



## Calhoun: The NPS Institutional Archive

---

Theses and Dissertations

Thesis Collection

---

2004-09

# An architectural framework for describing Supervisory Control and Data Acquisition (SCADA) systems

Ward, Michael P.

Monterey, California. Naval Postgraduate School

---

<http://hdl.handle.net/10945/1335>



Calhoun is a project of the Dudley Knox Library at NPS, furthering the precepts and goals of open government and government transparency. All information contained herein has been approved for release by the NPS Public Affairs Officer.

**Dudley Knox Library / Naval Postgraduate School  
411 Dyer Road / 1 University Circle  
Monterey, California USA 93943**

<http://www.nps.edu/library>



**NAVAL  
POSTGRADUATE  
SCHOOL**

**MONTEREY, CALIFORNIA**

**THESIS**

**AN ARCHITECTURAL FRAMEWORK FOR  
DESCRIBING SUPERVISORY CONTROL AND DATA  
ACQUISITION (SCADA) SYSTEMS**

by

Michael P. Ward

September 2004

Thesis Advisor:

Co-Advisor:

Second Reader:

Cynthia E. Irvine

Deborah S. Shifflett

Daniel F. Warren

**Approved for public release; distribution is unlimited.**

THIS PAGE INTENTIONALLY LEFT BLANK

<b>REPORT DOCUMENTATION PAGE</b>			<i>Form Approved OMB No. 0704-0188</i>
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instruction, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188) Washington DC 20503.			
<b>1. AGENCY USE ONLY (Leave blank)</b>	<b>2. REPORT DATE</b> September 2004	<b>3. REPORT TYPE AND DATES COVERED</b> Master's Thesis	
<b>4. TITLE AND SUBTITLE:</b> Title (Mix case letters) An Architectural Framework for Describing Supervisory Control and Data Acquisition (SCADA) Systems			<b>5. FUNDING NUMBERS</b>
<b>6. AUTHOR(S)</b> Ward, Michael P.			<b>8. PERFORMING ORGANIZATION REPORT NUMBER</b>
<b>7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)</b> Naval Postgraduate School Monterey, CA 93943-5000			<b>10. SPONSORING/MONITORING AGENCY REPORT NUMBER</b>
<b>9. SPONSORING /MONITORING AGENCY NAME(S) AND ADDRESS(ES)</b> DON CIO, Presidential Tower Suite 2100, 2511 Jefferson Davis Highway Arlington VA 22202			<b>11. SUPPLEMENTARY NOTES</b> The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.
<b>12a. DISTRIBUTION / AVAILABILITY STATEMENT</b> Approved for public release; distribution is unlimited.			<b>12b. DISTRIBUTION CODE</b>
<b>13. ABSTRACT (maximum 200 words)</b> Two recent trends have raised concerns about the security and stability of Supervisory Control and Data Acquisition (SCADA) systems. The first is a move to define standard interfaces and communications protocols in support of cross-vendor compatibility and modularity. The second is a move to connect nodes in a SCADA system to open networks such as the Internet. Recent failures of critical infrastructure SCADA systems highlight these concerns. To ensure continued operations in times of crisis, SCADA systems, particularly those operating in our critical infrastructure, must be secured. Developing an abstract generic framework for defining and understanding SCADA systems is a necessary first step. A framework can provide the tools to understand the system's functions and capabilities, and how components in the system relate and interface with each other. This thesis examines and describes SCADA systems, their components, and commonly used communications protocols. It presents a matrix approach to describing and defining the features, functions and capabilities of a SCADA system. Two small SCADA systems, using industry standard components and simulating real world applications, were designed and constructed for this thesis to provide context for applying the matrix approach.			
<b>14. SUBJECT TERMS</b> Supervisory Control and Data Acquisition, SCADA, Critical Infrastructure Protection, CIP			<b>15. NUMBER OF PAGES</b> 122
			<b>16. PRICE CODE</b>
<b>17. SECURITY CLASSIFICATION OF REPORT</b> Unclassified	<b>18. SECURITY CLASSIFICATION OF THIS PAGE</b> Unclassified	<b>19. SECURITY CLASSIFICATION OF ABSTRACT</b> Unclassified	<b>20. LIMITATION OF ABSTRACT</b> UL

THIS PAGE INTENTIONALLY LEFT BLANK

**Approved for public release; distribution is unlimited.**

**AN ARCHITECTURAL FRAMEWORK FOR DESCRIBING SUPERVISORY  
CONTROL AND DATA ACQUISITION (SCADA) SYSTEMS**

Michael P. Ward  
Captain, United States Marine Corps  
B.S., College of William and Mary, 1989

Submitted in partial fulfillment of the  
requirements for the degree of

**MASTER OF SCIENCE IN COMPUTER SCIENCE**

from the

**NAVAL POSTGRADUATE SCHOOL  
September 2004**

Author: Michael P. Ward

Approved by: Dr. Cynthia E. Irvine  
Thesis Advisor

Deborah S. Shifflett  
Co-Advisor

Daniel F. Warren  
Second Reader

Dr. Peter J. Denning  
Chairman, Department of Computer Science

THIS PAGE INTENTIONALLY LEFT BLANK

## **ABSTRACT**

Two recent trends have raised concerns about the security and stability of Supervisory Control and Data Acquisition (SCADA) systems. The first is a move to define standard interfaces and communications protocols in support of cross-vendor compatibility and modularity. The second is a move to connect nodes in a SCADA system to open networks such as the Internet. Recent failures of critical infrastructure SCADA systems highlight these concerns. To ensure continued operations in times of crisis, SCADA systems, particularly those operating in our critical infrastructure, must be secured. Developing an abstract generic framework for defining and understanding SCADA systems is a necessary first step. A framework can provide the tools to understand the system's functions and capabilities, and how components in the system relate and interface with each other. This thesis examines and describes SCADA systems, their components, and commonly used communications protocols. It presents a matrix approach to describing and defining the features, functions and capabilities of a SCADA system. Two small SCADA systems, using industry standard components and simulating real world applications, were designed and constructed for this thesis to provide context for applying the matrix approach.

