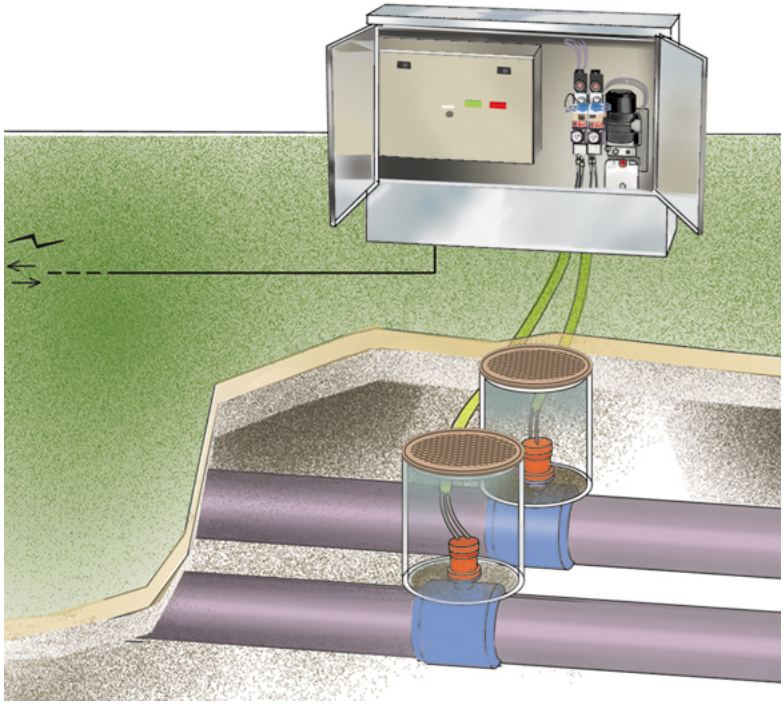
CHALMERS



Wireless remote control of a PLC system

Master's thesis in Systems, Control and Mechatronics

Martin Hjalmarsson
Stefan Johansson

Department of Signals and Systems Division of Automatic Control,
Automation and Mechatronics
Chalmers University of Technology
Göteborg, Sweden 2011
Master's Thesis EX017/2011

Master's thesis in Systems, Control and Mechatronics

Wireless remote control of a PLC system

Martin Hjalmarsson

Stefan Johansson

Supervisor:
Fredrik Görfelt
M.Sc. Automation and Mechatronics Engineering
Industriarmatur-ARI AB

Examiner:
Martin Fabian
Associate Professor
Department of Signals and Systems
Chalmers University of Technology

March 5, 2011

Wireless remote control of a PLC system

Martin Hjalmarsson
Stefan Johansson

March 5, 2011

Commissioned by: Industriarmatur-ARI AB
Industriarmatur-ARI AB
Kämpegatan 16
SE - 411 04 Göteborg
Tel: +46 (0)31-80 95 50

Department of Signals and Systems
Chalmers University of Technology
SE - 412 96 Göteborg
Tel: +46 (0)31-772 1000

Abstract

Industriarmatur-ARI AB (IA) design, manufacture and sell products for the district energy, HVAC, water and industry markets. Focus is valves and actuators, but the product range also includes level gauges, water meters, and control cabinets etc.

This thesis aims to further develop IA Hecon, a patented system for remote control of sectioning valves for district energy, to support wireless connectivity. IA Hecon is a combination of an electronic control system, a hydraulic powerpack and hydraulic valves. In its current version, it is only possible to control the valves by the operator panel in the control cabinet or by connecting the cabinet with a cable to the control room. In many situations no cable exists between the cabinet and the control room or the cable quality is very bad. To overcome this restriction a wireless solution is wanted.

Based on the prerequisites for IA an analysis was performed to find a solution suitable for further development and future product realization in line with the company's strategy.

The result of this thesis is a recommendation of a wireless solution able to remote control IA Hecon over GPRS or 3G. The solution is a combination of the mVio M2M system from Wireless Maingate and a GPRS modem or M2M platform from iOWA AB. The result also includes a prototype of a remote control wireless solution able to control IA Hecon. We also identify and judge several alternative solutions.

Keywords: PLC, M2M, Remote control, Communication protocol, SCADA, Industrial control system (ICS), Web-based Internet SCADA, GSM, GPRS, 3G, C#

Acknowledgements

We would like to thank Industriarmatur-ARI AB and its cooperative personal for their contribution and support. A special thanks go to our supervisor at Industriarmatur-ARI AB, Fredrik Görfelt. Fredrik was always available with great support and gave us important input to keep us focused.

We would also like to thank our examiner, Martin Fabian at Chalmers University of Technology, for giving us good advice and support and leading us in the right direction when we were in doubts.

Several other people have also provided valuable aid:

Kalle Wård at Wireless Maingate for help and information about M2M systems and the mVio M2M platform.

Fredrik Östbye at iOWA AB for help, support and inspiration about M2M systems and the iOWA M2M platform.

INL System AB for support and inputs about GSM/3G modems.

Sören Almqvist and Börje Anddersson at Göteborg Energi for information about district heating systems.

Contents

1 Introduction **1**
- 1.1 Background . . . 1
- 1.2 Purpose . . . 2
- 1.3 Objective . . . 2
- 1.4 Goals . . . 3
  - 1.4.1 Goals for the thesis . . . 3
  - 1.4.2 Goals for the concept . . . 3
  - 1.4.3 Goals for the prototype . . . 3
- 1.5 Method . . . 3
- 1.6 Limitations . . . 4

2 IA Hecon **5**
- 2.1 Current IA Hecon . . . 5
- 2.2 Customers control systems . . . 6
- 2.3 Prerequisites . . . 7

3 Underlying technologies **9**
- 3.1 M2M . . . 9
  - 3.1.1 Operators . . . 10
- 3.2 TCP/IP . . . 11
- 3.3 Control systems . . . 12
  - 3.3.1 PLC/HMI . . . 12
  - 3.3.2 SCADA, direct drivers and OPC . . . 13
  - 3.3.3 Web based Internet SCADA . . . 15
  - 3.3.4 Stand-alone software . . . 16
- 3.4 PLC communication . . . 16
  - 3.4.1 PLC vendor specific languages . . . 17
  - 3.4.2 Fieldbuses . . . 17
  - 3.4.3 TCP socket connection and ISO on TCP . . . 18
- 3.5 Wireless communication . . . 19
  - 3.5.1 GSM, GPRS and EDGE . . . 20
  - 3.5.2 3G and future standards . . . 21
  - 3.5.3 Performance, coverage and reliability . . . 22
- 3.6 Issues to handle . . . 24
  - 3.6.1 GPRS communication . . . 24
  - 3.6.2 Connection monitoring . . . 26

3.6.3 Minimizing data traffic . . . 27
3.6.4 Security . . . 28

4 IA Hecon wireless **30**
4.1 Possible solutions . . . 30
4.1.1 SCADA, direct drivers and OPC . . . 31
4.1.2 Stand-alone software with TCP communication . . . 32
4.1.3 mVio Internet SCADA . . . 33
4.1.4 iOWA M2M platform + mVio Internet SCADA . . . 35
4.2 Selected concept solution . . . 35
4.3 Selected prototype solution . . . 36

5 Prototype construction **38**
5.1 The old prototype . . . 38
5.2 Hardware . . . 38
5.2.1 PLC . . . 39
5.2.2 HMI . . . 39
5.2.3 GPRS modem . . . 40
5.3 Software . . . 40
5.3.1 Communication protocol . . . 41
5.3.2 PLC and HMI software . . . 42
5.3.3 C# PC-application . . . 43

6 Result **46**
6.1 Data usage . . . 49

7 Discussion **51**

8 Conclusion **53**

Appendices

A Data protocol **i**

Chapter 1

Introduction

In the first chapter we give an introduction to this thesis. The company where the thesis is carried out and the product is introduced as a background. Thereafter the purpose and goals are stated.

1.1 Background

Since 2006 Industriarmatur-ARI AB (IA) sell IA Hecon® (Hydraulic Electronic CONtrol), an own developed patented system for control of sectioning valves for district heating and cooling distribution networks. IA Hecon is a combination of an electronic control system, a hydraulic powerpack and hydraulic valves.

The IA Hecon control cabinet is installed above ground and connected to valves below ground with hydraulic hoses, see figure 1.1. The valves are operated locally or remote over optical fiber or copper cables.

IA Hecon makes it safer, easier and more efficient to operate valves. For example, the current swedish working environment legislation states for safety reasons that two workers are required if a manual underground valve is to be operated. With IA Hecon the valve is fast and safely operated from the control cabinet or the control room. This eliminates the need to deploy personnel to installations, cut travel time and cost, increase personnel safety and improve system response time.

The need for opening, closing and monitoring valves is gradually increasing as distribution networks are growing bigger, increases in complexity and the demand for sectioning grows. This means a great opportunity for IA Hecon. However, as valves often are installed far from the control rooms signaling cables must be installed for remote control. Installing the cables can become very expensive and thus needs to be eliminated. In other words, IA Hecon needs wireless connectivity.

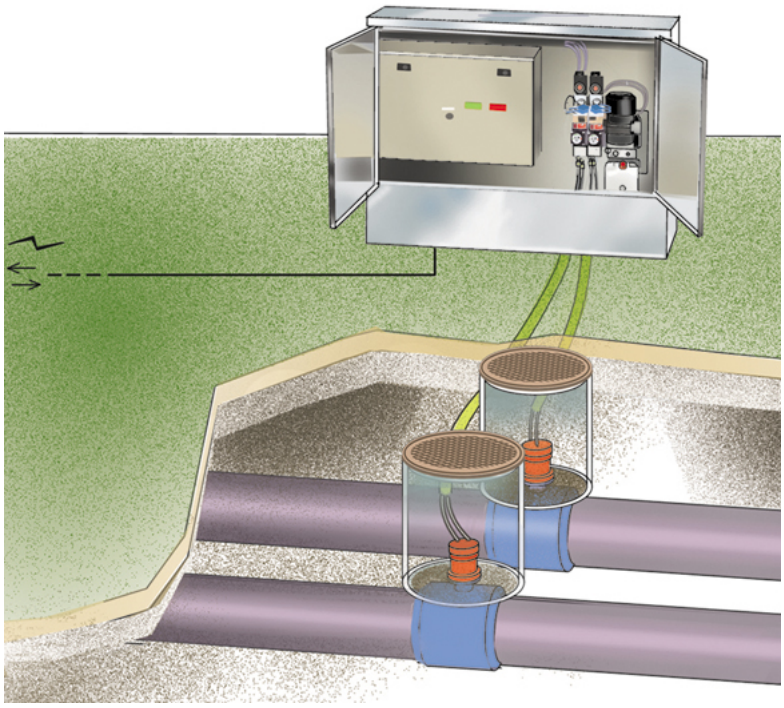


Figure 1.1: IA Hecon installation

1.2 Purpose

The purpose of this thesis aims to find and implement a general solution to allow the IA Hecon control cabinet to be wireless remote controlled over long geographical distance. This includes evaluation of different solutions and development of a prototype.

By adding wireless connectivity the need for signaling cables is eliminated. This is especially beneficial in existing distribution networks when no cables are available and the cost to install new can become very expensive.

Wireless remote control is also of great interest to Industriarmatur due to the possibility of opening for new business opportunities. One scenario could be a solution where Industriarmatur or a partner sells maintenance and support agreements to customers for maintaining the wireless remote connection.

1.3 Objective

The main objective of the thesis is to carry out an overall investigation in the subject how Industriarmatur can develop IA Hecon for wireless remote control. The investigation will be the foundation for Industriarmatur when developing a new product possible to launch to the market to withhold Industriarmaturs market-leading position in remote control of sectioning

valves.

To ensure that the stakeholder's expectations are fulfilled and all aspects are carefully considered, a number of goals have been established.

1.4 Goals

1.4.1 Goals for the thesis

- Develop a concept for the product in such way that a product realization is possible
- Develop a product prototype that fulfills the goals specified further down

1.4.2 Goals for the concept

- Wireless remote control of IA Hecon over long geographical distance ($\geq 50\ km$)
- Remote reading of status and data such as current position of the valves
- Reliable and convenient remote control
- Fulfill security standards set for remote control of district heating and cooling networks
- Alarm if communication with IA Hecon fails
- Be cost efficient to ensure a sellable product, both in start investment and running charge

1.4.3 Goals for the prototype

- Develop the current prototype of IA Hecon to enable wireless remote control
- Develop a PC-application that can control IA Hecon

1.5 Method

This thesis includes a diversity of different tasks. At first, the current IA Hecon product was studied and analyzed. A project plan was developed to establish a common idea about the project together with the stakeholders. Thereafter a market/customer study was performed to collect important information about current used systems in the industry, and to get an understanding about how possible solutions could look. After this a lot of different manufactures and companies were contacted to collect information about how different possible solutions could look like. At next stage a solution for the prototype was selected and the different parts were bought. Finally the parts were programmed and configured to work as a prototype.

1.6 Limitations

Some limitations were introduced to limit the scope of the project. The following subjects will not be investigated and further developed:

- The hydraulic system (power pack, actuators, valves etc)
- The control algorithm for opening and closing the valves

Also, the main focus has been to implement a prototype with a Siemens PLC, the main reasons for this is that the current prototype is built with a Siemens PLC and that most delivered systems is equipped with a Siemens PLC.

Chapter 2

IA Hecon

In this chapter we focus on the current IA Hecon system and look into the control cabinet to reveal the hardware. To get a clearer understanding of the complete chain, customers control systems are described and gives a brief introduction to integration issues. In the end of the chapter, we shortly analyze the prerequisites on Industriarmatur as organization in terms of developing and offering a technically advanced product.

2.1 Current IA Hecon

The current IA Hecon system exists in a couple of different variants. The main difference is the PLC fabricate and the number of valves that can be maneuvered. The reason for the difference in PLC fabricate is that many of the customers of IA wants to have the same PLC fabricate in IA Hecon as the rest of their control system. Based on the customer's preferred PLC fabricate the customer can choose from either a Siemens S7-200 or an ABB AC 800M PLC. The number of valves that can be controlled is between one and nine, but can quite easily be further extended by adding more I/O-modules to the PLC. The most normal case is two or four valves.

Figure 2.1 shows the layout of the IA Hecon cabinet.

To operate the valves, an operator panel is connected to the PLC. From the operator panel it's possible to open/close the valves and see the current position of the valves. It is also possible to make settings how the valves are controlled. For example it is possible to decide the valve speed when opening and closing the valves. This is important to avoid dangerous pressure transients. A number of alarms exist to ensure that the system is functioning correctly. The PLC monitors for example that the maximum runtime for opening/closing the valves is not exceeded, if so, it is an indicator for oil leakage or low oil level. To remote control IA Hecon it's possible to connect the PLC to a Profinet fieldbus or use multiconductor cable. This functionalty is basic and it is for the moment only possible to open/close the valves without any feedback.

