft* data points are used for program control and only exist in the SCADA system. Most SCADA software has soft points for simple things such as updating counters or results of math operations.

Hardware implementation of an HMI SCADA system can look very different from one system to another. Most system implement several meters or sensors wired to a central network. This network is often a serial data line wired to the control center. Some very complex SCADA networks implement wireless data paths as well as Ethernet based data paths.

## DATA POINTS

It is possible to build an HMI SCADA system with no programming and little cost. It can be difficult to define the exact use of a data point in your SCADA system. Here are some general guidelines that can help.

- What needs to be monitored? DO NOT MONITOR EVERYTHING. You can't afford to monitor everything.
- What needs to be controlled? NOT EVERYTHING NEEDS TO BE CONTROLLED.
- What needs to be documented? If you document everything chances are your data collection will be so large it will never be used or looked at.

So monitor data points that are:

- critical to correct operation of the machine.
- critical to safe operations.
- critical to troubleshooting problems.

Control data points that:

- regularly need adjustment.
- require some math or function provided by the computer.
- need human intervention for machine operation.

Document data points that are:

- critical to machine operation and meaningful for diagnostics
- important for process documentation.

Now lets build an HMI SCADA system from the ground up. In this example we will use off the shelf hardware and software that works without programming. It is important to define your goals early in this process. Do not limit yourself with too many rules.

Our rules are:

- The system must be low cost to start.
- The system must have no programming.
- It must expand to my needs and grow with my production.
- It must serve as a supervisor to my process.
- It must be easy to understand.

In the future I want to:

- Expand the system to several machines or plants.
- Add pager or phone alarms to alerts the correct people or departments.
- Add report generation.  Some daily reports and some maintenance reports.
- BE ABLE TO TROUBLESHOOT MY PROCESS QUICKLY

These rules make sense but are very hard to adhere to.  It is not easy to keep costs low and flexibility high.  The good news is that there are a number of SCADA software packages on the market that do just that.  A simple search on the Internet will show that many companies offer some sort of software for SCADA or interface to a SCADA system.

**EXAMPLE OF A SYSTEM WITH COSTS**

Start with the data points.  Our example is a three point system.  We want to monitor a PH reading and a temperature.  We want to control a pump to add a reagent when the PH is out of range.  This example is very small compared to what can be done.  There is virtually no limit to the power of a SCADA system when properly implemented.

So we start with a PH transmitter and a temperature transmitter.  We chose a transmitter to show the power of a SCADA system.  We could easily use a PH meter with automatic temperature compensation (as you probably should) but that does not show the power of a SCADA system.  So that leaves the pump control.  We chose a network ready relay board for pump control.  We need a relay to turn our pump on and off.

The SCADA system must be able to handle standard I/O.  So what is standard I/O?  For our example we will stick to very common interface options:

- 4-20mA output from the PH transmitter.
- 4-20mA output from the temperature transmitter.
- Digital I/O for the relay.

Some other standards such as 0-5VDC and 0-10VDC are common.  In fact there are so many standard I/O options that it becomes important for you to choose an interfacing approach and stick to it.

We chose 4-20mA output for several reasons.  First we can take the low voltage reading from the PH probe and amplify it.  We can isolate it and transmit it for long distances using one interface board and low cost cable.  This board is the calibration point if we do not want the SCADA system to handle calibration.  One PH probe with cable and isolation board helps keep the system costs low. The temperature probe is a J-Type thermocouple to a similar 4-20mA isolated transmitter board.

Now we have the 4-20mA signals. So what do we do with them? We look for a device that can take 4-20mA signals and convert them to a digital value that our SCADA system can use. And it would be great if such device could also give us a relay for our pump control.