THIS PAGE INTENTIONALLY LEFT BLANK

# TABLE OF CONTENTS

<b>I.</b>	<b>INTRODUCTION.....</b>	<b>1</b>
<b>II.</b>	<b>BACKGROUND .....</b>	<b>3</b>
<b>A.</b>	<b>OVERVIEW OF CONTROL SYSTEMS .....</b>	<b>3</b>
<b>1.</b>	<b>Sensors and Actuators .....</b>	<b>4</b>
<b>2.</b>	<b>Remote Terminal Units .....</b>	<b>4</b>
<b>a.</b>	<b><i>Programmable Logic Controllers.....</i></b>	<b>5</b>
<b>b.</b>	<b><i>Analog Input and Output Modules .....</i></b>	<b>6</b>
<b>c.</b>	<b><i>Digital Input and Output Modules .....</i></b>	<b>6</b>
<b>d.</b>	<b><i>Communications interfaces.....</i></b>	<b>7</b>
<b>3.</b>	<b>Master Stations.....</b>	<b>7</b>
<b>B.</b>	<b>SUPERVISORY CONTROL AND DATA ACQUISITION .....</b>	<b>8</b>
<b>C.</b>	<b>PROTOCOLS AND STANDARDS .....</b>	<b>9</b>
<b>1.</b>	<b>RTU Design and Programming Standards .....</b>	<b>10</b>
<b>2.</b>	<b>Communications Protocols .....</b>	<b>10</b>
<b>a.</b>	<b><i>IEC 60870.....</i></b>	<b>10</b>
<b>b.</b>	<b><i>DNP3 .....</i></b>	<b>11</b>
<b>c.</b>	<b><i>HDLC .....</i></b>	<b>12</b>
<b>d.</b>	<b><i>Modbus .....</i></b>	<b>12</b>
<b>e.</b>	<b><i>Profibus .....</i></b>	<b>12</b>
<b>f.</b>	<b><i>Foundation Fieldbus.....</i></b>	<b>12</b>
<b>g.</b>	<b><i>UCA .....</i></b>	<b>12</b>
<b>III.</b>	<b>ARCHITECTURAL ANALYSIS.....</b>	<b>15</b>
<b>A.</b>	<b>TYPICAL THREE-LEVEL SCADA SYSTEM .....</b>	<b>15</b>
<b>1.</b>	<b>Introduction.....</b>	<b>15</b>
<b>2.</b>	<b>Business Systems .....</b>	<b>17</b>
<b>3.</b>	<b>Process Regulation .....</b>	<b>18</b>
<b>B.</b>	<b>REFINEMENT OF THE THREE LAYER MODEL .....</b>	<b>19</b>
<b>1.</b>	<b>Supervisory Control.....</b>	<b>20</b>
<b>2.</b>	<b>Process Control .....</b>	<b>21</b>
<b>3.</b>	<b>Field Instrumentation Control .....</b>	<b>21</b>
<b>4.</b>	<b>Boundaries .....</b>	<b>22</b>
<b>C.</b>	<b>OPERATIONAL AND MANAGEMENT FUNCTIONS OF THE THREE LAYERS .....</b>	<b>23</b>
<b>D.</b>	<b>OPERATIONAL AND MANAGEMENT FUNCTION REFINEMENT .....</b>	<b>24</b>
<b>1.</b>	<b>Mission .....</b>	<b>25</b>
<b>2.</b>	<b>Application Criticality .....</b>	<b>27</b>
<b>3.</b>	<b>Data Sensitivity.....</b>	<b>28</b>
<b>4.</b>	<b>Operating Environment .....</b>	<b>30</b>
<b>5.</b>	<b>System Interfaces .....</b>	<b>32</b>

6.	Communications Requirements .....	34
7.	Hardware and Software (Operational Functions).....	36
8.	Users and Personnel Actions (Management Functions).....	36
9.	Conclusion .....	37
E.	COMMUNICATIONS .....	38
1.	Communications Among Supervisory Control Peers.....	39
2.	Communications Between Supervisory Control and Process Control and Among Process Control Peers.....	39
3.	Communications between Process Control and Field Instrumentation Control .....	39
IV.	LABORATORY CONFIGURATION .....	41
A.	HARDWARE CONFIGURATION .....	41
1.	SLC-5/05 .....	41
a.	Processor .....	41
b.	Chassis & Power Supply.....	42
c.	Analog Input/Output Modules.....	43
d.	Digital Input/Output Modules .....	44
2.	Programming Console .....	44
a.	Hardware Configuration Baseline .....	44
b.	Software Configuration Baseline .....	44
3.	Operator Console .....	45
a.	Hardware Configuration Baseline .....	45
b.	Software Configuration Baseline .....	45
4.	Laboratory Configuration.....	45
a.	Simple Electrical Lab.....	45
b.	Simple Mechanical Lab .....	47
4.	Network and Communications .....	50
B.	SOFTWARE CONFIGURATION.....	51
1.	Programming Console .....	51
a.	RSLinx (Communications).....	51
b.	RSLogix500 (Programming IDE) .....	53
c.	RSView32 Works (Operator Interface IDE).....	54
2.	Operator Console .....	55
a.	RSLinx (Communications).....	55
b.	RSView32 Runtime (Operator Interface).....	56
3.	SLC-5/05 .....	56
a.	Simple Electrical Lab.....	56
b.	Simple Mechanical Lab .....	58
C.	APPLICABILITY OF ARCHITECTURAL ANALYSIS TO LABORATORY .....	61
1.	Simple Electrical Laboratory.....	61
a.	Boundary.....	61
b.	Operational and Management Functions Analysis.....	61
2.	Simple Mechanical Laboratory .....	64
a.	Boundary.....	64

<i>b.</i>	<i>Operational and Management Functions Analysis.....</i>	<i>65</i>
<b>V.</b>	<b>SUMMARY AND RECOMMENDATIONS.....</b>	<b>69</b>
<b>A.</b>	<b>SUMMARY .....</b>	<b>69</b>
<b>B.</b>	<b>RECOMMENDATIONS.....</b>	<b>70</b>
	<b>LIST OF REFERENCES.....</b>	<b>73</b>
	<b>APPENDIX A - SCADA PROTOCOLS.....</b>	<b>77</b>
<b>1.</b>	<b>IEC 60870 .....</b>	<b>77</b>
<b>2.</b>	<b>DNP3 .....</b>	<b>79</b>
<b>3.</b>	<b>HDLC .....</b>	<b>80</b>
<b>4.</b>	<b>MODBUS .....</b>	<b>81</b>
<b>5.</b>	<b>PROFIBUS .....</b>	<b>82</b>
<b>6.</b>	<b>FOUNDATION FIELDBUS .....</b>	<b>83</b>
<b>7.</b>	<b>UCA.....</b>	<b>84</b>
	<b>APPENDIX B - LADDER LOGIC FOR SCADA TECHNOLOGY TESTING</b>	
	<b>LABORATORY DEMONSTRATION MODEL .....</b>	<b>87</b>
<b>1.</b>	<b>LADDER LOGIC - PROGRAM ENTRY .....</b>	<b>87</b>
<b>2.</b>	<b>LADDER LOGIC - SIMPLE ELECTRICAL LAB.....</b>	<b>88</b>
<b>3.</b>	<b>LADDER LOGIC - SIMPLE MECHANICAL LAB .....</b>	<b>90</b>
	<b>APPENDIX C - ROBOTIC ARM MOVEMENT COMMANDS .....</b>	<b>95</b>
<b>1.</b>	<b>FROM: A TO: C .....</b>	<b>95</b>
<b>2.</b>	<b>FROM: B TO: A .....</b>	<b>96</b>
<b>3.</b>	<b>FROM: C TO: B .....</b>	<b>97</b>
	<b>INITIAL DISTRIBUTION LIST .....</b>	<b>99</b>