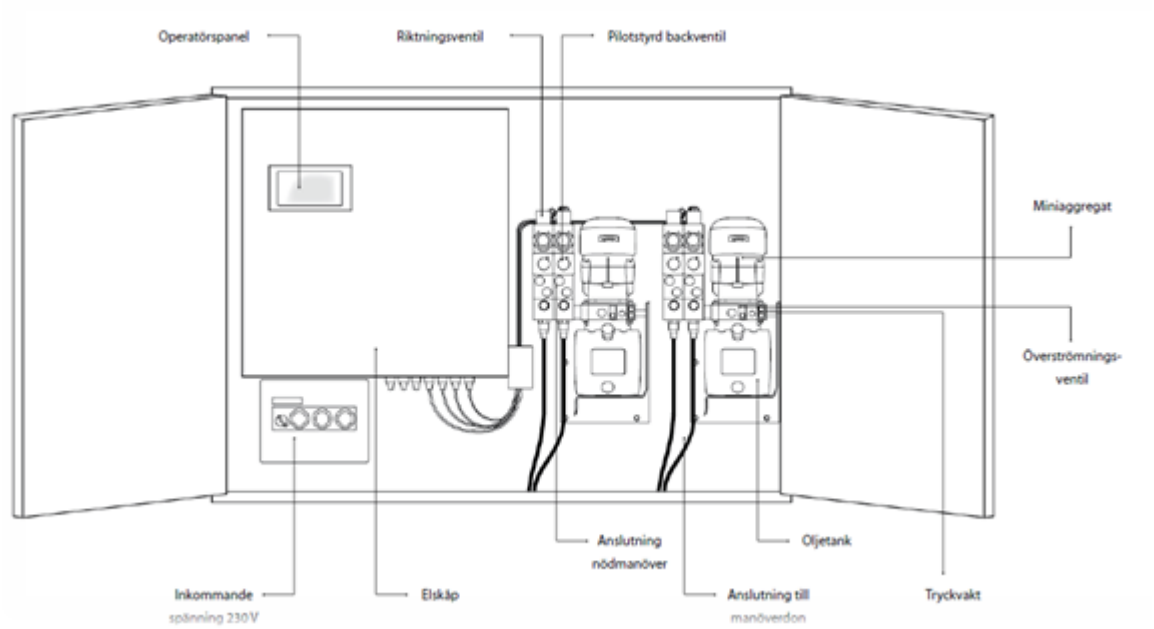


Figure 2.1: IA Hecon control cabinet

The safety demands on opening and closing sectioning valves for remote heating is very high. It is therefore crucial to always be able to maneuver the valves. To accommodate this it's possible to connect a manual pump to IA Hecon in case of breakdown of the components or power failure.

2.2 Customers control systems

Industrial control systems (ICS) is an umbrella term for control systems ranging from large supervisory control and data acquisition (SCADA) systems and distributed control systems (DCS) to smaller programmable logic controllers (PLCs). These systems are often found in the industrial and infrastructure sectors.

Industriarmaturs customers have everything from small to large systems. Common is that they all have industrial control systems for controlling their processes. Depending on the size the complexity of the industrial control system varies. For a smaller system it can be enough with a larger PLC. The operators manage the system using an operator panel communicating directly with the PLC. Large customers, for example Göteborg Energi, have several production plants spread geographically. Often the operators are placed central in one control room from which all the different plants are remotely controlled. From the control room the operators plan and optimize which plants to run with respect to factors such as weather forecasts, available energy sources, energy prices etc. This is a clear example of a large scale SCADA system.

A commonly used SCADA system is ABB Network Manager (often used in combination with

the ABB 800xA control system). ABB Network Manager is an upgrade of the former ABB S.P.I.D.E.R. and ABB Ranger systems that initially was developed for energy generation and distribution management for power companies. Today ABB Network Manager is a complete suite also able to manage heat and power plants and district heating and cooling networks. ABB SattLine is an older but still widely used system. Also the Siemens WinCC flexible SCADA system is found here and there.

Another common SCADA system is Invensys Wonderware with many installations worldwide. Wonderware is a more general SCADA system targeting no particular application compared to ABB's Network Manager that is developed for a more limited set of applications.

To be able to connect different sites and stations many actors have their own optical or electrical communication networks covering their area of interest. Often installed side by side with pipelines for water, gas and heat/cooling etc. The quality of the networks, in terms of reliability and performance, differs a lot. Newer installations often imply robust twisted-pair and/or optical networks, while older networks many times have poor signaling capabilities not applicable for use today. These old networks are one target where a wireless solution could be preferable.

To integrate IA Hecon with the customers control system the two need to communicate. If a Siemens PLC is installed in the IA Hecon control cabinet, the customers control system must support and implement communication with the Siemens PLC. If an ABB PLC is installed in the IA Hecon control cabinet, the customers control system must support and implement communication with the ABB PLC.

The mix of PLC manufacturers, communication ways (electrical, optical, profibus, profinet, modbus etc.) and SCADA systems places Industriarmatur in a quite complex position. If the customers require that IA Hecon is equipped with hardware and communication to fit their individual preferences the customers must be involved in the process making the product more of a project. For instance, the customers that today communicate with their IA Hecon's over Profibus have all been involved in the PLC software development.

Moreover, by adding wireless connectivity over GSM/GPRS one adds even more complexity to the system. Wireless communication over GSM/GPRS is unfortunately not in practice completely transparent compared to a wired IP network. This comes with a set of issues explained later that complicates the matter.

Integrating different systems is off course not a new problem and there are numerous solutions. The question is how Industriarmatur should handle this.

2.3 Prerequisites

Industriarmatur specializes in district energy, HVAC, industry valves, actuators and provides support for all aspects of valve procurement to installation projects - from component selection and application engineering to assistance during installation and start up. The technical expertise in the automation area on the contrary has in the past been very limited in the

company and there is at present no plans to offer services for integration of automation systems.

The development of IA Hecon has brought the company to do more than just sell products. To become a company that also develop their own products with a closer relationship to the customer. The technical level of IA Hecon is relative different compared to the other products offered. This put special demands on the organization.

Development of a wireless variant of IA Hecon place the product in an even more advanced segment. This put some requirements on the development of the wireless solution - the solution has to be simple enough, otherwise it would put too high demands on the organization. The company's plan is not to become a system integrator that works with custom integration with customers' systems. If possible, a solution that is easy to explain, sell and integrate with different systems is to prefer.

Due to the limited technical level at Indsutriarmatur a possible solution is to have a technical partner that develop the wireless product. One interesting concept is to develop and sell the product as a service with a monthly fee instead of as a plain product.

Chapter 3

Underlying technologies

In this chapter theory is presented for the technologies used and discussed through the thesis. The first section, M2M, covers the basics of M2M and what M2M can be used for. The second section gives a crash course in TCP/IP, the core of the Internet. In section three basics of different types of control systems is presented as a background to understand how industrial systems are controlled and monitored. The PLC communication section digs deeper in how communiction actually is implemented. In the wireless communication section, primary GSM and GPRS are described. Lastly we discuss important issues to handle.

3.1 M2M

Machine-to-machine, machine-to-man, or man-to-machine (M2M) communication refers to the set of technologies that enable data communication between machines, devices, systems and people. A concept often talked about is the "Internet of things", where "smart" devices are connected to the Internet.

The idea of M2M is itself nothing new. In the beginning M2M communication took place over wired cables, with its obvious disadvantage and constraint that a cable must be installed to the device in question. Assets such as oil and gas wells have been monitored via wired cables for long, often with very simple protocols between the devices. Today, M2M communication based on packet data is well established. The availability of GPRS, EDGE and 3G in cellular networks has broadened the use of M2M communication and the connection to the Internet has made numerous applications possible.

M2M applications range from heavy machinery worth millions, to vending machines worth tens of thousands, down to electric meters worth a couple of hundreds. All can benefit from being online. In the transportation sector M2M is used for tracking solutions, in the energy sector for smart metering. The healthcare industry use M2M to improve patient care through instant device communication and remote monitoring. Other typical M2M applications are traffic cameras, parking machines and car rental vehicles.

M2M has many benefits to businessess. Cost can be decreased by reducing downtime by performing maintenance and repairs on products in the field. Customer service can be improved by constantly monitoring products, making it easier to spot problems before they become severe.

M2M has a huge growth potential, and is expected to grow very rapidly over the next years. By the year 2020, Ericsson predicts there will be 50 billion wireless devices connected over broadband connections. Today approximately 70 million devices are connected (Ericsson, Asia-Pacific key to 50 billion connected devices, 2010).

As comparison, there are approximately 7 billion people in the world. Of these Ericsson estimates almost 5 billion have a mobile cellular subscription. See figure 3.1.

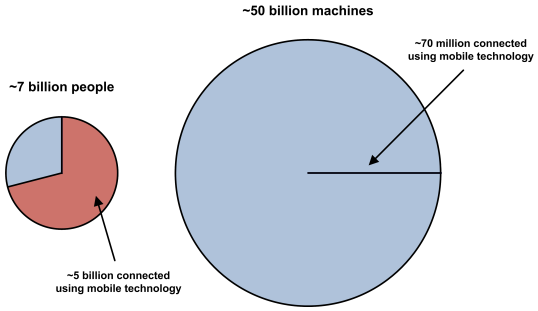


Figure 3.1: M2M growth potential

3.1.1 Operators

In Sweden, the largest operators offering M2M services are Wireless Maingate, Telia, Telenor and Multicom Security.

The cost for a M2M subscription resembles the cost for a standard mobile subscription. A monthly fee for the subscription plus a monthly fee for an optional VPN tunnel is very common. Both per MB and flat-rate subscriptions are possible. Besides this, many operators offers different service of level agreements.

A common difference between a standard mobile subscription and a M2M subscription is that a VPN solution often is used to provide a secure two-way data flow. VPN is described further in chapter 3.6.4.

Another differences is better administration possibilities. For example, a Wireless Maingate M2M customer have access to an administration tool to gain full control over subscriptions, services, orders and supplies of SIM cards. Invoicing is also adapted to the customer's needs.

A M2M SIM card is often designed to meet needs of robustness and longevity for use in harsh environments. It can also be surface-mounted, specially designed to be mounted by a robot on a curcuit board.

3.2 TCP/IP

The TCP/IP protocol suite is the set of communication protocols used for the Internet and other similar networks. TCP/IP consists of dozens of different protocols of which the "core" protocols are the transmission control protocol (TCP) and the Internet protocol (IP) (Wikipedia, Internet Protocol Suite, 2010).

The protocol suite is constructed as a set of layers where each layer solves a set of problems involving the transmission of data, and provides a well-defined service to the upper layer protocols based on using services from lower layers, see figure 3.2.

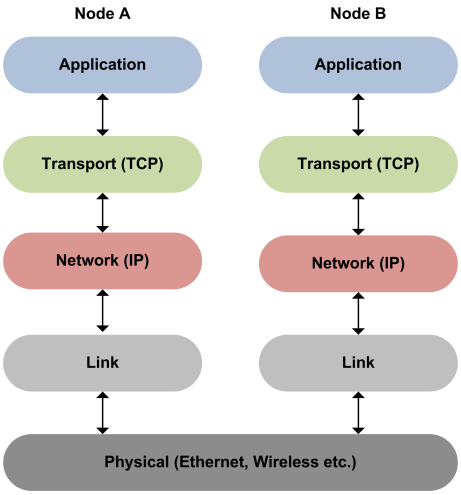


Figure 3.2: TCP/IP

The Internet protocol (IP) is the network layer protocol and provides addressing and datagram routing allowing packets to be sent over rather complicated topology of interconnected networks. The transmission control protocol (TCP) is the primary transport layer protocol, responsible for connection establishment and management and ensures that data arrives without errors and in the correct sequence.

The IP protocol transmit data without consideration for the sequencing of the packets or reliability of the connection in terms of data errors and lost packets. TCP divides the data into packets that the IP protocol in the network layer can transmit. TCP is also responsible for error checking and ensuring that the packets are not lost or received out of sequence. If necessary the TCP will request retransmission of any lost packets and will place out of sequence packets into the correct order before passing them up to the application (Blank, Andrew G., 2002).

The IP address (location) and port number (application) of a packet is combined into a functional address called a socket. This socket must exist at both nodes for communication to occur. At the server side, a socket is established by an application binding itself to a port number.

Upper layers are logically closer to the user and deal with more abstract data, relying on lower

layer protocols to translate data into forms that can eventually be physically transmitted.

By using TCP/IP as a foundation for communication a lot of the concerns associated with communication can be neglected in the development of a user application. The well-defined services in the lower layers makes the development of the communication in the application layer rather simple.

A great benefit with TCP/IP is that it's a widely published and open standard, well known and used in many different computer systems, hardware and network configurations.

In the transport layer another common protocol is UDP (User Datagram Protocol). UDP compared to TCP uses a simple transmission model without implicit hand-shaking dialogues for guaranteeing reliability, ordering, or data integrity. Thus, UDP provides an unreliable service and datagrams may arrive out of sequence, appear as duplicates or go missing. UDP assumes that error checking and correction is either not necessary or performed in the application. This avoids the overhead associated with such processing (Wikipedia, User Datagram Protocol, 2010).

3.3 Control systems

There are many different types of industrial control systems. A very simple control system can be a single PLC. A more advanced control system can consist of several PLCs connected to a master PLC or SCADA system.

Instead of a PLC, there are dozens of manufacturers that build their own hardware that aren't PLCs but perform the same task as a PLC. Therefore, in some sense, when describing a PLC, this could target other hardware aswell.

3.3.1 PLC/HMI

For smaller factories and processes, it is possible to use a single master PLC as a control system. The PLC is often a bigger and more powerful PLC, in charge of the whole system. The PLC executes instructions by the help of sensors and actuators spread in the plant connected to the PLC using distributed I/O communicating over an electrical or optical network . The PLC can also be connected to smaller PLCs that are in charge of a specific and smaller task in the system. The communication between the PLCs is usually implemented over a fieldbus communication protocol.

Operators manage the plant using a HMI (human machine interface) which can be an operator panel or a simple control software communicating directly with the master PLC. One scenario could be that the operator enters a new set point for a process using the HMI connected to the master PLC. The master PLC then distributes this new set point to the PLC in charge of the specific process over a fieldbus communication protocol. Finally, the PLC changes its output to steer the process to the set point.

In recent years many new PLCs comes with a built-in web server, making it possible to control the PLC from a web browser on a LAN or over the Internet. Using the web browser the operator can retrieve data, alter the PLC configuration and receive statistics in a very simple manner. This has made it easier to communicate and get a clear overview of the PLC system.

Also, the HMI manufacturers in the industry have developed and adapted to the new IP technologies. More advanced HMI's include a VNC (virtual network computing) server, which is a graphical desktop sharing system for remote control of another computer (or in this case, an HMI). The VNC server transmits the keyboard and mouse events from one computer to another and relays the graphical screen updates over a network. By this, an operator can connect to the HMI over the network, and use the mouse and keyboard to operate the HMI and thus the control system remotely.

One possibility for remote control of IA Hecon could be to have a PLC with built-in web-server in the control cabinet and let the operators control IA Hecon using a web browser. But due to the somehow complicated way to get an overview of a larger system (the operator need to log on to each PLC to see its current status), this solutions was not further investigated.

3.3.2 SCADA, direct drivers and OPC

For more complex and advanced systems, a SCADA system is often a good candidate. SCADA is the acronym for supervisory control and data acquisition. SCADA systems are used to monitor and control processes and are often found in auto manufacturing, electrical transmission, water distribution systems, chemical plants and other industries.

What SCADA systems include varies. Some say it's only the supervisory software that is used to monitor and control. Others say it includes the whole set of equipment, that is, sensors, actuators, switches, computers, servers, databases, etc., necessary for the SCADA to do its job.

A typical definition of a SCADA system is:

"SCADA is the technology that enables a user to collect data from one or more distant facilities and/or send limited control instructions to those facilities." (Stuart A. Boyer., 2004).

In this thesis we use the definition that a SCADA system is the supervisory software.

A SCADA system has two basic functions. The first is to display information about the current operating conditions of a piece of a plant in an informative and graphical interface. The second is to allow supervisory control of the plant by company personnel.

An industrial control system (ICS) typically incorporates equipment that is deployed in widely dispersed locations ranging from a few meters to thousands of kilometers. Figure 3.3 shows an example of an ICS. The three processes 1, 2 and 3 are controlled by the PLCs installed close to the processes and the processes are run either be an operator using the corresponding HMI or by instructions from the SCADA servers. This means that the PLCs run the inner control

loops while the SCADA servers run the outer control loops. A big plant could for example have thousands of PLCs running PID-regulators spread over the plant which all are running by themselves but listens after set-points from the SCADA servers. At the same level where the SCADA servers reside other servers used for example process optimization can operate.

For planning and other tasks an operations network is available to give access to process data but not to the actual process control. This is possible by firewalls and user authentication system. The network is further split to a corporate network to allow even more limited data.