CB I/O Board

We found just the product we are looking for. We chose a three board system from Measurement Computing. The first board is #CB-7017. It is an eight channel 4-20mA input board. The second is #CB-7060. It is a four channel relay output module. The third board is a #CB-7520. It is a communication board that allows us to chain our I/O boards together and connect them to a PC. It is isolated for additional safety. We can add a bunch more boards as we expand our control system. The installation requirements for this network requires a 24VDC power supply to power the sensor modules.

#CB-7017 8ch 4-20mA input board $249.00
#CB-7060 4ch relay output board $139.00
#CB-7520 network interface board $99.00
#PS-24050D 2A 24VDC power supply from Automation Direct $76.00

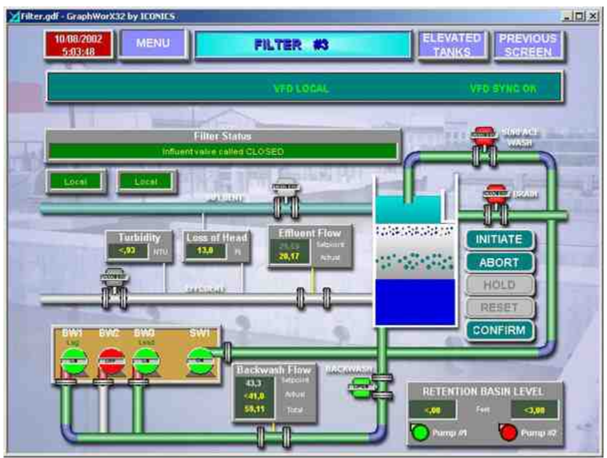
Total cost so far: $563.00

Now we have a network ready to talk to a PC. How do we bridge this network to our SCADA package? The answer is an OPC driver. An OPC driver is a program that talks to your device or network of devices and connects them to your SCADA software using a standard interface. In other words, it translates the device language to the standard OPC language. This helps to keep system costs low. Software and hardware manufacturers don't have to invest large amounts of time and money into custom drivers for every device or software package they make. Instead they choose the OPC standard and write their interface code once. This cuts the cost of networking devices by a huge amount. It makes this project easy and cost effective.

The OPC driver we chose is #NAPOPC DA SERVER from ICPDAS Company.  It is packed with setup and troubleshooting features.  But what we liked the most is that it is free.

Now we get to the SCADA software.  There are a lot of software packages available.  We chose one that is widely used, has a lot of expansion, costs incrementally as we expand, and requires no programming.  We chose Genesis32 by Iconics.  Their software gave us control, graphing, alarms, and trending.  We can expand to Internet and paging as required.  They charged us based on the number of points we need.  The best part is the software is fully functional during development for free.  It does limit us to two hours at a time.  If it times out we just stop and restart the program and work for another two hours.




Genesis32 SCADA Software Example Screen

Cost of the standard SCADA package: $595.00  (add $900 for alarms and data logging)

That takes our total cost for implementing a SCADA system to $1158.00.  We have six unused 4-20mA channels and three unused relays.  The software has 72 points available for expansion before we have to buy a software upgrade.  This truly is a SCADA system that will grow with you.  This puts the cost per data point at about $100.00 if we use all points.

The additional hardware cost for the PH Probe, cables, amplifiers, temperature probe, pump, twisted pair wire, mounting box for boards, and shipping was about $1450.00.  We separated this cost as it will vary based on your requirements.  Most of the cost is in the PH probe and isolation boards.  If isolation is not necessary then the cost drops to $950.00.

So the total hardware and software cost was about $2600.00 for the base system for three data points.  This includes the basic components to monitor and control several other data points which can be added very reasonably before additional system components are necessary. And you have done this all yourself. No programmers to pay. Simple installation. Lots of room to expand.

Just a note: We downloaded all the software for free and had the system running before any product was ever ordered!

**SUMMARY**

- You can use state-of-the-art hardware and software to monitor and control your process
- You can start with a basic system and expand it as needs require
- No expensive consultants
- Your system is compatible with other controls
- You can have both Ethernet and Internet communication.