THIS PAGE INTENTIONALLY LEFT BLANK

## LIST OF FIGURES

Figure 1	Typical Control System. ....	3
Figure 2	Typical RTU Configuration (After Ref. [2]). ....	5
Figure 3	Three-levels of a typical SCADA system. ....	16
Figure 4	The three-layer model with the addition of business systems above and process regulation below. ....	17
Figure 5	Business Systems define the overall policy and rules that are implemented in a SCADA system, and receive feedback for auditing and accounting purposes. ....	18
Figure 6	Process Regulation describes the effect the SCADA system's components have on a physical object. (After Ref. [20]). ....	19
Figure 7	The SCADA system's three levels separated into a functional representation. (After Ref. [21]) ....	20
Figure 8	A boundary will normally be defined for components of a SCADA system with similar functional or geographic characteristics. ....	22
Figure 9	A complex SCADA system can define multiple boundaries among many layers, making evaluation difficult. ....	23
Figure 10	Each of the three layers has unique operational functions, and each is managed differently. ....	24
Figure 11	A SCADA system and its components must be evaluated in terms of nine considerations, for both operational functions and managerial functions. ....	37
Figure 12	Typical SCADA systems will communicate through a variety of methods and protocols between components and layers. ....	38
Figure 13	Physical Overview of Simple Electrical Lab. ....	46
Figure 14	Electrical Overview of Simple Electrical Lab. ....	47
Figure 15	Physical Overview of Simple Mechanical Lab. ....	49
Figure 16	Photograph of Simple Mechanical Lab. ....	49
Figure 17	Electrical Overview of Simple Mechanical Lab. ....	50
Figure 18	Network Topology. ....	51
Figure 19	The RSLinx Software package from Rockwell Software provides an interface between the client application and the underlying process control protocol. ....	52
Figure 20	This screenshot from RSLinx illustrates the main screen, showing the Ethernet driver configured to communicate with a SLC-5/05 programmable controller. ....	53
Figure 21	A screenshot of the RSView32 IDE main screen, annotated to show the different SLC-5/05 control functions available to a programmer. ....	54
Figure 22	A screenshot of a RSView32 Works sample application showing an operator interface to a process monitoring system. [19]. ....	55
Figure 23	The main operator interface for the Simple Electrical Lab, showing the manual override alarm engaged. ....	57
Figure 24	The logical flow of the Simple Electrical Lab Ladder Logic program. ....	58

Figure 25	The main operator interface for the Simple Mechanical Lab, showing the current position of the simulated barrel at Position A. ....	59
Figure 26	The logical flow of the Simple Mechanical Lab Ladder Logic program. ....	60
Figure 27	The boundary for the Simple Electrical Lab. ....	61
Figure 28	The boundary for the Simple Mechanical Lab. ....	65
Figure 29	Message Structure under IEC 60870-5-101 (bit serial links). (After Ref. [25]).....	78
Figure 30	A simplified breakdown of the DNP3 message format, showing encapsulation and assembly across layers. ....	80
Figure 31	High Level Data Link Control Frame Structure. The Control field is used to determine which class of message is being used. ....	81
Figure 32	The Modbus frame format. ....	82
Figure 33	The Foundation Fieldbus message format as it travels up the stack from the physical to the application layer. (After Ref [30]).....	84

## LIST OF TABLES

Table 1	IEC Standard 61131 Description [11][12].....	10
Table 2	IEC Standard 60870 [13] .....	11
Table 3	The nine concepts that must be examined in order to define the operational and management functions of a SCADA system.....	25
Table 4	Sample Mission analysis for the Electrical Demonstration Laboratory .....	26
Table 5	Sample Application Criticality analysis for the Electrical Demonstration Laboratory.....	28
Table 6	Sample Data Sensitivity analysis for the Electrical Demonstration Laboratory.....	30
Table 7	Sample Operating Environment analysis for the Electrical Demonstration Laboratory.....	31
Table 8	Sample System Interfaces analysis for the Electrical Demonstration Laboratory.....	33
Table 9	Sample Communications Requirements analysis for the Electrical Demonstration Laboratory .....	35
Table 10	Features of the SLC-5/05 Programmable Controller [22] .....	42
Table 11	1747-P1 Power Supply Characteristics [22].....	43
Table 12	The Operational Functions analysis for the Simple Electrical Lab .....	62
Table 13	The Management Functions analysis for the Simple Electrical Lab .....	63
Table 14	The Operational Functions analysis for the Simple Mechanical Lab .....	66
Table 15	The Management Functions analysis for the Simple Mechanical Lab .....	67

THIS PAGE INTENTIONALLY LEFT BLANK

## GLOSSARY

802.11a/b/g	A series of Institute of Electrical & Electronics Engineers (IEEE) Wireless LAN protocols.
AC	Alternating Current
Actuators	Field Instrumentation devices used to cause a change in a process.
Analog	A description of data represented by continuously variable, measurable, physical quantities, such as length, width, voltage, or pressure.
Analog/Digital Converter	A hardware device that transforms binary data to an analog representation.
ASDU	Application Service Data Unit. A message, following a specified format, that originates from the application and is passed to lower levels of the communications stack. Refers to the IEC 60870 protocol.
Bit	A discrete unit of measure having one of two possible values, described as 0 or 1.
Boundary	A logical segregation of related components in a system. The segregation may be based on physical location or function.
Bus	An electrical circuit that connects major components of an electronic device, allowing the transfer of electric impulses from one connected component to any other.
Byte	A set of eight bits.
Control System	The generic term for the hardware, software, and procedures used to control and monitor manufacturing and industrial processes, and to manage accumulated data for later study.
Current	The amount of electric charge flowing past a specified circuit point per unit time.
DC	Direct Current
DCS	Distributed Control System. A term used to describe a subset of control systems.
Digital	A description of data represented as a sequence of discrete symbols from a finite set, such as "on/off".
Digital/Analog Converter	A hardware device that transforms analog data to a binary representation.
DNP3	Distributed Network Protocol Version 3.3. Standard describing communications in a control system
Ethernet	A networking technology for local area computer networks.
Ethernet Port	A hardware interface that implements Ethernet networking.
Field Instrumentation Device	A hardware or combination hardware and software device designed to interact directly with a process. Examples include flow meters, valves, and switches.
Foundation Fieldbus	Standard describing communications in a control system
HDLC	High Level Data Link Control. Standard describing communications in a control system
IEC 60870	International Electrotechnical Commission standard describing communications in a control system
Intelligent Sensor/Intelligent Actuator	Sensors or actuators that contain embedded processing functionality to perform complex sensing and actuating tasks.
Interface	A point at which independent systems interact or exchange information.
IPX	Internetwork Packet Exchange, a networking protocol used by the Novell NetWare operating systems.