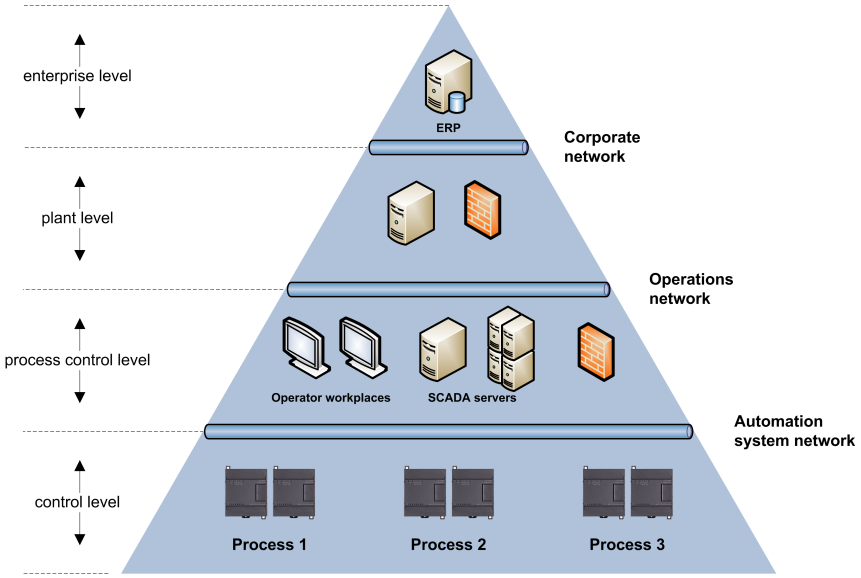


Figure 3.3: Industrial control system architecture

To communicate between a PLC and a SCADA system the two need to communicate over a common interface, and talk the same language - the communication protocol or the driver. Common communication interfaces for PLCs are for example RS-232, RS-485 and Ethernet.

Commercial SCADA systems are generally not designed for a particular brand or model of a PLC. The software is generic and covers all types of controllers. To make this generic software work with various manufacturers' equipment, a software driver is written which uses a controllers unique Applications Programming Interface (API) function calls to provide communications. The required drivers are installed depending on the hardware being used.

In the early years of SCADA systems this was often a big issue. A SCADA software manufacturer needed to implement each PLC manufacturer's drivers. Today SCADA manufacturers still implement drivers targeting specific PLCs, but also the OPC standard is available. OPC is an industrial standard developed to solve integration problems by providing a common interface for communication between different products from different vendors. By taking advantage of OPC technology the only component the SCADA manufacturer needs to implement is an OPC client that can talk OPC with an OPC server. The OPC server in turn communicates with the PLC, see figure 3.4. The PLC manufacturers develop OPC servers that fit their products. There are also OPC servers developed by third party.

Instead of using drivers or OPC another solution is to develop a custom data protocol that

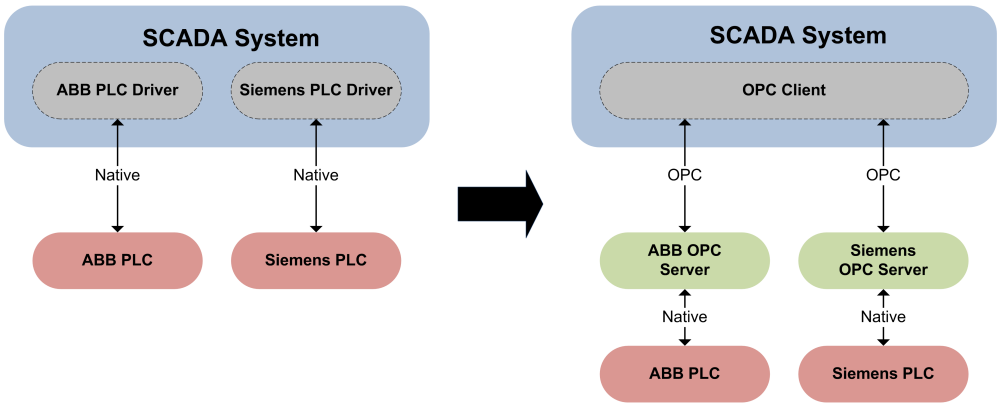


Figure 3.4: Direct drivers vs. OPC

is transmitted over for example RS-232 or Ethernet (in a TCP/UDP packet). This of course requires that the SCADA system and PLC is equipped with hardware for this. A disadvantage compared to drivers or OPC is that logic must be implemented in the PLC software for parsing the protocol. It's not possible to modify the PLCs input/outputs or memory directly.

The first generations of SCADA system were independent system with no connectivity to other systems. The next generation was distributed networks connected using proprietary network protocols. Each station was responsible for a particular task thus making the size and complexity of each station less than the one used in first generation. Since the protocols were proprietary, very few people beyond the developers and hackers knew enough to determine how secure a SCADA installation was. Since both parties had vested interests in keeping security issues quiet, the security of a SCADA installation was often badly overestimated, if it was considered at all.

The current and third generation uses open system architecture and utilize open standards and protocols. The usage of standard protocols and security techniques means that standard security improvements are direct applicable to the SCADA systems. On the other hand, the systems are potentially vulnerable to remote cyber-attacks due to the use of standard protocols. The development of Internet and TCP/IP protocol has made it possible to transform the SCADA system to use regular LANs and WANs for communication between the master station and communication equipment (Wikipedia, SCADA, 2010).

3.3.3 Web based Internet SCADA

The development of SCADA systems has also made it possible to put the actual SCADA system on the Internet. Hosted SCADA services are being offered that allow users to monitor and control remote equipment by using a standard web browser logging onto a secure website.

mVio is an Internet based control and monitoring system hosted by Wireless Maingate, especially developed to communicate to machines over GSM and 3G networks. Maingate's services are operated in a telecom quality environment and the communication between Maingate and the web browser uses SSL encryption. Basically, mVio is a web based service that let the user

maneuver and monitor connected wireless devices from a web browser or existing application or infrastructure such as SCADA, logistics and business systems.

mVio's main advantage is that it's developed to deal with the issues related to communicating with wireless GSM/3G devices. By having the functionalities to handles these problems, mVio will drastically reduce the demand on the customers own control system to handle this kind of problems. If the customer has an own control system it's possible to integrate this with mVio by the help of an API. In a smaller organization, it might be sufficient to use the mVio web portal.

3.3.4 Stand-alone software

A possible control system for a specific task is to use a stand-alone software. The software can look in various ways, depending on the task it is supposed to do. The software can be running continually or only when the system needs to be controlled.

The main advantage with this kind of control system is that the software can be developed and optimized for the specific task. There are often no license fees or other software's needed (runtime) to run the software and it's run cost is therefore very low. The software can be developed in almost any kind of programming language, depending on the preferences of the developer.

On the other side, two disadvantages of a stand-alone software is that it most certainly doesn't follow any control standards and it is an additional software that needs to be handled within the organization. Also, if the software is part of a bigger control system, the operator might has to switch between the systems. It is also a new type of software that the operator needs to understand and get familiar with.

A stand-alone software has to be running in the background (for example as a windows service) to be able to collect data.

RedDetect XTool is a good example of a stand-alone control system used for systems similar to IA Hecon. XTool is used to control and collect data from devices measuring water levels, humidity, temperatures etc. in HVAC chambers. To communicate between RedDetect XTool and devices Ethernet is used. This means that the customer can use a regular LAN, Internet or connect the device to a wireless GPRS modem. XTool has a built in OPC server that makes it possible to integrate it with other control systems (Wideco Sweden, RedDetect XTool, 2010).

3.4 PLC communication

Somehow the control system needs to communicate with the PLC, as shown in figure 3.5. This can be done in a number of ways. In the industry there are many different opinions how this should be done, but there's no general solution. In an ideal world all PLCs and devices

would talk the same communication language and it wouldn't matter which unit to connect to the other. Unfortunately, this is not the case. From the early development of PLCs all manufactures developed their own communication protocol to communicate with and between their PLCs. One important reason for this was to prevent the customers from changing and mix PLC brands. Selecting the correct communication language is a very important issue. Below follows a couple of different solutions how to communicate with a PLC (Bailey, David., 2003).

Figure 3.5: Industrial control system architecture

3.4.1 PLC vendor specific languages

Each PLC manufacturer has its own communication languages that they prefer to use. This is the language that usually is used by their own software to up- and download PLC software and for direct driver and OPC communication. Siemens has a communication protocol named S7, which is a proprietary protocol that is not official. This makes it almost impossible to implement it in an own developed software. ABB have a preferred communication protocol to communicate with their PLCs, named MMS (Manufacturing Message Specification).

Vendor specific languages can't be used for communication between different PLC brands.

3.4.2 Fieldbuses

To overcome the problematic around vendor specific languages, different types of fieldbuses started to develop. The benefits to the end users are that all devices can use the same protocol and thereby the users could buy any product and plug it into the system without having interfacing problems. Unfortunately the support for different fieldbuses are limited and often it is necessary to buy an extra communication module or an extra license to the PLC to enable the use of a fieldbus.

In table 3.1, and table 3.2, some of the most common fieldbus protocols are listed. Besides these, there are many more. This cause to some extent the same problems mentioned for the PLC vendor specific languages (Park, John. Mackay, Steve. Wright, Edwin., 2003).

Fieldbuses are mainly used for communication within factories, but the emerging of the TCP/IP protocol has created new possibilities. Industrial Ethernet is the name given to the use of Ethernet in an industrial environment, for automation and process control. A number of techniques are used to adapt the Ethernet protocol for the needs of industrial processes, which must provide real time behavior. By using non-proprietary protocols, automation systems

Fieldbus	Year introduced
Profibus DP/PA	1994-95
Modbus RTU	1979
CANOpen	1995
DeviceNet	1994

Table 3.1: Fieldbuses, serial

Fieldbus	Year introduced
PROFINET IO	2003
PROFINET IRT	2003
Modbus TCP	1999
EtherCat	2005
Ethernet/IP	late 1990's

Table 3.2: Fieldbuses, Ethernet based

from different manufacturers can be interconnected throughout a process plant (Wikipedia, Industrial Ethernet, 2010).

In the left side of figure 3.6, a couple of Ethernet based fieldbus protocols are placed in the OSI-model. To the right of the separator line, the two serial fieldbus protocols Profibus DP and Modbus RTU are placed.

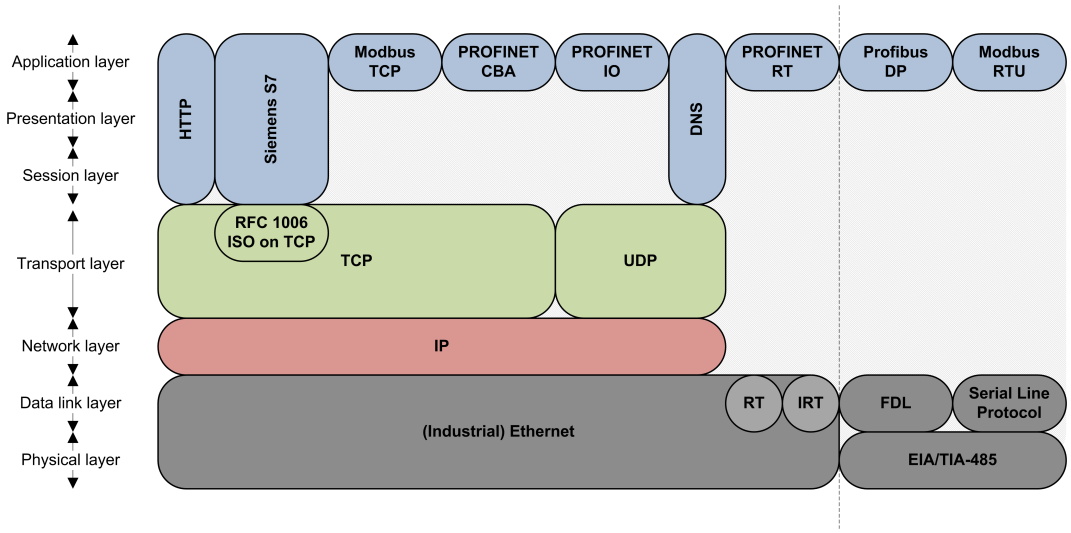


Figure 3.6: Fieldbus protocols

3.4.3 TCP socket connection and ISO on TCP

A possible solution to communicate with a PLC is to use the TCP/IP protocol. The benefits with using a protocol built on the fundamentals of the TCP/IP protocol are that a lot of platforms have support for this (SCADA systems, hardware etc.) and it's easy and free

to implement in own developed software. Siemens PLCs support communication over the protocols RFC 793 (native TCP/IP), and the add-on RFC 1006 (ISO Transport Service on top of the TCP).

The disadvantage using either of these is that it is impossible to directly access PLC tags and memory, which is possible when using a vendor specific languages or a fieldbus protocol. Instead the functionality to write/read information in the PLC has to be integrated in the PLC software by the developer. This can be quite time consuming to develop.

3.5 Wireless communication

Wireless communication is a field that has been around for over hundred years starting around 1897 with Marconi's successful demonstration of wireless telegraphy (Eberspächer, J. Vögel, H-J. Bettstetter, C. Hartmann, C., 2008).Television transmission, in its early days, was broadcast by wireless radio transmitters. Later many wireless transmitters were replaced by cable transmission. Similarly, the point-to-point microwave radio links that form the backbone of many mobile telephone networks are being replaced by optical fibers that allow higher bandwidths.

In the first example the wireless technology become outdated when a wired distribution network was installed. In the second example the wireless technology was replaced by new optical fiber technology. The opposite is happening today. Wireless cellular technology is more and more replacing wired telephone lines (particular in parts of the world where wired networks are not well developed). The interesting thing here is that there are many scenarios in which there is a choice between wireless and wired technology, and that the choice often changes when new technology becomes available.

Wireless communication is gaining more and more popularity in the industrial sector. There are several different wireless networking technologies available. Some are:

- Radio modem
- GSM/3G
- ZigBee
- Wireless HART

Radio modem is a technology suitable for creating a private radio network. Radio modem can be used for short and long range communication and often implies use of licensed frequencies in the UHF or VHF band. Licensed frequencies have the advantage that they are reserved for a certain user in a certain area thus ensuring that there is a less likelihood to have radio interference from other RF transmitters. One issue that can arise is that there are areas and buildings sensitive to RF interference where it's directly impropriate and unfeasible to setup a wireless communication link. One good example of this is Norrenergi in Södertälje who were in the process of setting up a radio modem link but was forced to abandon this because

of the nearby Karolinska Institute (KI) being sensitive to interference (Industriarmatur-ARI AB, 2010).

ZigBee is a standard developed for targeting low data rate and long battery life applications. ZigBee operates in three different unlicensed frequency bands, 2.4GHz (Global), 868 MHz (Europe) and 915 MHz (North America and Australia). Transmission range for ZigBee is between 10 to 75 meters and up to 1500 meters for ZigBee pro, although heavily dependent on the environment. To achieve long-range communication ZigBee supports mesh-networking which mean that all ZigBee nodes in the network acts as routers forwarding packets. Since ZigBee requires infrastructure for long-range communication this technology was abandoned early in the project. In Göteborg ZigBee is used for automatic meter readings (Göteborg Energi, 2010). The ZigBee network is owned by Göteborg Energi.

Wireless HART is a wireless mesh network communications protocol for process automation applications. The network uses IEEE 802.15.4 compatible radios operating in the 2.4GHz radio band. Also Wireless HART was disregarded early in the project for the same reason as ZigBee.

GSM/3G technology is today the most natural choice for a M2M solution. GSM/3G is widely adopted worldwide and therefore the infrastructure required for an M2M application already exists. This a great benefit. The range could be a problem, but in general, GSM/3G has good coverage. In chapter 3.5.3 this will be discussed further. There exist numerous applications that use GSM, for example Scania and Volvo for fleet management and iOWAs SIPP node for emptying of transformer pits. The already existing infrastructure makes it possible to rather fast make a device wireless. Obviously this made the choice of technology easy.

3.5.1 GSM, GPRS and EDGE

The GSM (Global System for Mobile Communication, originally from Groupe Spécial Mobile) family (GSM, GPRS, EDGE) has become one of the most successful innovations in history. As of June 2008, more than 2.9 billion subscribers were using GSM, corresponding to a market share of more than 81% (Eberspächer, J. Vögel, H-J. Bettstetter, C. Hartmann, C., 2008). The GSM standard is the result of the work started to develop a definition of a Europe-wide standard for digital mobile radio.

The original data transmission mechanism of GSM is circuit switched data (CSD) and requires the establishment of a dedicated point-to-point connection. When making a normal voice call the circuit switched mechanism is used. The circuit switched design charges for the time the user is connected. Therefore, the user often terminates the connection after data has been transmitted, and set up a new connection when needed. The problem with this is that it takes some time to set up a new connection. Also it does not fit a scenario when the user need to be connected for long times.

GPRS is a technology added to the GSM architecture to allow transmission of packet switched data (PSD). A packet-switched network makes it possible to transmit data as packets where each packet contains a destination address. This mean that all packets does not need to

travel the same path, and can be routed via different paths as traffic conditions change. The packets can even arrive out of order as each packet has a sequence number. By using the packet switched technology it's possible to be online virtually all the time without having to pay large amount of money for merely being online.

GPRS uses and share the capacity of GSM radio channels, and the base GSM network components. GPRS is integrated into the GSM architecture by so called GPRS Support Nodes (GSNs). A Gateway GPRS Support Node (GGSN) acts as an interface to external data packet networks, e.g. to the Internet. The Access Point Name (APN) is a logical name referring to a GGSN.

To exchange data packets with external packet data networks two things must occur. First the mobile station (MS) performs a GPRS attach procedure to enter the GPRS network. Second a PDP (Packet Data Protocol) context activation is carried out to apply for an IP address to use in the external packet data network.

When the MS apply for an IP-address the operator either hands out a dynamic or a static IP-address. Depending on the arrangement the operator may also place the MS in a private Virtual LAN (VLAN) which increases the security as the MS is not public on the Internet and only can contact other devices in the VLAN. This is also a way to spare the limited number of IP-addresses of IPv4. If placed in a VLAN this most often means that to reach the MS from the Internet a VPN tunnel must be opened. This also provides a security aspect, as the GPRS device is not accessible by so called scanners used by hackers since the IP address cannot be accessed.

For some GPRS networks the IP address cannot be accessed from outside as the providers perform the addressing to the "normal Internet" via a NAT table (Network Address Translation). If this is the case, the MS must initiate the communication.

As GPRS is billed by the amount of data, this prevents unwanted and costly data traffic.

Since GSM is standardized it is possible to roam between different networks. Roaming is defined as the ability to make and receive calls, and send and receive data when travelling outside the geographical coverage area of the home network, by means of using a visited network. See figure 3.7 where the green line shows the standard route, and the blue line shows the route when the mobile station is in the visited network. To make this possible the home operator must have a roaming agreement with the other network operators. By this a subscription with for example home network Telia can be reached in places and countries where Telia either have poor reception or has no own network.

3.5.2 3G and future standards

The main reason to use 3G instead of GSM is the higher bandwidth and lower latency. From an IP perspective there really is no difference. The radios for GPRS and 3G are different, but the IP networks look the same.

Also, while GSM is a European standard, the third generation partnership project (3GPP)

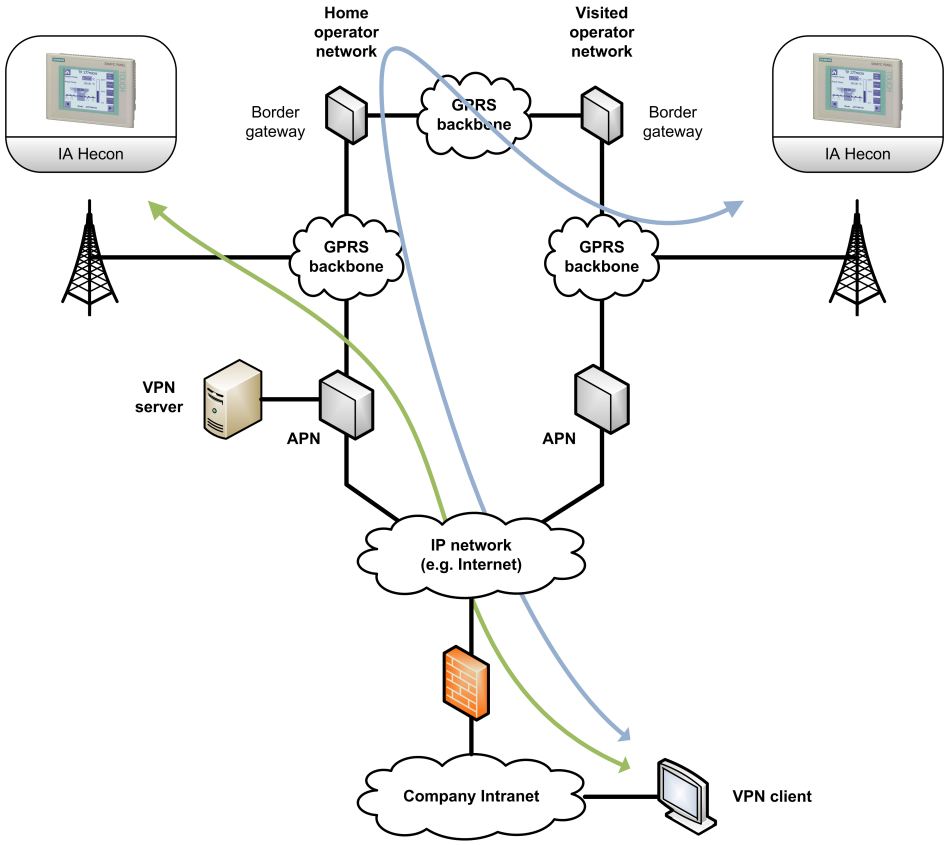


Figure 3.7: GSM and GPRS architecture

was founded in cooperation with other standardization committees worldwide. What this means is that 3G devices does not suffer as much from band compatibility issues as GSM.

3.5.3 Performance, coverage and reliability

Performance can be divided into throughput and response time. These depend on, among other things, the hardware, the channel quality (reception), number of simultaneous users and the quality of service (QoS) level.

GPRS uses timeslots for data transfer, at least 1 and at most 8. Depending on how the data is coded the data rate differs. Coding scheme 1 (CS-1) yields a data rate of only 9.05 kbps per slot but a very reliable coding. Coding scheme 4 (CS-4) yields a data rate of 21.4 kbps per slot but requires good channel conditions (good reception). The theoretical maximum throughput of GPRS is thus 171.2 kbps. In practice however, multiple users share time slots and much lower data rate will be available to the individual user. Moreover, if the reception is bad, coding scheme 4 will not be possible to use. Which coding scheme the mobile station use is automatically selected.

The multislot class of a mobile station defines the number of timeslots the mobile station may use for upload and download of data. For example, class 12 allows at most 4 upload slots

and at most 4 download slots and a maximum of total 5 active slots. If 4 slots are used for download and 1 for upload the theoretical maximum download rate will be 85.6$kbps$ and the max upload rate 21.4$kbps$. However, data throughput between $10 - 50kbps$ is more realistic (Eberspächer, J. Vögel, H-J. Bettstetter, C. Hartmann, C., 2008).

To improve the bandwidth of GPRS most operators today have implemented Enhanced Data rates for GSM Evolution (EDGE). EDGE is a technology which improves the data rate by using other transmission techniques. With EDGE data is coded using other techniques than for GPRS. The highest data rate per timeslot when EDGE is used is 59.2 kbps. Theoretical maximum for 8 timeslots is therefore 473.6 kbps. A multislot class 12 EDGE mobile station can therefore theoretically download or upload data at 236.8 kbps, almost 3 times faster than standard GPRS. In practice these are, as for standard GPRS, lower.

Response times for GPRS and EDGE can vary a lot. Experiments on the prototype described later on show response times from 300 ms up to several seconds. INSYS Microelectronics, a large manufacturer of GSM/UMTS hardware, states typical PING times for GPRS to 700 ms and EDGE to 350 ms.

Another factor that affects the performance is the quality of service (QoS) level. A mobile station with a high QoS level is more prioritized than a mobile station with a lower QoS level. If mobile stations compete for resources the one with the highest QoS will win, in practice a mobile station with lower QoS loses some fraction of the capacity, seldom all capacity. With GPRS it's possible to request a certain QoS level from the service provider, but often the service providers always supply "best effort" as QoS.

In Sweden the GSM networks cover most parts while the 3G networks mostly cover larger cities. In figure 3.8 is the Telia GSM coverage map of 1st April 2010. In figure 3.9 is the same for Turbo-3G, Turbo-3G+ and 4G for the same date. As one can see there are areas where the GSM coverage is bad, but in large GSM coverage is very good.

Figure 3.8: Coverage map GSM

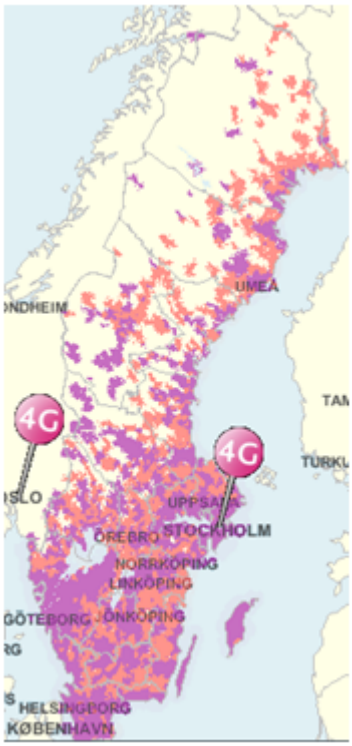


Figure 3.9: Coverage map 3G, 4G

If reception is bad one solution is to change to a better antenna and use an amplifier. Depending on the sampling rate and amount of data to transfer, one must decide whether GPRS is enough or if EDGE or even 3G is required. For this thesis GPRS was selected. The application only require a very low bandwidth and doesn't require the lowest response times.

3.6 Issues to handle

When combining the technologies discussed in this chapter to a complete solution a number of issues arise which need to be handled to get a functional and reliable system. In the following sections the most important are analyzed.

3.6.1 GPRS communication

Devices connected over GPRS aren't as "connected" as devices connected with a cable. Sometimes the connection to the unit isn't available and it is therefore impossible to communicate with the unit. The three major issues that make GPRS not as transparent to an Ethernet cable are:

1. A mobile station can become unreachable because the network closes the PDP context

and detach the mobile station from the GPRS network

2. A connection can be dropped because of timeout

3. A connection can be dropped (for a certain time) because a higher QoS level mobile station arrive to a cell

The first issue can arise when a mobile station has been idle for a period of time or if the mobile station loses reception for a prolonged time. The second issue can arise when the connection itself has been idle for some time. Both are caused by the service providers who don't want a mobile station to take up system resources unnecessarily. The third issue occur when a more prioritized mobile station arrive to a cell. Most often the mobile station with the lower QoS loses a fraction of the capacity, but it's also possible to lose the connection totally. The only way to avoid this is to have the highest QoS.

To minimize the two first issues from happening one can periodically send data between the mobile station and a server. Most mobile stations for M2M application has a built-in function (often named PING or keep alive timer) that can be set to trigger at periodic time. Also many mobile stations has a reset function that can restart the whole unit at predetermined times.

A problem with a TCP socket connection is that the TCP protocol by default is an idle protocol. When a TCP socket connection is opened, there is a handshake. Similar, when closing a TCP connection there is a handshake. However, once a TCP connection has been established, if neither side sends data, no packets are sent over the connection. This is by design and very efficient in the sense that no polling packets are sent across the network to check the connection state. This means however, that neither side will be notified that the connection has been lost.

To detect a broken TCP connection the only way is to send data. When one side tries to send data to the other side the sender will receive an acknowledgement if the connection is still active. When using UDP instead of TCP, the sender will not receive this acknowledgement. In that case other logic must be implemented to detect a broken connection.

TCP keep-alive is an optional feature of the TCP protocol with the main purpose to notify when a peer dies. If enabled, when setting up a TCP connection, a number of timers are initialized. When the keep-alive timer reaches zero, a keep-alive probe packet is sent. If a reply is received for the keep-alive probe, this confirms that the connection still is active. Besides its major purpose, TCP keep-alive can also be used to prevent inactivity from disconnecting the channel. As TCP keep-alive is an optional feature, many implementors don't implement it. If so other measures must be used to detect and prevent loss of communication.

On the mobile station side there are often two mechanisms to keep the connection alive. The first is a PING function that periodically PING a specific server. If the server does not respond the mobile station initiates a GPRS attach and PDP context activation to reestablish the connection. The second is a reset timer that periodically restarts the mobile station and thereafter initiates a GPRS attach and PDP context activation.

If the server tries to connect to the mobile station but the connection is broken there are three alternatives:

1. Communicate with the mobile station by SMS
2. Force a GPRS attach and PDP context activation by sending a SMS and thereafter try establishing a connection
3. Wait for the PING or periodic reset timer of the mobile station to trigger and thereafter try establishing a connection

The reason to use SMS is that SMS in some senses is more reliable than GPRS. A number of reasons are:

1. The SMS service operates independent of the GPRS service, which means that the GPRS could be down while SMS is still functional
2. If the mobile station is unreachable, a SMS is buffered in the network for a validity period until the receiving mobile station is reachable and then delivered
3. SMS works with weaker signal levels

Because of the issues explained above, a wireless communication solution must include functionality to monitor the connection state, and also if the underlying function of the system is critical, provide SMS functionality as a backup to the GPRS service.

This is a somewhat new situation for most of the common control systems. To have a operative system with high demands on monitoring and availability, the control system need to have the necessary functionality to compensate for this.

If each customer of Industriarmatur should implement these functions in their own system, it could lead to a rather complicated and expensive investment, especially to implement SMS as a backup data carrier. Due to the fact that it exists numerous of different control system, it put high demands on Industriarmatur and requires good technical skills to give good advice to customers. Another way would be if the product itself or the associated control system already had these functions.

3.6.2 Connection monitoring

The potential IA Hecon customers requires that the connection to IA Hecon is continually monitored. Normally, the valves are only operated a few times every year, but it is very cruical to the customers to know if a control cabinet is working and reachable or not. For example, if an emergency closing of a valve is required, the operators need to be able to do this fast without any troubleshooting. Reasons for a failed connection to IA Hecon could for example be a power loss, error in the PLC or GSM related malfunction. To make sure IA

Hecon is working, the connection and function needs to be monitored. If the connection fails the system need to inform the operator that something is wrong, the operator can thereafter take necessary action to correct the problem before the valves need to be operated. A very simply implementation could be that on given intervals connect to IA Hecon and read a variable that is updated from time to time, for example the system clock.

3.6.3 Minimizing data traffic

Minimizing the data traffic is good for several reasons. One reason is that data traffic cost money. Another that more data could mean slower response times since the mobile station must occupy air resources for longer.

Two obvious ways to reduce the traffic are:

- Lower the sampling rate or sample only when needed. For example logic can be implemented in the remote unit such that data is only sent to the supervisory system when specific levels are reached.
- Lower the bit depth of signals. It may be sufficient to use 8 bits instead of 32 bits.

When traffic is sent over a network (GPRS, Ethernet etc.) overhead data is added to the actual data to achieve things as routing, fragmentation and error correction. The amount of overhead depends on what kind of network and what protocols are used. If the IP protocol is used for routing this will add overhead. If also the TCP protocol is used this will add even more overhead. If a custom data protocol is used it may be possible to optimize the overhead data, but if a vendor specific protocol is used, such as the Siemens S7 protocol, the overhead is impossible to optimize.