Ladder-logic	A programming language typically used in Programmable Logic Controllers. It derives its name from the ladder-like appearance of a program. The "rungs" of the program contain program instructions.
LED	Light Emitting Diode
mA	Milliampere. A unit for measuring current.
Master Station	A generic term used to describe an operator console or a programming console in a typical SCADA system.
MIS	Management Information System. A computer system designed to help managers plan and direct business and organizational operations.
Modbus	Standard describing communications in a control system
Multiplexer	A hardware component capable of interleaving two or more different signals.
Non-Volatile Memory	Computer storage that is not lost when the power is turned off
OSI 7-Layer Model	Open System Interconnection model that defines a networking framework for implementing protocols in seven layers. Control is passed from one layer to the next.
Peer to Peer	Communications between two or more components that have equal status.
PLC	Programmable Logic Controller. A Remote Terminal Unit that employs ladder logic programming.
Point to Point	Communications between two components directly connected to each other.
Process	A series of actions, changes, or functions bringing about a result.
Profibus	Standard describing communications in a control system
RF	Radio Frequency.
RS-232	Abbreviation for Electronics Industry Association (EIA) Recommended Standard 232. Defines a serial port.
RS-442	Abbreviation for Electronics Industry Association (EIA) Recommended Standard 442.
RS-485	Abbreviation for Electronics Industry Association (EIA) Recommended Standard 485
RTU	Remote Terminal Unit. A standalone data acquisition and control unit that monitors and controls actuators and sensors at a remote location.
SCADA	Supervisory Control and Data Acquisition. A subset of control systems, commonly referring to applications responsible for distribution of a commodity such as electricity or natural gas.
Sensors	Field Instrumentation devices used to detect a change in a process.
Serial Port	A hardware interface that allows for transmission of data one bit at a time.
TCP/IP	Transport Control Protocol/Internet Protocol. A suite of protocols for network communications.
UCA	Utility Communications Architecture standard describing communications in a control system
Volatile Memory	Computer storage that is lost when the power is turned off
Voltage	Electromotive force or potential difference, usually expressed in volts.
X.25	An International Telecommunications Union (ITU) standard describing communications in a packet switched network.

## **ACKNOWLEDGMENTS**

The author received considerable assistance and support from the staff of The Instrumentation, Systems, and Automation Society (ISA), particularly Mr. Charley Robinson, Mr. Mathew Franz, and the members of ISA-SP99 (Manufacturing and Control Systems Security) Committee. Charley, Matt and ISA-SP99 provided access to draft standards and technical reports and extended an invitation to monitor the standards creation process first hand. It was an informative and educational experience, and the opportunity to observe and interact with seasoned control systems professionals at work was invaluable to this research effort.

The guidance, support and patience exhibited by Dr. Cynthia Irvine, Ms. Deborah Shifflett and Professor Dick Harkins ensured that this thesis remained an enjoyable and interesting endeavor, and hopefully one that will benefit the Department of the Navy.

Finally, this thesis would not have been possible without the financial backing and support of the Department of the Navy Chief Information Officer.

THIS PAGE INTENTIONALLY LEFT BLANK

## I. INTRODUCTION

While Supervisory Control and Data Acquisition (SCADA) systems have been employed to monitor and control industrial facilities for decades, the designs of these systems, their components, and the communications protocols are primarily proprietary. There has been a trend of late to define standard interfaces and communications protocols, primarily driven by the growth of the Internet and consolidation of utility companies and industries. These efforts are a means of providing cross-vendor compatibility and modularity: since replacing an existing SCADA network is an expensive proposition, integrating existing systems is the most economical approach.

Further, communications between nodes in a SCADA system has been, until recently, over closed networks. With the advent of the Internet, many SCADA monitoring and control networks have been connected, at some level, to open networks, thereby inheriting all the problems and concerns associated with nodes on the Internet.

Because of these two trends, there are concerns about the security and stability of SCADA systems, especially since the September 11, 2001 attacks. Recent failures of critical infrastructure SCADA systems, such as the North East blackout in August 2003, highlight these concerns. Most of our nation's infrastructure is controlled in one way or another by a SCADA system, and the Department of Defense (DoD) relies heavily on the existing commercial infrastructure for its operations. To ensure continued operations in times of crisis, the SCADA systems on which the DoD depends must be secured.

An abstract generic framework for defining and understanding SCADA systems is needed as a first step toward securing them.

SCADA systems are not designed with security in mind; rather the priority for developers has been reliability, availability, and speed. This does not mean they cannot be secured, however. If we can understand a particular system's features, functions and capabilities, we can address its limitations. A generic abstract framework provides a tool to understand the system's features, functions and capabilities, and how components in the system relate and interface with each other. With that information about the system, we can begin the process of securing it.

This thesis begins with an examination of control systems, and SCADA systems in particular. It describes the different components in a SCADA system and the variety of open communications protocols that have been defined. The thesis then refines a three-tiered model and ultimately provides a matrix approach to describing and defining the features, functions and capabilities of a SCADA system.

Several examples illustrating how to apply the matrix approach to describing and defining SCADA systems are provided, using the Naval Postgraduate School's SCADA Technology Testing Lab Demonstration Model (NPS SCADA Lab). The NPS SCADA Lab was developed as part of this thesis using industry standard hardware from Allen-Bradley and software from Rockwell Automation, and provides several working systems to simulate real-world applications of SCADA technology. The NPS SCADA Lab configuration is described in the thesis.

## II. BACKGROUND

### A. OVERVIEW OF CONTROL SYSTEMS

The use of automation in manufacturing and industrial processes presupposes a mechanism for the operator to control and monitor physical functions in real time. As complexity of these processes increases, the ability for remote control and monitoring from a central location provides increased labor and cost efficiency and offers opportunities to increase the economies of scale. Further, aggregation of feedback data provides supervisors and management personnel the ability to monitor trends, forecast requirements, and optimize procedures. A *Control System* is the generic term for the hardware, software, and procedures used to control and monitor these processes, and to manage the accumulated data for later study.