3.6.3.1 GPRS and TCP/IP overhead

When a GPRS operator charges the customer this is normally done per 1 kilobyte (kb) data. However, the actual useful data is as described above not 1 kb.

GPRS uses IP for routing (for example data about the source address and destination address) which means that the overhead of IP is impossible to escape. There are two versions of IP, IPv4 and IPv6. IPv4 normally adds 20 bytes of overhead while IPv6 normally adds 40 bytes of overhead. See figure 3.10.

Above IP the most common protocols to use are UDP and TCP. These are for transmission control and adds among other data about the source port and destination port. In normal cases TCP adds 20 bytes of overhead. See figure 3.11.

Therefore, total 40 or 60 bytes data is ”wasted” on IP and TCP. This means that if the length of the actual data is very small, the efficiency is poor.

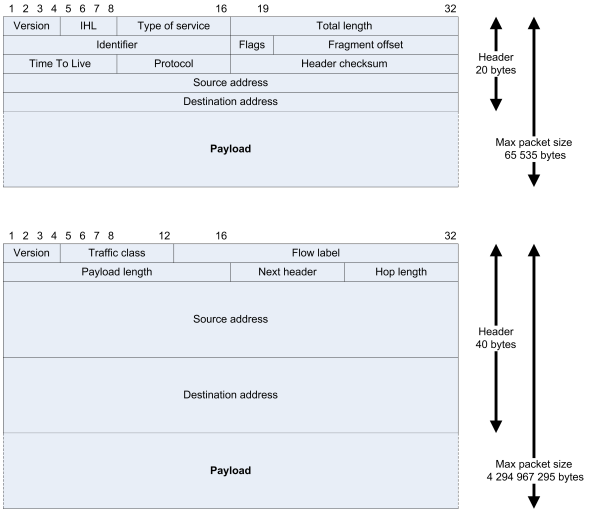


Figure 3.10: IP packet

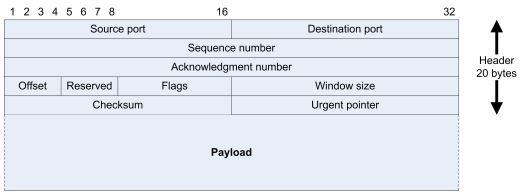


Figure 3.11: TCP packet

3.6.4 Security

The GSM and UMTS networks are used on daily basis by millions of users. Since a wireless network unlike a fixed network (which offers some level of physical security), could be listened to by anyone with a receiver it's arguable that reasonable security measures are implemented to ensure privacy.

The GSM and GPRS networks use encryption to protect privacy. For GSM the ciphers are called A5/1 and A5/2 and for UMTS the cipher is called A5/3. A5/1 is used in Europe and the United States. A5/2 is a weakened cipher for certain export regions (security in the GSM system, Wikipedia). The A5/1 cipher has repeatedly been shown insecure and is constantly under attack. Therefore to further improve security in a wireless system more techniques are used. The GSM encryption is simply not enough.

A VPN (virtual private network) is a computer network that lies above an underlying computer network. It can be envisioned as a secure pipe in a pipe where the outer pipe most often is an Internet connection. Most common are secured VPNs which means that all data sent through the tunnel is encrypted using strong algorithms. There are also so called trusted VPNs.

For professional applications it's common to place the mobile station, or a set of mobile stations, in a virtual LAN (VLAN) which a server have access to through a virtual private

network (VPN) tunnel. This means that the mobile stations aren't public and only can be accessed by using the VPN tunnel. By this hackers can't access the mobile stations before breaking the actual VPN, or the computers having access to the VPN.

There are in practice four ways to implement communication with a mobile station: 1) No security, only GSM/UMTS encryption, 2) Secured VPN tunnel to operator VLAN, 3) Secured VPN tunnel to mobile station and 4) Trusted (hybrid) VPN

Alternative 1 is seldom used for critical M2M applications. If used, the mobile station would be exposed public on the Internet, making it an easy target for potential hackers. Alternative 2 is very common. To communicate with the mobile station, a VPN tunnel must be setup to the operator VLAN. When connected, all mobile stations placed in the VLAN are reachable. Normally, each customers mobile stations are within the customers specific VLAN. The third alternative is to have a VPN server in the actual mobile station. This would make all communication between the mobile station and the host encrypted, but the mobile station would as for alternative 1 be public on the Internet.

Even if a hacker fails to gain access to a mobile station in case of alternative 1 and alternative 3, the mobile station can be still be attacked using so called denial-of-service attacks (DoS attacks). A DoS attack is an attempt to make a computer resource unavailable to its intended users and could if successful prevent the mobile station from functioning efficiently or at all, temporarily or indefinitely.

The fourth alternative is not that common. This is for extremely critical M2M applications. In this case the operator provides the customer with own paths through their networks to ensure that customers' traffic is routed over a trusted path. Companies who use trusted VPNs do so because they want to know that their data is moving over a set of paths that has specified properties and is controlled by one or a trusted confederation of providers. This allows the customer to use their own private IP addressing schemes, and possibly to handle their own routing. Note that it is usually impossible for a customer to know the paths used by trusted VPNs, or even to validate that a trusted VPN is in place; they must trust their provider completely.

Chapter 4

IA Hecon wireless

By combining the technologies described in chapter 3, there are countless numbers of different possible solutions for remote controlling IA Hecon. Depending on the situation where IA Hecon is supposed to be used, some are better than others. A good solution for one place or customer could be bad for another. If the goal would have been to develop a solution fitting one special case for one specific customer, the pre-study and implementation would have been rather fast and easy. However, the goal for this thesis is not to develop that kind of solution. Instead, it is to find a general solution that fits as many customer cases as possible, and in the end, become one single product. Another important aspect is that it must be possible for Industriarmatur to sell and handle the product without too much customization for each customer.

4.1 Possible solutions

Based on the assumptions stated in chapter 2.2, chapter 2.3, and the technologies treated in chapter 3, a couple of possible solutions where selected for further research. In figure 4.1 the different system scenarios and possible solutions are sketched up. Four of them will be further investigated in this chapter. Common for all solutions is the wireless connection between the control system and IA Hecon.

Based on the reasons stated in chapter 3.5 a GPRS solution was selected as the wireless communication link. The most important arguments for using a GPRS solution is the good network coverage, good security and technically acceptance it has in this type of applications and business. The GPRS connection can be accomplished by adding a GPRS modem or a M2M platform in the IA Hecon cabinet. To secure the communication from the control room to the M2M operator a VPN connection is used. VPN is a common solution for M2M applications, recommended by the M2M operators and commonly used by Industriarmaturs customers in the industry. Reason for choosing GPRS instead of using SMS as a data carrier is the better possibilities to add extra features in the future and lower data traffic cost.

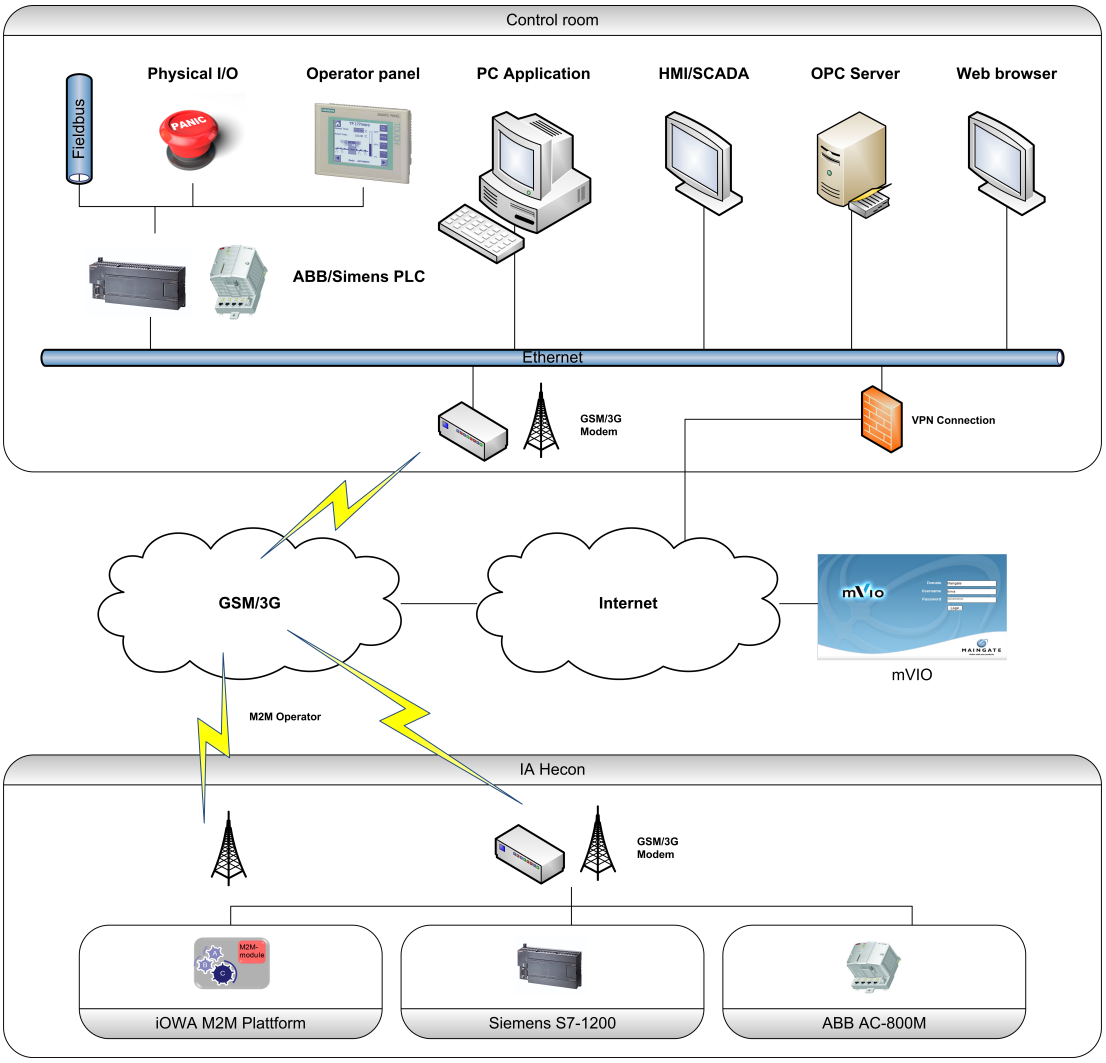


Figure 4.1: Possible solutions

4.1.1 SCADA, direct drivers and OPC

SCADA connection using a direct PLC driver, or via an OPC server with a PLC driver, is a very common way to communicate with PLCs. With a directe PLC driver in the SCADA system or an OPC Server for the specific PLC, it is possible to connect the control system to the PLC. This is a very common used method in the industry, perhaps the most used. A big advantages with this solution, when the system is installed and configured, is that IA Hecon would be completely integrated in the customers control system with well accepted standards.

However, it has some consequences. As mention before, there exist many different customer control systems. This make almost each customer a unique case. Each customer needs to configure their system to work with IA Hecon and to handle the GPRS connection and its related issues mentioned in chapter 3.6. Also, if not the correct direct PLC driver/OPC server exist within the customers system, investment in new licenses and drivers is required, which

could be costly.

Wonderware, a large manufacturer of SCADA systems, has a very elegant solution for maximizing re-use of common components. This concept is called component object-based systems. By creating a SCADA template object, the development and integration at each customer system goes much faster. By creating a template one time, the template can be re-used over and over again. For example, Industriarmatur could develop a template 'IA Hecon' object. The template includes all the necessary settings and functionality needed to control and monitor the valves, see figure 4.2. The customers import the 'IA Hecon' object into their Wonderware SCADA system, and enter the required installation parameters specific for each IA Hecon cabinet.

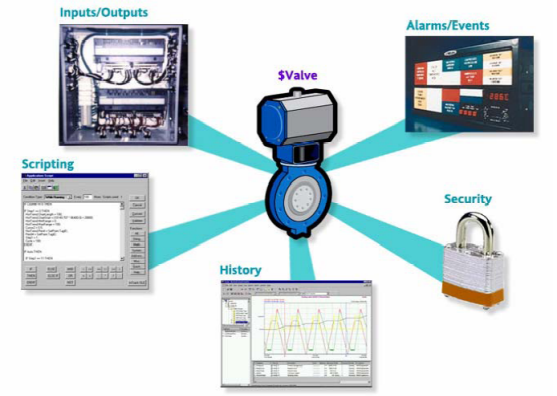


Figure 4.2: Component object-based SCADA

The main problem with the component object-based system is that it is not a standard; it is only working in Wonderware SCADA system, which is not very common in the district heating industry. The development of component object-based systems in the SCADA industry is quite new, but in the future it might get standardized which in turn could make it possible to exchange objects between different SCADA systems (Invensys Wonderware, 2010).

SCADA systems using a direct PLC driver or communicating via an OPC server with a PLC driver is not created specifically to communicate over a GPRS communication link. This makes it difficult to handle the problems associated with this. Especially it is hard to keep the data traffic low because it is impossible to specify how the data packets are sent and received.

To monitor the connection most of the SCADA/OPC software's have built in functions to do this. For example Kepware's OPC server has a ping driver, that in a reliable way monitor devices status. If a device does not respond in a number of tries, the system trigger an alarm to inform the operator (Kepware, SNMP OPC Server, 2010).

4.1.2 Stand-alone software with TCP communication

Another possible solution would be to develop a stand-alone software described in chapter 3.3.4 with a TCP socket connection described in chapter 3.4.3. The application would com-

municate with IA Hecon over an own developed protocol. Based on the PLCs currently being used this could be done by implementing a native TCP socket connection or ISO on TCP for the Siemens PLC. In the PLC the logic to interpret the protocol needs to be implemented in ladder logic. The use of an own protocol make it possible to minimize the data transferred and thus keep the transferred data at a minimum.

The main concern is how well Industriarmaturs customers cope with the idea of having a new stand-alone software from a small company that is unknown regarding development of software applications. It can be hard to convince the customers to feel trust in this, for example in turns of security and stability.

Otherwise the solution has great potential. It is possible to develop an efficient application that can work very well for this specific purpose. Due to use of TCP/IP for communication, the implementation in software is rather easy and follows standard communication methods. The choice of software and hardware is very flexible and it is possible to integrate the communication protocol in the spectrum from a small micro-processor all the way to a standard web-server. Based on the author's earlier experience and knowledge, a good candidate for the software language used for this solution would be C#. Also, this solution doesn't depend on any other software or licenses (like an OPC server or SCADA system), so the installation cost can be kept low.

4.1.3 mVio Internet SCADA

The third solution is to use the mVio Internet SCADA systems from Maingate, as described in chapter 3.3.3. Figure 4.3 shows the architecture of this solution.

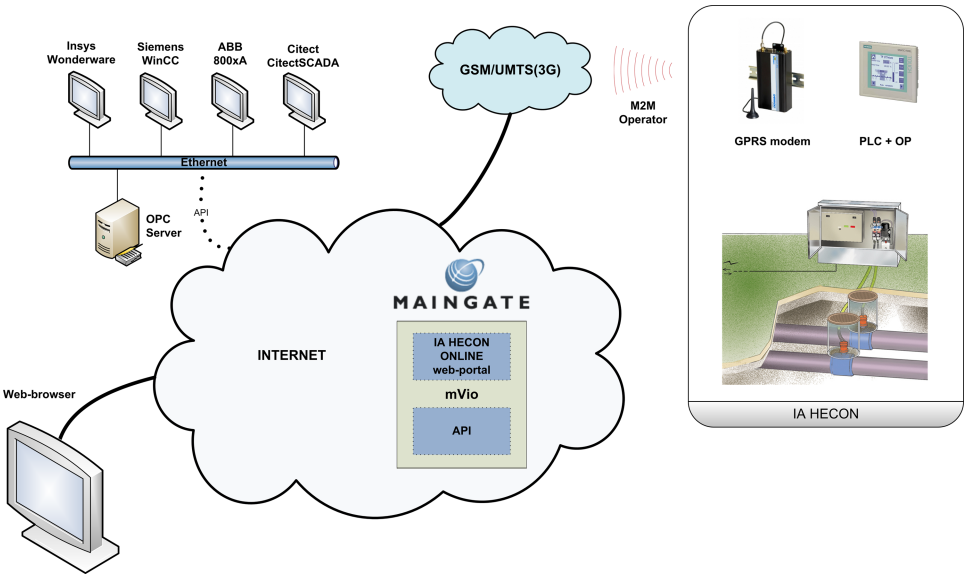


Figure 4.3: mVio Internet SCADA, architecture

First of all, the mVio system needs to be customized to fit IA Hecon. This is done in two parts, first the web portal need to be configured so the customers can overview and maneuver

their valves in a convenient way. This is the part of the system that the customer sees. It is also possible to define specific tasks (open/close valves) and for example create reports. This customization is done by Maingate. Figure 4.4 shows a view from an example portal.

The second part is the communications between mVio and the PLC. Somehow the PLC needs to communicate with mVio, and this needs to be customized in mVio to fit the specific PLC. Together with the engineers at Maingate a specification of the communication is put together. This specification defines how, what and when to communicate. Unfortunately, mVio can only communicate pure TCP/IP traffic and therefore, put higher demands on the logics in the PLC. If mVio could have talked any of the most common fieldbuses the implementation phase in the PLC would have been much easier and faster. Instead, the PLC needs to be programmed to match the mVio protocol specification, which is more complicated.

When these two parts are done, the system is ready to use from a standard web browser. The main benefit is the very easy customer implementation phase. Basically, IA Hecon can be built and configured at Industriarmatur and delivered to the customer, and from day one the customer can maneuver the valves from a web browser without any need of implementation. Also, is is possible for the customers to integrate IA Hecon in their own control system. By setting up a VPN connection to Maingate and implementing the mVio API in the control system, the customers can control and monitor all functions from their own system. The benefit with this, is that mVio act as a layer/gateway between the control system and the device, and takes care about all the problems associated with a GPRS connection.

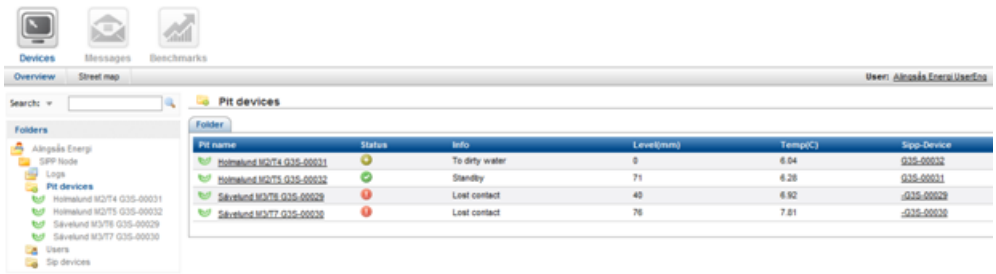


Figure 4.4: mVio Internet SCADA, web browser page

Compared to earlier presented solutions, this and the next solution takes away a lot of the integration, support and maintenance work from the customer. Instead it rely on the mVio Internet SCADA system and the functionality of IA Hecon. In a normal integration project, the time and cost for implementing a M2M system into a customer system is often very high due to the complexity and the number of issues needed to be handle correctly. By using an out-of-the-box solution a lot of money can be saved in the implementation budget, but it could be hard to convince the customer that this is the case. The pay-off for this is that Industriarmaturs customers have to pay a monthly fee to maintain the system functioning. Instead of selling a product, this is more like a offering a service.

4.1.4 iOWA M2M platform + mVio Internet SCADA

The fourth possible solution is to use a M2M platform, designed to simplify rapid development of new M2M applications. iOWA AB is a company located in Gothenburg and has a M2M platform developed to communicate directly with mVio, Maingate's Internet SCADA system. iOWA have some very successful M2M products that are similar to IA Hecon, for example SIPP-node^{TM}.

SIPP-node is a product developed by iOWA that automatically removes and clean water from transformer pits. SIPP-node continually monitors the water level in the transformer pit and when needed, the water is cleaned and removed from the pit. The clearing process of the water is done by removing particles and oil using a filter. When the process is completed, the device sends documented information via the GSM network to SIPP-Warehouse^{TM}on the Internet, which is a modified version of mVio. In SIPP-Warehouse the SIPP-nodes continually display their status, call for help when needed and can be upgraded to the latest functionality and software. Reports, alarms and information can be automatically sent via e-mail, FTP or SMS.

This solution is very similar to the previous solution, as shown in figure 4.4. If the iOWA M2M platform is used with IA Hecon, the M2M platform replaces the GPRS modem. Some type of communication between the PLC and platform need to be developed, either using serial or Ethernet communication. The communication between the M2M platform and mVio is already developed and does not need any reconfiguration at all, which is a great advantage. The implementation in mVio is slightly easier compared to the solution presented in previous chapter.

The main advantage with a M2M platform is that most of the functionalities are already included and the development to customize it to communicate with a new device is rather fast. The M2M platform has basically all the functions included to compensate for the issues discussed in chapter 3.6. For example, if a device is unreachable, mVio can send data using SMS, which increases the reliability and makes the system more failsafe.

In a not too far future, it would be possible to completely replace the PLC in IA Hecon and let the M2M platform from iOWA handle all the functions in IA Hecon. This could be done by adding an I/O module to the M2M platform and incorporate all the logic into the microcontroller of the platform. There are many benefits with this. First of all, it is a big cost saver. Secondly, the development and support of one device instead of two is a lot more efficient.

4.2 Selected concept solution

The objective of this thesis is to find the best possible solution how Industriarmatur should develop and sell a wireless version of IA Hecon. This might not always be the same as the best technical solution. The circumstances, the prerequisites and the company profile has a great influences on the selected solution.

Based on Industriarmaturs current organization and business model, the product has to be easy to integrate in most of the customers systems. To be able convince customers about the product, it has to be easy to explain by persons to persons without great technical knowledge. Due to the number of different control system on the market, it is almost impossible to know how the integration to each of them works without getting into too detailed technical discussions.

The solution built on a SCADA, direct drivers and OPC (chapter 4.1.1) technology put a lot of responsibility on the customer and the integration is cumbersome. Due to the lack of customers having systems for handling communication over GPRS this could be really difficult and a lot of customization at each system is needed.

The solution based on a stand-alone software with TCP communication (chapter 4.1.2) as the control system is from a technical perspective a very good solution. It has all the potential to work in a very efficient way with high stability, low data usage, cost effeicient and easy to install. There is really only one big concern; it is a stand-alone software outside the customers normal control system, and it will not be able to integrate it into the control system without integrating an OPC server or OPC client in the stand-alone software. So, if the customer absolutely requires that the control of IA Hecon is part of their control system, this is not a good solution. Another concern could be that Industriarmatur as a software developer is a too small actor, not able to keep the software updated.

The third and fourth solutions use mVio as the control system, either with a GPRS modem or with iOWAs M2M platform in the control cabinet. Compared to each other, from the customer point of view, they are almost identical. From Industriarmaturs point of view, they are quite different. For the third solution, a lot of development is required and it is a long road to get to a fully working product. For the fourth solution, most of the development is already done, only minor customization is needed to bring the product online and working.

The main advantage with these two solutions is the ease of using them. In the simplest case, the user only needs a web browser to monitor and control all the valves in their system without any concerns of the sometimes unreliable GPRS communication. The mVio system has the ability to take care of all this problems. If the customer wants to integrate it together with their current control system, this is done via an API that exchange information between the two systems which is a lot easier than developing a completely new system.

4.3 Selected prototype solution

The different solutions were presented to Industriarmatur after about half of the time of the project. Based on this Industriarmatur decided to establish a business relation together with iOWA to develop the suggested solution described in chapter 4.1.4. Since the delivery time of the M2M platform was beyond the time frame of this thesis, it was impossible to use this for the prototype. It was also questionable if the small integration part between the PLC and the M2M platform was of enough technical level to be the foundation of this thesis.

Because of this, it was decided that the prototype was to be developed based on the solution in

chapter 4.1.2. To get some use of this prototype for the later iOWA M2M platform project, it was decided that the communication protocol between the C# application and the PLC should be implemented based on iOWA's protocol definition to later be used when communicating with the M2M platform. In the next chapter the construction of the prototype is described in detail.

Chapter 5

Prototype construction

To be able to demonstrate the wireless communication capabilities of IA Hecon to customers, a prototype was built. This is described in this chapter. Because of time constraints from Iowa, the prototype could not be built as the proposed solution. Instead the prototype was built using technologies described in chapter 4.1.2, stand-alone software with TCP communication. Even though the prototype differ from the proposed solution, the two solutions share core technologies. Both use TCP/IP and the same communication protocol.

5.1 The old prototype

The old IA Hecon prototype was based on a Siemens S7-200 PLC with an I/O module to extend the number of inputs and outputs. An operator panel (HMI) was connected via a serial RS-232 interface.

Since the PLC only had a single serial RS-232 interface, the possibilities for external communication was limited. The only way to connect a GSM modem was to equip the PLC with a communication module. However, since the PLC with development environment is an outgoing product, and the cost of the communication module was rather high, it was decided to replace the PLC with its successor.

5.2 Hardware

The new prototype is based on a Siemens S7-1200 PLC. The old operator panel had to be replaced because of compatibility issues. For wireless communication the prototype has been fitted with a GSM modem.

Besides this the new prototype use the same hardware as the old prototype. This means the same hydraulic power pack, the same direction valves and the same valve position indicators etc.

5.2.1 PLC

Quite early in the project the reasons to upgrade the PLC to a new became clear. The reasons were many, first of all, the old PLC lacked communications options. Second, the S7-200 is a product that is about to get phased-out in the near future and therefore Siemens doesn't recommend to use it in new projects.

When selecting the new PLC available models and brands were investigated. Due to the reason that many IA customers use Siemens systems, a PLC from Siemens was preferred. The selected Siemens S7-1200, see figure 5.1, is a brand new PLC introduced in 2009 and is the successor of S7-200. S7-1200 is a powerful PLC with a modular design, possible to upgrade with I/O and communication modules. It has an integrated Ethernet port with support for a number of protocols. The development environment, SIMATIC STEP 7 Basic, is used to configure both the PLC and the HMI panels available. This is a great benefit and required two separate development environments earlier.

The price of the PLC is comparable with S7-200 even though the performance and features have improved.

Figure 5.1: Siemens S7-1200

The most common scenario for IA Hecon is that 2 valves are to be controlled. The basic version of Siemens S7-1200 can provide those I/Os without the need of an extra I/O module, which results in a great cost reduction compared to the old PLC, which required an extra I/O module.

5.2.2 HMI

Together with the new PLC a new operator panel was ordered from Siemens. KTP 600 Basic color PN is a 5.7 inch color touch screen with 6 tactile keys, see figure 5.2. It has an Ethernet

connection which gives new and useful capabilities for IA Hecon. First of all, it makes the connection to the PLC a lot more flexible and easier. With a standard network cable and a switch it's possible to connect one or many panels to one or many PLC's. It is even possible to route the traffic over for example Internet, a LAN or a GPRS connection, which give the possibility to place the operator panel in the control room and maneuver IA Hecon remotely.

Figure 5.2: Siemens KTP 600

5.2.3 GPRS modem

A couple of different GPRS and 3G modems were investigated during the project. The basic requirement was an Ethernet port to connect the modem to the PLC. Since the application is very lightweight in the sense that very little data need to be sent there were in practice no performance requirements. Most GPRS modems with an ethernet port can be thought as a normal router used to share an Internet connection at home, the only difference is the wireless GSM connection used to connect to the Internet.

Because of the issues related to GSM connections as described in chapter 3.6.1, functionality to handle these issues was looked for. Some functionality examples are periodic Ping and modem restart.

CalAmp LandCell-882 GPRS was selected as GPRS modem. See figure 5.3. The modem has Ping and restart capabilities and can be configured from a web browser. The modem comes with an external antenna for good coverage.

5.3 Software

The heaviest part of the construction of the prototype was the software development. The S7-1200 development environment is completely new and lacks the possibility to import software

Figure 5.3: CalAmp LandCell-882 GPRS modem

from the S7-200 development environment (as of the date of this thesis). Due to the new port and memory structure the old PLC software was needed to be completely rebuilt.

Besides this, new functionality was developed for communication over TCP/IP with the C# PC-application. For this a common communication protocol was defined to fit both the prototype and the iOWA M2M platform.

5.3.1 Communication protocol

The communication protocol is the carrier of the data that is sent between the PLC and the control system software. As mentioned earlier, it was a desire that this protocol should be used later when connecting the PLC and iOWA's M2M platform in the proposed end product. Together with the technical engineers at iOWA this was discussed and defined to suit all.

iOWA suggested a dynamic tag based communication protocol built up as plain text with settings and values. For example:

<plc message 1.0><openvalve1=1><openvalve2=1><...><...>

This kind of protocol was investigated, but quite soon it was found out that this is not a suitable language for a PLC. Parsing and building messages like this is quite complex for a PLC and would use too much of the CPU resources. Also, in terms of data usage, this is not very efficient. Each character in the message takes 1 byte. The example message above would be 55 bytes long.

Instead a more fixed protocol was suggested by the authors. The suggested protocol is made out of bits in a predefined order, which is more suitable for a PLC and uses a lot less data. The protocol consist of two types of messages, one is from the control system to the PLC, and the other one is from the PLC to the control system.

The message from the control system to the PLC is 6 bytes long (see Appendix A) where the first byte is a command and the rest is data. The commands concern opening, closing and stopping valves, status retrieval and alarm reset functionality.

The message from the PLC to the control system is 64 bytes long (see Appendix A).

The PLC will send one or more status messages when:

- The control system asks the PLC for status and the PLC respond with a status message
- The control system sends an action (e.g. open valve 1) and the PLC continuously every third second sends status messages until finished
- One of the following PLC events occur
  - Stop valve request from operator panel
  - Valve x - Runtime overrun
  - Valve x - Jammed
  - Alarm - pump x, no pressure
  - Alarm - frequency inverter x
  - Alarm - acknowledge from operator panel
  - Click on 'Sänd Status' on the operator panel 'Inställningar' screen

If any other product (like iOWAs M2M platform or a customer control system) implement the protocol as defined, and use a TCP/IP socket connection, it's possible to communicate with the PLC. This is a great benefit when IA Hecon is to be connected to the iOWA M2M platform.

5.3.2 PLC and HMI software

As the old PLC software had all the basic functionality, each function was first analyzed in the old prototype, and thereafter implemented in the new development environment.

Unfortunately, ladder was the only available PLC language for S7-1200. Ladder is a very simple programming language and has a lot of shortcomings. For example, it is not possible to make an object for each valve, instead, the whole code has to be copied and slightly modified for each single valve. This makes the code large, clunky, hard to understand and time consuming to update and maintain. Figure 5.4 shows a part of the PLC ladder software. A big advantage with the new development environment is the possibility to use tags for all variables and ports on the PLC, instead of using the actual hardware addresses.