A typical control system consists of one or more remote terminal units (RTU) connected to a variety of sensors and actuators, and relaying information to a master station. Figure 1 illustrates this generic three tiered-approach to control system design.

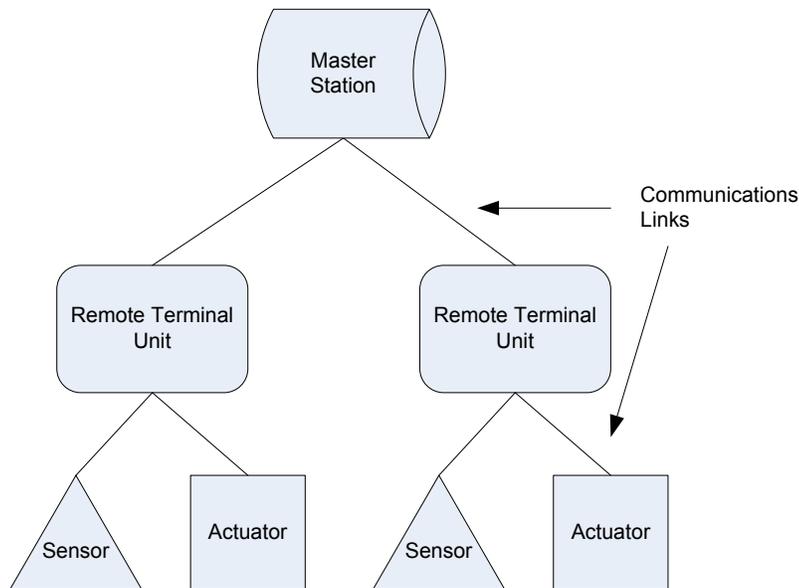


Figure 1 Typical Control System.

The design and function of the RTUs, sensors, actuators, and master station, as well as the means of communication between components, are implementation details

that will vary depending on the manufacturing or industrial process being controlled. A distributed control system may have multiple master stations or layers of master stations.

### **1. Sensors and Actuators**

The philosophy behind control systems can be summed up by the phrase "If you can measure it, you can control it." [1] Sensors perform measurement, and actuators perform control.

Sensors measure level, pressure, flow, current, voltage, temperature, a binary status ("on" or "off"), or react to some other external stimulus. The acquired data can be either analog (continuously variable values, usually proportional to the measured quantity) or digital (sequence of discrete values from a finite set). The results of the measurements are transmitted via a communications link to the RTU in either a raw form or manipulated by a processor found within the sensor itself before transmission to the RTU. The communications link itself may be analog or digital.

Actuators open or close valves, regulate pumps, open or close relays, trip circuit breakers, or perform other mechanical functions. The command passed to an actuator can be either analog or digital, and the communications link may be either analog or digital.

### **2. Remote Terminal Units**

A Remote Terminal Unit (RTU) is a standalone unit used to monitor and control sensors and actuators at a remote location, and to transmit data and control signals to a central master monitoring station. Depending on the sophistication of the microcontroller in the RTU, it can be configured to act as a relay station for other RTUs which cannot communicate directly with a master station, or the microcontroller can communicate on a peer-to-peer basis with other RTUs. RTUs are generally remotely programmable, although many can also be programmed directly from a panel on the RTU.

Small size RTUs generally have less than 20 analog or digital inputs and medium size RTUs typically have 100 digital and up to 40 analog inputs, while an RTU with greater than 100 digital or 40 analog inputs is considered large. Many RTUs are modular and thus expandable, and several RTUs can be logically combined as one, depending on the model and manufacturer. [2]

Figure 2 shows a typical RTU. A RTU consists of a power supply, a central processing unit (CPU), memory (both volatile and non-volatile), and a series of inputs and outputs. The CPU controls communications with the sensors and actuators through the inputs and outputs, and with the master station through a serial port, an Ethernet port, or some other interface. A programming interface can also be connected to any of these interfaces. The Central Bus serves as the conduit for communications between the components of the RTU.

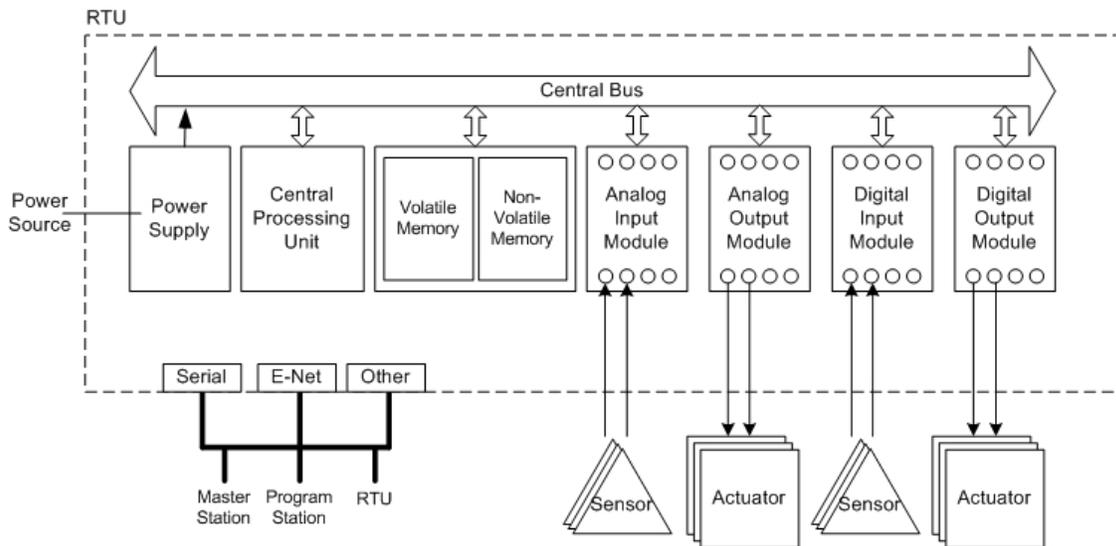


Figure 2 Typical RTU Configuration (After Ref. [2]).

**a. Programmable Logic Controllers**

Advances in CPUs and the programming capabilities of RTUs have allowed for more sophisticated monitoring and control. Applications that had previously been programmed at the central master station can now be programmed at the RTU. These modern RTUs typically use a ladder-logic approach to programming due to its similarity to standard electrical circuits--the majority of RTU programmers are engineers, not computer programmers. A RTU that employs this ladder logic programming is called a Programmable Logic Controller (PLC). PLCs are quickly becoming the standard in control systems.

Ladder-logic development environments mimic electrical circuits by drawing two vertical lines representing power, with the horizontal "rungs" representing the logic required to "close" a circuit and thus send a signal to an actuator. Each "rung" is a step in a sequential program. [2] (Appendix B contains a detailed explanation of the ladder logic programming used in the SLC-505 controller for the demonstration laboratory.)

***b. Analog Input and Output Modules***

The configuration of sensors and actuators determines the quantity and type of inputs and outputs on a PLC or RTU; depending on the model and manufacturer, modules can be designed solely for input, output, digital, analog, or any combination.

An analog input module has a number of interfaces, usually binding posts or screw posts, which are wired directly to a number of sensors. A multiplexer in the module samples each of the analog interfaces in turn and passes the reading to an Analog/Digital (A/D) converter to convert the analog signals to digital representations, usually 8 or 12 bits, for transmission to the CPU over the central bus. Typical analog input modules have 8, 16, or 32 inputs.

Analog output modules work in reverse: they take digital values from the CPU and convert them to analog representations, which are then sent to the actuators. An output module usually has 8, 16 or 32 output binding or screw posts, and typically offers 8 or 12 bits of resolution. [2]

***c. Digital Input and Output Modules***

Digital input modules typically are used to indicate status and alarm signals. A number of binding or screw posts (usually 8, 16 or 32) receive a signal from the sensor to indicate either an "open" or "closed" circuit, and can usually be configured to read a variety of voltages or currents. Depending on the manufacturer, modules also have an LED to indicate the current value of the signal.

A specialized digital input module is used for counting pulses of voltage or current, rather than for strictly indicating "open" or "closed." This functionality, however, can also be implemented using standard input modules and functions found in the ladder-logic programming language of the PLC.

Digital output modules drive a voltage to a binding or screw post, and typically have an LED to indicate the current value of the signal. [2]

*d. Communications interfaces*

Modern RTUs and PLCs offer a wide variety of communications means, either built in directly or through a module. The following list represents a variety of transmission methods supported:

- RS-232/RS-442/RS-485
- Dialup telephone lines
- Dedicated telephone lines
- Microwave
- Satellite
- X.25
- Ethernet
- 802.11a/b/g
- Radio (VHF, UHF, etc) [2]

Each of these methods could be used to communicate with the master station, other PLCs or RTUs, the programming station, or operator consoles. Chapter II, Section D discusses the variety of communications protocols available and the media for each.

**3. Master Stations**

Master stations have two main functions:

- Periodically obtain data from RTUs/PLCs (and other master or sub-master stations)
- Control remote devices through the operator station [3]

Master stations consist of one or more personal computers (PC), which, although they can function in a multi-purpose mode (email, word processing, etc), are configured to be dedicated to master station duties. These duties include trending, alarm handling, logging and archiving, report generation, and facilitation of automation. These duties may be distributed across multiple PCs, either standalone or networked.

A master station may communicate to one RTU via a serial, Ethernet, wireless, radio, or other means. It could also communicate to a number of RTUs which are networked together, or it could communicate with one RTU that is in a peer-to-peer relationship with other RTUs. A master station may also aggregate data from any number of sub-master stations. The design of the master stations is situationally dependant.

## **B. SUPERVISORY CONTROL AND DATA ACQUISITION**

Control systems are used at all levels of manufacturing and industrial processing. A manufacturing plant that employs robotic arms will have a control system to direct robotic arms and conveyor belts on the shop floor. It may use that same system for packaging the finished product and tracking inventory. It may also use a control system to monitor its distribution network. A chemical company will use control systems to monitor tank levels and to ensure that ingredients are mixed in the proper proportions. A Las Vegas casino will use control systems to direct the spray from water fountains in coordination with the lights and music. Control systems are also used in the drilling and refining of oil and natural gas. [9] They are used in the distribution of water and electricity by utility companies, and in the collection of waste water and sewage. [10] Virtually every sector of the economy employs control systems at all levels.

Because of the ubiquitous nature of control systems, a variety of terms have originated to describe them: process control systems, distributed control systems, automation control systems, industrial control systems, and supervisory control and data acquisition systems. All the terms refer in the broadest sense to control systems as defined above. Some focus on manufacturing and the actions that take place on the factory floor, and each industry sector will define its own terms. [2, 4, 5, 6]

The term "supervisory control and data acquisition" (SCADA), however, is generally accepted to mean the systems that control the distribution of critical infrastructure public utilities (water, sewer, electricity, and oil and gas). [7] <sup>1</sup> Confusion

---

<sup>1</sup> ANSI C37.1 defines "Supervisory Systems" as "all control, indicating, and associated telemetry equipments at the master station, and all of the complementary devices at the remote station, or stations." This definition further confuses the issue, because it appears to define control systems in general, and not SCADA systems. [8]

arises, however, because while control systems are actively used in other critical infrastructure sectors (transportation, chemicals), the term SCADA is not used to describe them. Other critical systems, such as those aboard Navy ships, are referred to as SCADA systems even though they do not meet the generally accepted definition of the term.

The General Accounting Office has reduced this confusion by using the generic term "Control Systems" and describing two types of control systems: Distributed Control Systems (DCS) and SCADA:

[DCS] typically are used within a single processing or generating plant or over a small geographic area. [SCADA] systems typically are used for large, geographically dispersed distribution operations. A utility company may use DCS to generate power and a SCADA system to distribute it. [4]

This thesis will adapt the GAO approach, with one modification: the term SCADA will not be limited to describing "large, geographically dispersed" systems, but rather will include any distribution system, whether "large" or not; "geographically dispersed" or not. *The key word in the definition is "distribution."*

This modification will allow the inclusion of Navy ship-to-shore utility control systems in our definition. Shipboard control systems are more properly defined as DCS unless they are involved in managing the distribution of some commodity. Nevertheless, the models presented in Section III make little or no distinction between distribution systems and non-distribution systems.

### **C. PROTOCOLS AND STANDARDS**

In SCADA Systems, the three major categories of protocols involve the specifications for design and manufacture of sensors and actuators, specifications for RTUs, and the specifications for communications between components of a control system.

The specifications for design and manufacture of sensors and actuators are concerned with the engineering requirements for specific industrial components such as valves and measurement equipment, and also dictate safety tolerances, measurement thresholds, and environmental considerations. They are typically issued by the International Standards Organization (ISO) or the International Electrotechnical

Commission (IEC). A thorough examination of these standards is beyond the scope of this thesis.

## 1. RTU Design and Programming Standards

The prevalent standard for industrial control RTU design and programming is the IEC 61131 series, developed by the two IEC working groups, the *Industrial Process Measurement And Control* group and the *IT Applications In Industry* group. It is a series of seven publications that serve to standardize the programming languages, instruction sets, and concepts used in industrial control devices such as RTUs and PLCs. Table 1 briefly describes the volumes in the standard.