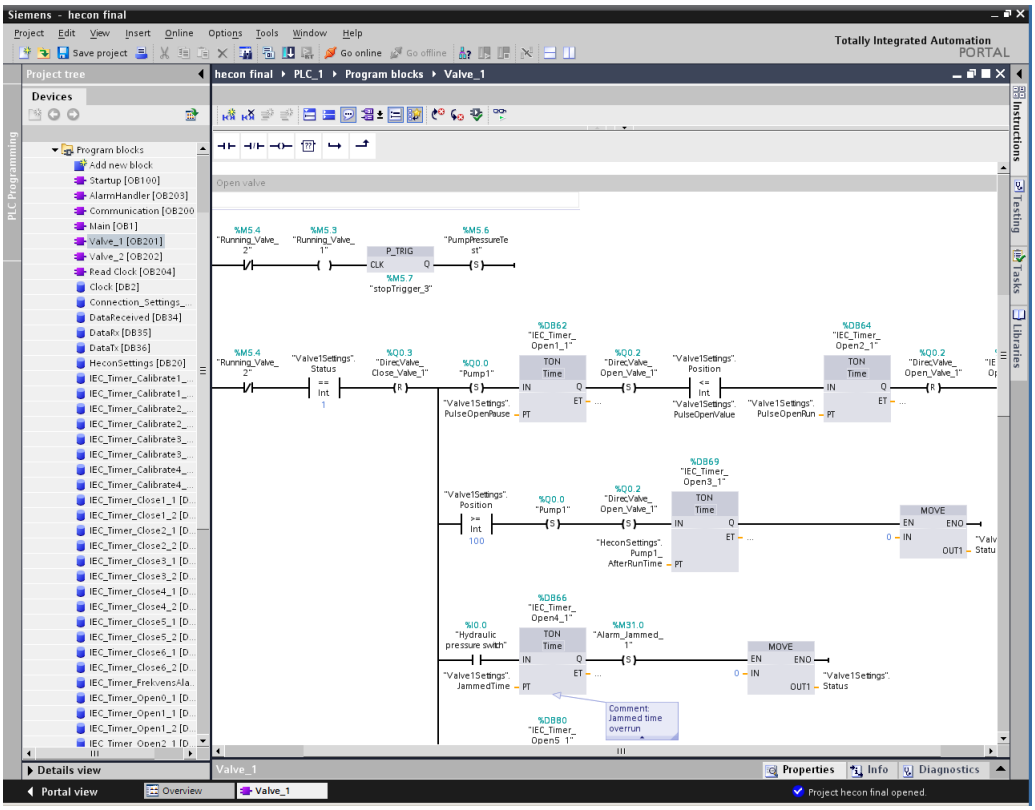


Figure 5.4: PLC ladder code example

In the beginning it was very cumbersome to develop the functions in ladder and understand the development environment. In the authors opinion very basic problems sometimes was very hard to solve, if even possible at all. For example, it was impossible to enable/disable buttons in the HMI based on a given variable. Another example is the big communication module that was developed to parse and build the communication protocol messages.

A desire was to make one single version of the PLC program with the possibility to configure it for different numbers of valves. Based on the author's knowledge, this was impossible due to the limitation in the ladder and HMI development software. Instead, each setup for the PLC need to be changed and configured in the development environment, and thereafter downloaded to the corresponding PLC and HMI.

5.3.3 C# PC-application

The prototype PC application was developed in Microsoft's C#. C# is a programming language developed within Microsoft's .Net initiative and is very suitable for rapid application development. Besides C#, JAVA could be a candidate as well as C++. Since both authors have very good knowledge of C#, and rather poor knowledge of both JAVA and C++, the choice of programming language was simple.

To fulfil the goals for the prototype, as specified in the beginning of the thesis, the application

development was carried out with respect to these. The first function that was developed was to be able to send and receive TCP packets between a PC and the PLC over a LAN. After this was successful the defined protocol was implemented in the C# application as well as in the PLC.

The application has the functionality needed to fulfil the prototype goals and thereby making it possible to demonstrate IA Hecon's wireless communication capabilities. However, it lacks some features required in a live scenario. If the application is further developed this would be possible. This is discussed in chapter 7. Figure 5.5 shows the main screen of the PC-application.

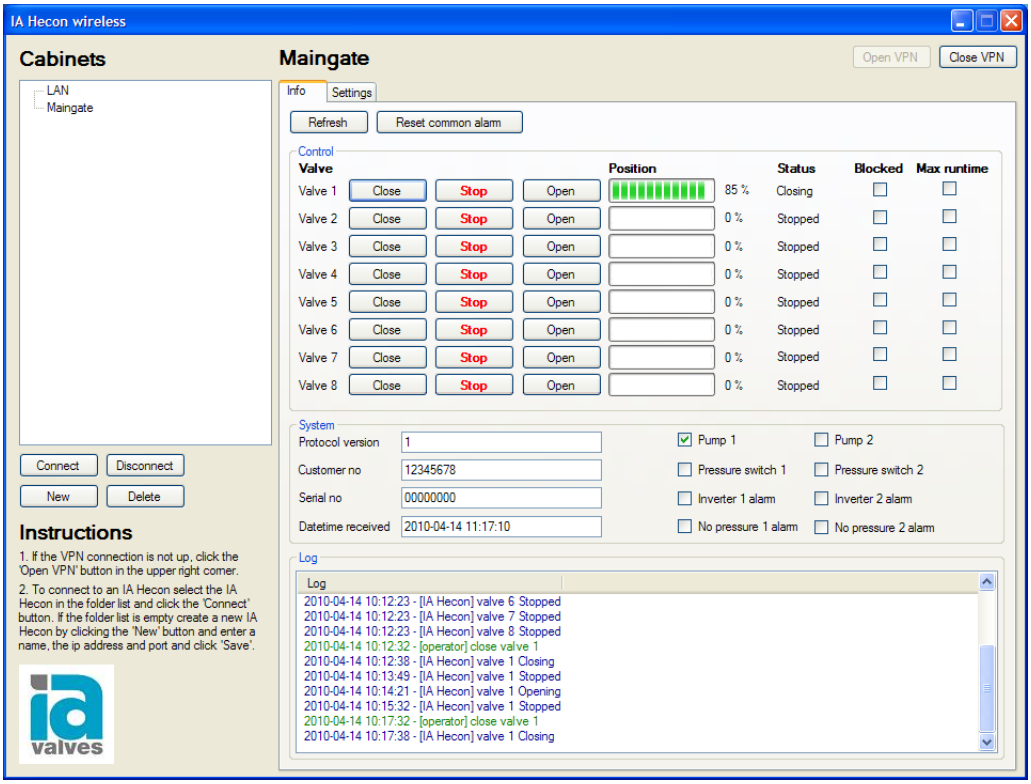


Figure 5.5: C# PC-application

The functionality of the C# application has been placed in separate classes. This makes the application code easier to follow and it also simplifies further development. Important classes are:

'Comm.cs' handles all communication

'VPN.cs' is used for integration of the cisco VPN client and makes it possible to execute cisco VPN client commands from within the C# PC-application

'IAHecon.cs' contains the 'IAHecon' and 'IAHeconDatabase' classes. An 'IAHecon' object store the name, IP-address and port number of an IA Hecon cabinet. An 'IAHeconDatabase' object is used for handling an .xml database of IA Hecon cabinets

'Test.cs', 'TestIAHeconConnect.cs', 'TestIAHeconStatus.cs', 'TestPing.cs', 'JobTestIAHeconConnect.cs', 'JobTestIAHeconStatus.cs' and 'JobTestPing.cs' are all classes for communication tests carried out during the development. These are not used for other purposes and therefore not necessary for the valve operation functionality.

There are also files related to the user interface. These are:

The 'frmControl.cs' with related designer files, describes the user interface with methods for updating the user interface and triggering methods in for example the 'Comm.cs' file.

The 'frmConnectionTest.cs' with related designer files. This is used for the communication tests user interface.

The 'frmDebug.cs' with related designer files, describes the debug window which can be reached by clicking the 'Debug' button under the 'Settings' page. This window is good to have when developing. In the debug window raw data is sent and received.

The application is multithreaded and has a thread for the communication and another thread for updating the user interface. The debug window also runs on a separate thread. This makes the application very responsive and ensure full functionality. When programming multithreaded applications it's very important to make sure that the different threads write and read shared data in a correct manner. For example, when receiving data the receive data buffer is locked until finished so that the thread updating the user interface doesn't read wrong data.

Chapter 6

Result

This chapter presents the results of the prototype solution and the concept solution.

Early in the project the goals for the prototype was stated as:

- Develop the current prototype of IA Hecon to enable wireless remote control
- Develop a PC-application that can control and monitor IA Hecon

The old prototype at IA was upgraded with the new PLC, the new operator panel and the GSM modem. All the hardware as pumps, acctuators and senors was connected to the input and output of the new PLC. Some fine tuning of logic and timmers were done in the PLC software to work as the old prototype. All functions were carefully node and system tested.

The developed prototype fulfills the goals. The developed system has in theory no geographical distance limit, basically it can be used all over the world. The PC-application can at this moment control and monitor the position of up to 8 valves. As the application was developed in a object oriented way this can be further expanded without too much effort.

To perform communication tests the PC-application includes test functionality. Three tests were defined. The first test, 'TestPing', pings the GPRS modem in the IA Hecon control cabinet. This test tells whether it's possible to reach the GPRS modem. The second test, 'TestIAHeconConnect', test whether it's possible to open a TCP socket connection to the PLC from the PC-application. This test tells whether the PLC itself is reachable. The last test, 'TestIAHeconStatus', opens a TCP socket connection to the PLC and ask the PLC for it's status. The test then expect to receive status data from the PLC. A scheduler makes it possible to run the different tests at predefined times or intervals.

The prototype was tested in a couple of weeks with very good results in terms of performance and reliability. The response time was normally around 300 ms which is better than necessary. As the third status test gives the most input this was setup to connect to the PLC and ask for its status two times per hour during a period of 72 hours. All of the 144 tries were successful.

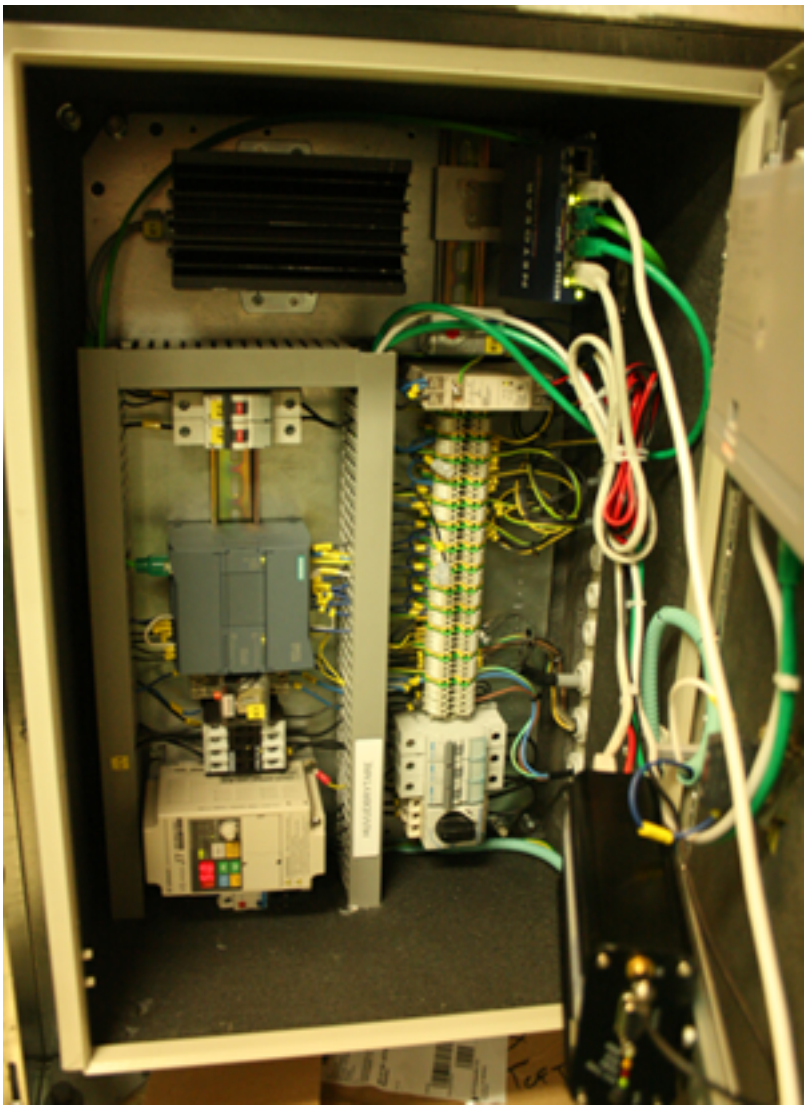


Figure 6.1: Wireless IA Hecon prototype cabinet

The main screen of the new operator panel is shown in figure 6.2.

Functionality that has been successfully implemented in the PLC and C# PC-application are:

- Send commands to IA Hecon
- Receive heartbeats from IA Hecon
- Present current status of the system

Functionality that has not been implemented but desired in a live scenario are:

- Data transmission over SMS as backup in case of GPRS failure

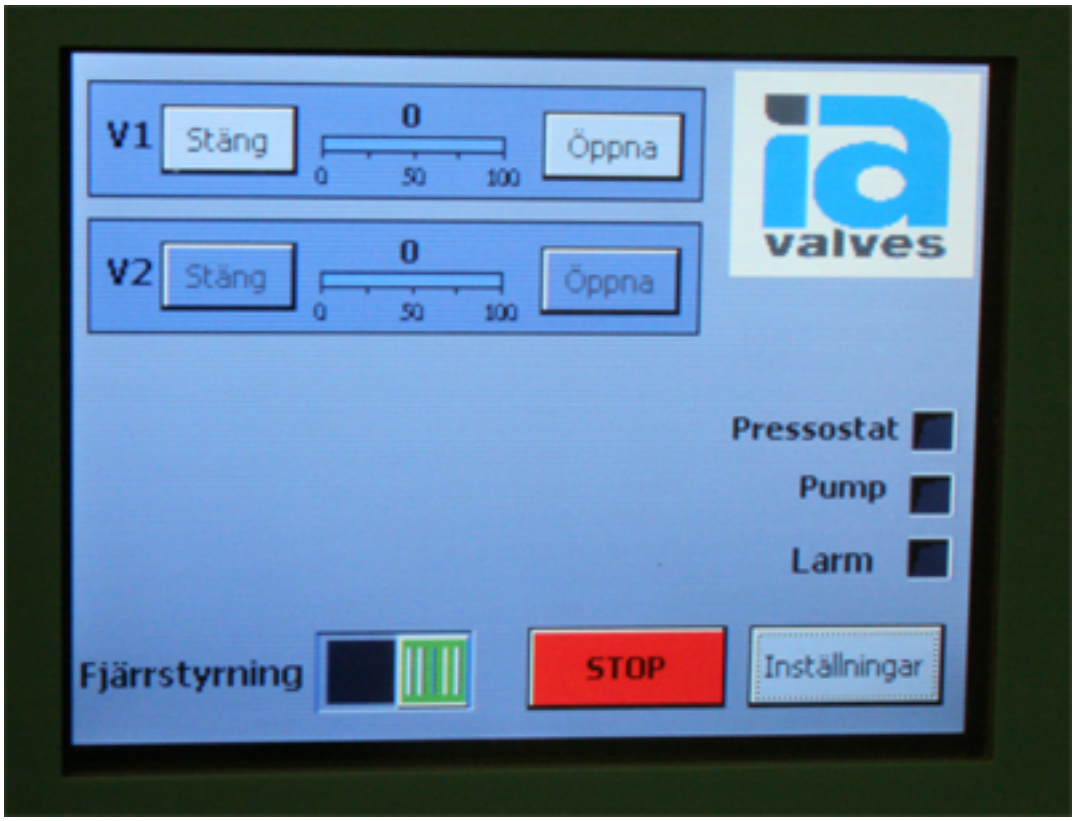


Figure 6.2: Operator panel main screen

- Alarm over SMS/email/etc. if communication with IA Hecon fails
- Alarm over SMS/email/etc. if certain limits are exceeded
- Integrated OPC server to integrate with customer control systems, it could also be enough with an OPC client or by reading/writing data using a text file
- Run the core functionality of the application as a windows service which could reduce the risk that a person by mistake close the PC-application thereby stopping all communication

Some of the functions can be implemented in alternative ways. For example, if a OPC server or OPC client is integrated in the C# application, a customer's main SCADA system could handle the alarms instead of having this functionality in the C#-application. Also, the heartbeat functionality can either be implemented as letting IA Hecon periodically initiate the communication and thereafter send its current status, or by letting the C#-application periodically initiate the communication and ask IA Hecon for its current status.

6.1 Data usage

An important result from the prototype and the developed communication method is how much data is consumed during one month of normal usage. This was measured and investigated using Wireshark, a network protocol analyzer software. The traffic consists of two parts, one part associated with monitoring the connection (heartbeat), and the other associated to operating the valves. The first part mainly depends on how often heartbeats are sent. The second depends on how many times the valves are opened/closed.

Our data measurements show that approximately:

- 500 kB data is needed for hourly heartbeats per month (roughly 20% is useful data, 80% overhead is overhead data, and the connection is connected/disconnected at each heartbeat)
- 280 kB data is needed for opening a valve 10 times and closing it 10 times (incl. data for connect/disconnect)

Often a normal subscription agreement with a M2M operator includes 1 MB of data usage per month. With this setup less than 1 MB data usage per month is possible within margins. If for example the heartbeats are sent twice as often, the data usage for this would be 1 MB, or if the valves are only opened and closed 5 times, this will result in approximately 140 kB data.

In figure 6.3, frame number 3 shows a heartbeat packet from the PLC. As can be seen in the figure, the IP packet is 104 bytes long.

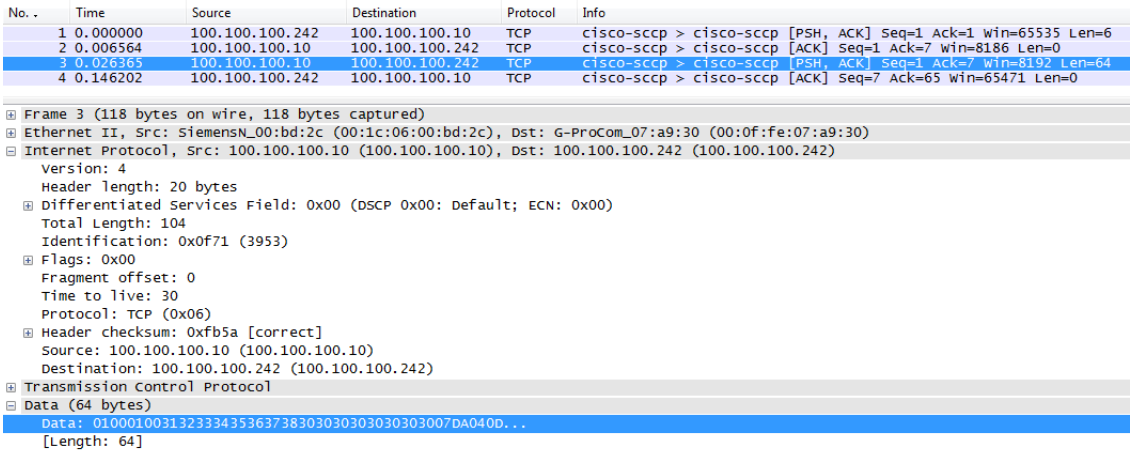


Figure 6.3: Wireshark, heartbeat from PLC

To optimize data usage it's important to be aware of the overhead. Overhead origin from different things. Firstly protocol overhead from IP and TCP, secondly data required to connect, disconnect as shown in figure 6.4. Finally a keep the connection alive packageg is sent. If a TCP socket connection is open the Siemens S7-1200 PLC will every 30 seconds send

a keep-alive packet of 41 bits. The PC-application responds to this with a ACK packet of 40 bits, see figure 6.5. This means that if a TCP socket connection is left open for 24 hours for a month, approxmately 854 kB keep-alive data is needed just for this, see equation 6.1.1. The keep-alive packet from the PLC can hopefully be configured to be sent less often. The Siemens support were contacted but no answer has yet been received. Another way to work around this issue is to disconnect/reconnect each time. This will on the other hand lead to extra data associated with the connection packages sent during handshake between the PLC and the C# PC-application.

$$\frac{2 \cdot 60 \cdot 24 \cdot 30 \cdot 81}{8 \cdot 1024} = 854\ kB \tag{6.1.1}$$

No.	Time	Source	Destination	Protocol	Info
43	8.098034	100.100.100.242	100.100.100.10	TCP	1168 > 2000 [SYN] Seq=0 Win=65535 Len=0 MSS=1260
44	8.102483	100.100.100.10	100.100.100.242	TCP	2000 > 1168 [SYN, ACK] Seq=0 Ack=1 Win=8192 Len=0 MSS=1260
45	8.102514	100.100.100.242	100.100.100.10	TCP	1168 > 2000 [ACK] Seq=1 Ack=1 Win=65535 Len=0
52	12.180464	100.100.100.242	100.100.100.10	TCP	1168 > 2000 [FIN, ACK] Seq=7 Ack=65 Win=65471 Len=0
53	12.181929	100.100.100.10	100.100.100.242	TCP	2000 > 1168 [ACK] Seq=65 Ack=8 Win=8192 Len=0
54	12.191655	100.100.100.10	100.100.100.242	TCP	2000 > 1168 [FIN, ACK] Seq=65 Ack=8 Win=8192 Len=0
55	12.191721	100.100.100.242	100.100.100.10	TCP	1168 > 2000 [ACK] Seq=8 Ack=66 Win=65471 Len=0

Figure 6.4: Wireshark, connect and disconnect packages

No.	Time	Source	Destination	Protocol	Info
1	0.000000	100.100.100.242	100.100.100.10	TCP	2000 > 2000 [SYN] Seq=0 Win=65535 Len=0 MSS=1260
2	0.012015	100.100.100.10	100.100.100.242	TCP	2000 > 2000 [SYN, ACK] Seq=0 Ack=1 Win=8192 Len=0 MSS=1260
3	0.012056	100.100.100.242	100.100.100.10	TCP	2000 > 2000 [ACK] Seq=1 Ack=1 Win=65535 Len=0
4	29.729058	100.100.100.10	100.100.100.242	TCP	[TCP Keep-Alive] 2000 > 2000 [ACK] Seq=0 Ack=0 Win=8192 Len=1
5	29.729103	100.100.100.242	100.100.100.10	TCP	[TCP Keep-Alive ACK] 2000 > 2000 [ACK] Seq=1 Ack=1 Win=65535 Len=0
6	59.727499	100.100.100.10	100.100.100.242	TCP	[TCP Keep-Alive] 2000 > 2000 [ACK] Seq=0 Ack=0 Win=8192 Len=1
7	59.727547	100.100.100.242	100.100.100.10	TCP	[TCP Keep-Alive ACK] 2000 > 2000 [ACK] Seq=1 Ack=1 Win=65535 Len=0
8	89.733246	100.100.100.10	100.100.100.242	TCP	[TCP Keep-Alive] 2000 > 2000 [ACK] Seq=0 Ack=0 Win=8192 Len=1
9	89.733301	100.100.100.242	100.100.100.10	TCP	[TCP Keep-Alive ACK] 2000 > 2000 [ACK] Seq=1 Ack=1 Win=65535 Len=0
10	119.720327	100.100.100.10	100.100.100.242	TCP	[TCP Keep-Alive] 2000 > 2000 [ACK] Seq=0 Ack=0 Win=8192 Len=1
11	119.720366	100.100.100.242	100.100.100.10	TCP	[TCP Keep-Alive ACK] 2000 > 2000 [ACK] Seq=1 Ack=1 Win=65535 Len=0
12	149.721039	100.100.100.10	100.100.100.242	TCP	[TCP Keep-Alive] 2000 > 2000 [ACK] Seq=0 Ack=0 Win=8192 Len=1
13	149.721078	100.100.100.242	100.100.100.10	TCP	[TCP Keep-Alive ACK] 2000 > 2000 [ACK] Seq=1 Ack=1 Win=65535 Len=0
14	179.723505	100.100.100.10	100.100.100.242	TCP	[TCP Keep-Alive] 2000 > 2000 [ACK] Seq=0 Ack=0 Win=8192 Len=1
15	179.723540	100.100.100.242	100.100.100.10	TCP	[TCP Keep-Alive ACK] 2000 > 2000 [ACK] Seq=1 Ack=1 Win=65535 Len=0

Figure 6.5: Wireshark, connect and keep-alive packages

Another source of data usage is the Ping function in the GPRS-modem. As explained earlier, this function periodically send (and receive) data to keep the GPRS connection alive. It was found that this function used too much data to be implemented in the prototype. The main reason for this is because the longest interval the periodically timer can be set to is 1 hour. Also, the GPRS-modem seem to send 4 pings each time. Each ping takes 84 x 2 (one pong is returned) bytes which yields approximately 500 kB data per month, see equation 6.1.2. To ensure that the GPRS-modem keeps its connection and sometimes restart and reconnect to the GSM network, a restart of the modem was scheduled twice a week.

$$\frac{84 \cdot 2 \cdot 4 \cdot 24 \cdot 30}{1024} = 472.5\ kB \tag{6.1.2}$$

Chapter 7

Discussion

In this chapter we discuss the results in our opinion of this project and the outcome of the prototype. We also make some suggestions about further development of the prototype.

We find that the suggested solution with a collaboration together with iOWA is the very best way for Industriarmatur to make a trustworthy product that will be welcomed and accepted. Due to number of issues associated with GPRS communication it is very important to handle these problems correctly, which we think iOWA's M2M platform has the ability to do.

In the future we think Industriarmatur can get a lot of benefits if they focus on selling a product that solves the general problem instead of selling a customer customized product. To get away from technical discussions regarding PLC fabricate and similar, we think it is important to completely abolish the PLC used today and instead use a own developed "black box" that solves the problem.

The developed prototype and the associated software is at the moment working very good, but it is still a prototype and not quite yet a completely finished product ready to sell to end customers. To take this product from the prototype phase to an end product, it needs to be evaluated and developed together with a pilot customer. For example, feedback is needed about the user interface of the software, the functions and the operator panel. Also, the data traffic and the reliability of the communication link need to be carefully analyzed under real circumstances under a long period of time. Due to the high security and accessibility demands of the product, it is important that all the hardware, functions, and especially the code is carefully analyzed and critical reviewed by an external party.

Early in the project we decided to change the old PLC to a new modern PLC from Siemens. This was a very good decision and it would have been very difficult to complete this project with the old PLC. Also, we hade from the beginning of the project an eager to use a TCP/IP soluion for communication. This was also a another wise dicision witch we are happy that we did.

A thing that is important but hard to get a real good answer to is how the prototype solution and specially the software is accepted by customers. The customers has traditionally a quite

conservative opinion about new techniques and usually do the things as it has always been done. It might be hard to convince them to accept this new way of monitor and operate their valves, especially since Industriarmatur has not developed anything similar and not known of development of complete control systems.

A outcome of this project that wasn't clear and thought of in the beginning is the possibility to use this prototype without a wireless connection. In fact, it's actually a very good solution to use in those cases when a wire to the control cabinet exist. All the uncertainties about reliability and data traffic can be neglected and the system can be setup fast, easy and implemented in the control room without too much fuss. This together with the competitive price of the new PLC and the fact that Industriarmatur at the moment doesn't have a complete solution for doing this this make it an interesting product.

Chapter 8

Conclusion

Today mobile technology is so mature that wireless solutions is found everywhere. Wireless connectivity is not "high tech". From the beginning to the end of this thesis, it has never been the question whether it's possible to find a solution to wireless remote control IA Hecon. The problem has been to find a solution, or the solution, that fits the customers to Industriarmatur and also Industriarmatur themselves.

Because of this, the first weeks was spent to understand what hardware, software and systems that are common in the industry. Many of the customers of Industriarmatur were contacted and we spoke with both small and large customers, managers as well as technicians. This resulted in a clearer view of the problem.

The process control industry is a very large industry. Being a large industry means there are thousands of manufacturers and developers of hardware and software. Trying to find the best solution of how to wireless remote control IA Hecon is therefore a somewhat complicated task.

The goal of the thesis was to find a complete solution that Industriarmatur can sell. If all customers would use system A to wireless remote control IA Hecon, it would be rather simple to implement a good working solution. But when the customers has systems A, B, C and D, the scenario changes. Instead of one solution, there are four solutions. Even if the basics are the same, it would be very hard for Industriarmatur to sell a complete product since all the customers control system must be further developed to support IA Hecon and handle issues regarding the wireless connection.

An installation procedure could be that Industriarmatur sell IA Hecon with GPRS hardware and that the customer itself:

1. Buy, install and setup an direct PLC driver or OPC server with PLC driver that can talk to the IA Hecon PLC

2. Develop an GUI and setup alarms etc in the SCADA (or smaller) control system

3. Buy and setup the mobile subscription
4. Implement connection monitoring and, if needed, the possibility to send data over SMS as backup in case of GPRS failure.

This is a procedure that Industriarmatur can't help with or support since the employees of Industriarmatur lacks the technical skills required. Therefore, even if this a perfectly working solution, it's not in the interest of Industriarmatur.

The proposed final solution could be thought as somewhat unconventional. Outsourcing and letting a third party run a part of the SCADA system is what we know from interviews not that common today. We do strongly believe in this architecture and the advantages that follows. Internet SCADA systems does gain popularity, and can make the customer focus more on its core business.

As the time frame of the thesis was limited, the final solution could not be realized. Therefore, to give the thesis some technical depth, a prototype solution was developed. The prototype is fully working and can be used for demo purposes even though it can't demonstrate the power of the mVio Internet SCADA system. Developing the communication over a TCP/IP socket connection using a custom communication protocol took quite some time since this isn't the standard way to do it.

Bibliography

Poole, Ian. Cellular communications explained: From basics to 3G, Newnes, 1st edition 2006.

Poole, Ian. GSM, Architecture, Protocols and Services, Wiley, 3rd edition 2008.

Blank, Andrew G. TCP/IP JumpStart: Internet Protocol Basics, p 12, Alameda, CA, USA: Sybex, Incorporated, 2002.

Stuart A. Boyer. SCADA: Supervisory Control and Data Acquisition, ISA The Instrumentation, Systems, and Automation Society, 3rd edition, 2004.

Wikipedia, Transmission Control Protocol [Electronic] http://en.wikipedia.org/wiki/Transmission_Control_Protocol, available 2010-10-01

Wikipedia, User Datagram Protocol [Electronic] http://en.wikipedia.org/wiki/User_Datagram_Protocol, available 2010-10-01

Wikipedia, Internet Protocol Suite [Electronic] http://en.wikipedia.org/wiki/Internet_Protocol_Suite, available 2010-10-01

Ericsson, Asia-Pacific key to 50 billion connected devices [Electronic] http://www.ericsson.com/news/100514_key_to_50_billion_connections_20100518110524, available 2010-10-01

Wikipedia, SCADA [Electronic] http://en.wikipedia.org/wiki/SCADA, available 2010-10-01

Wikipedia, Industrial Ethernet [Electronic] http://en.wikipedia.org/wiki/Industrial_Ethernet, available 2010-10-01

Wideco Sweden, RedDetect XTool [Electronic] http://www.wideco.se/produkter/larm/X_Tool_overblick.asp, available 2010-10-01

Bailey, David. Practical SCADA for industry, Oxford: IDC Technologies, 2003.

Park, John. Mackay, Steve. Wright, Edwin. Practical data communications for instrumentation and control, Oxford: IDC Technologies, 2003.

Göteborg Energi [Interview], 2010-05-01

Industriarmatur-ARI AB [Interview], 2010-05-01

Invensys Wonderware [Electronic] http://global.wonderware.com/EN/PDFLibrary/The_Benefits_of_Component_Object_Based_SCADA_and_Supervisory_System_Application_Development_White_Paper.pdf, available 2010-10-01

Kepware, SNMP OPC Server [Electronic] http://www.kepware.com/Spec_Sheets/Ping.asp, available 2010-10-01