Table 1 IEC Standard 61131 Description [11][12]

Standard	Description
IEC 61131-1	General Information
IEC 61131-2	Specifies requirements and related tests for PLCs and associated peripherals. Establishes definitions and identifies principal characteristics. Specifies the minimum requirements for functional, electrical, mechanical, environmental and construction characteristics, service conditions, safety, Electromagnetic Compatibility (EMC), user programming and testing.
IEC 61131-3	Specifies syntax and semantics of programming languages for programmable controllers
IEC 61131-4	Technical Report. Provides guidelines addressing the application PLCs and their integration into automated systems.
IEC 61131-5	Specifies communications aspects of a PLC. Specifies behavior of the PLC as it provides services on behalf of other devices and the services the PLC application program can request from other devices. Specified independent of the particular communication subsystem.
IEC 61131-6	Reserved for future use
IEC 61131-7	Specifies a means to integrate fuzzy control applications in the PLC languages as defined in Part 3.
IEC 61131-8	Technical report addressing the programming of PLCs using the PLC languages defined in Part 3

## 2. Communications Protocols

### a. IEC 60870

There are two major protocol descriptions for SCADA component communications, designed specifically for the purpose of process control applications. The first is IEC 60870, and is described in Table 2. IEC 60870 was defined primarily for the telecommunications of electrical system and control information and its data

structures are geared to that application. It is the favored standard in the United States for electrical power grid SCADA systems, but is not as popular in Europe.

Table 2 IEC Standard 60870 [13]

Standard	Description
IEC 60870-1	General Considerations
IEC 60870-2	Operating Conditions
IEC 60870-3	Interfaces - electrical characteristics
IEC 60870-4	Performance Requirements
IEC 60870-5	Transmission Protocols
IEC 60870-6	Telecontrol protocols compatible with ISO standards and ITU-T recommendations

Of particular importance is Part 5 which redefines the 7-layer OSI Reference Model (ISO 7498) to fit a SCADA environment. IEC 60870-5 and four companion standards define physical, link and application layers, as well as a "user process" above the application layer for non-networked (point-to-point) applications. It also defines a five layer (plus user process) model for networked applications, adding a network and transport layer. These standards are IEC 60870-5-101 and -104 respectively. [13]

Appendix A describes the implementation of the IEC 60870 standards in detail, including the structure of frames and messages.

***b. DNP3***

The second protocol specifically designed for SCADA communications is the Distributed Network protocol Version 3 (DNP3). Also created for the electrical industry, it has been adapted by other industry sectors and is the leading protocol employed in Europe for most SCADA applications. It enjoys a wide adoption in the United States and South America for all industries except the electrical power industry. It also provides a robust compliance certification framework, allowing hardware and software vendors to easily adapt their equipment.

DNP3 defines four layers, physical, data link, pseudo-transport, and application. It is less restrictive than IEC 60870 and thus allows for expandability beyond the electrical industry. [13] Details of the standard are found in Appendix A.

**c. HDLC**

Several other SCADA standards exist, primarily High Level Data Link Control (HDLC) and Modbus. HDLC, defined by ISO for point-to-point and multi-point links, is also known as Synchronous Data Link Control (SDLC) and Advanced Data Communication Control Procedure (ADCCP). It is a bit-based protocol, the precursor to Ethernet, and is rapidly being replaced by DNP3, Industrial Ethernet<sup>2</sup>, and TCP/IP. [14]

**d. Modbus**

Modbus is a relatively slow protocol that does not define interfaces, thus allowing users to choose between EIA-232, EIA-422, EIA-485 or 20mA current loop. While slow, it is widely accepted and has become a *de-facto* standard--a recent survey indicated that 40% of industrial communication applications use Modbus. [14]

Details of both HDLC and Modbus can be found in Appendix A.

**e. Profibus**

Profibus is a German standard that defines three types: Field Message Specification (FMS) for use in general data acquisition systems, Decentralized Peripherals (DP) for use when fast communication is required, and Process Automation (PA) for use when highly reliable and safe communication is required. It defines three layers: physical, data link and application. [15]

**f. Foundation Fieldbus**

Foundation Fieldbus is an extension to the 4-20mA standard to take advantage of digital technologies. It defines 3+1 layers (physical, data link, application, and user). [15]

**g. UCA**

The Utility Communications Architecture (UCA) is a new initiative from the Electric Power Research Institute (EPRI) designed for the electrical industry. It is more than just a protocol definition; it is a comprehensive set of standards designed to

---

<sup>2</sup> Industrial Ethernet is a term for proprietary implementations of TCP/IP or other protocols over standard IEEE 802.3 or proprietary Ethernet. One example is Ethernet/IP (Industrial Protocol) [17, 18].

allow "plug and play" integration into systems, allowing manufacturers to design off-the-shelf compliant devices. IEEE assumed the UCA standards process in 1999 and has developed extensions for the water industry. Other industries are also examining UCA for suitability. [16]

Details for Profibus, Fieldbus and UCA can be found in Appendix A.

THIS PAGE INTENTIONALLY LEFT BLANK

### **III. ARCHITECTURAL ANALYSIS**

Although many manufacturers and standards exist for Supervisory Control and Data Acquisition (SCADA) systems, some features are common to all systems. This chapter will review some of those previously described features and provide a framework for better understanding the characteristics of SCADA systems that can be applied to any system. Understanding a system is the first step to securing it from compromise.

#### **A. TYPICAL THREE-LEVEL SCADA SYSTEM**

##### **1. Introduction**

As noted in Chapter II, a control system aids in automating manufacturing and industrial processes, providing a mechanism for an operator to control and monitor physical functions (known as processes) either locally or remotely, and for supervisors and management to aggregate feedback data to monitor trends, forecast requirements, and optimize procedures.

A typical control system consists of one or more remote terminals connected to a variety of sensors and actuators, and relaying information to one or more master stations. A Remote Terminal Unit (RTU) is a standalone unit used to monitor and control sensors and actuators, and to transmit data and control signals to a central master monitoring station. Sensors and actuators are specialized hardware and software components that elicit information about the current status of or provide a means for influencing the process. The Master Station periodically obtains data from the RTU and provides an interface for control of remote devices.

Supervisory Control and Data Acquisition (SCADA) is the term commonly applied to control systems involved in the distribution of a commodity. Figure 3, reproduced from Chapter II, illustrates a generic three tiered-approach to SCADA control system design incorporating the three main components described above.

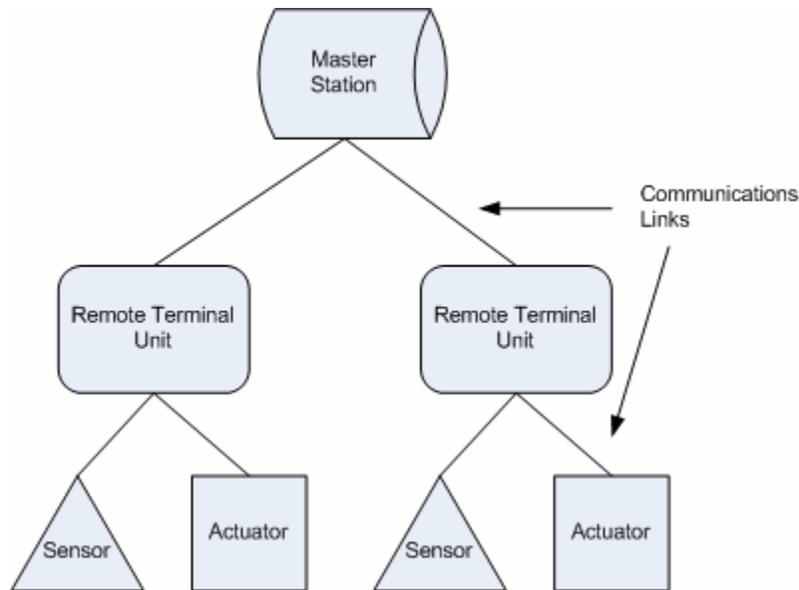


Figure 3 Three-levels of a typical SCADA system.

In addition to the Master Station, RTUs, and sensors and actuators, two additional influences can be added to the model as illustrated in Figure 4. In any organization, business systems dictate policy and procedures relevant to the control and monitoring of the process; conceptually, these reside above the Master Station. Similarly, the sensors and actuators directly act upon the physical objects we have been calling the "process." An in-depth examination of the Business System and Process Regulation layers is beyond the scope of this thesis, but they are briefly examined in subsequent sections.

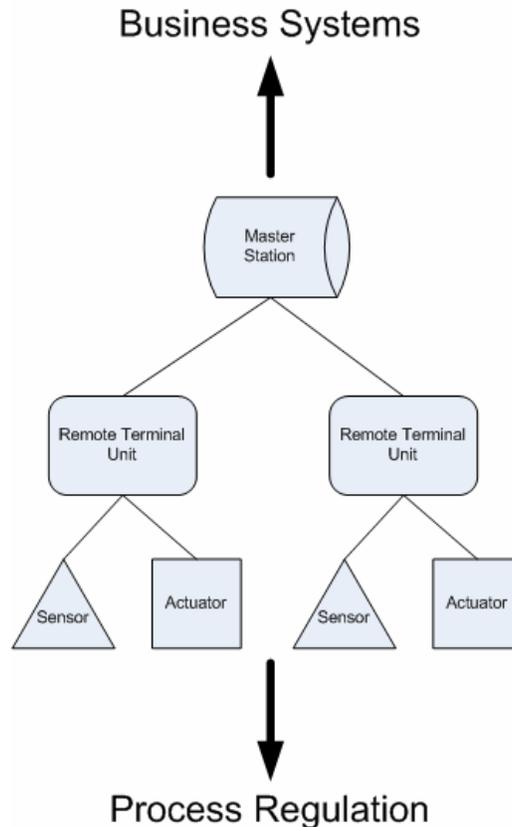


Figure 4 The three-layer model with the addition of business systems above and process regulation below.

## 2. Business Systems

A typical organization will generate policies and procedures that define the process that must be monitored and controlled, allocate resources to it, and dictate how collected data will be distributed and audited. A management information system (MIS) may facilitate access to the data supplied by the process, and can be used for forecasting, trending and optimization. Figure 5 illustrates some components of a Business System that will affect a SCADA implementation.

Policies and procedures at all layers within the Business System affect the design and operation of the SCADA system. For example, an enterprise level policy requiring access to the real-time status of a process for high-level decision making may affect the type of protection measures implemented at the Master Station, due to the integration of the control system communications network with the organization's business network. Similarly, a policy requiring compliance with health and safety regulations may require monitoring a process that does not directly impact operations and which would not

otherwise be monitored. In general, a thorough examination of all policies and procedures in an organization is required to understand or design a SCADA system; since these business systems vary widely from organization to organization, the SCADA system design is tailored to the specific organization's policies and procedures.

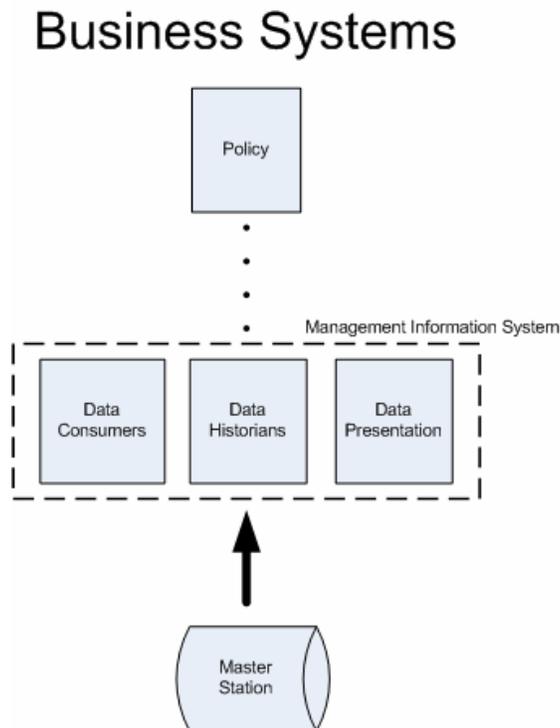


Figure 5 Business Systems define the overall policy and rules that are implemented in a SCADA system, and receive feedback for auditing and accounting purposes.

### 3. Process Regulation

Processes are a logical representation of the actions, changes, or functions bringing about a result on a physical object. Processes may have inputs and outputs, and be affected by external environmental disturbances. As shown in Figure 6, a SCADA system is designed to monitor and affect these processes via a control loop, and report back the consequences of the external stimulus on a process.

For example, one typical process would involve regulating the flow of JP5 fuel to a gas turbine engine. An operator (whether human or automated) determines the volume of fuel destined for the engine and instructs the system to open or close the valve as

required. The SCADA system's actuator, based on parameters transmitted to it by the RTU, directly controls the valve to open or close. The SCADA system's sensor will concurrently monitor the pressure in the fuel line, providing immediate feedback to the operator through the RTU. The operator has immediate awareness of the fuel line pressure, and can compensate the valve's settings to achieve the desired equilibrium. The sensor will also immediately perceive a rapid drop in line pressure, indicating perhaps a break in the fuel line. Since, in practice, there are innumerable processes, each tailored to the implementation, this thesis does not address processes themselves, except to describe the commonalities of "sensing" and "actuating."

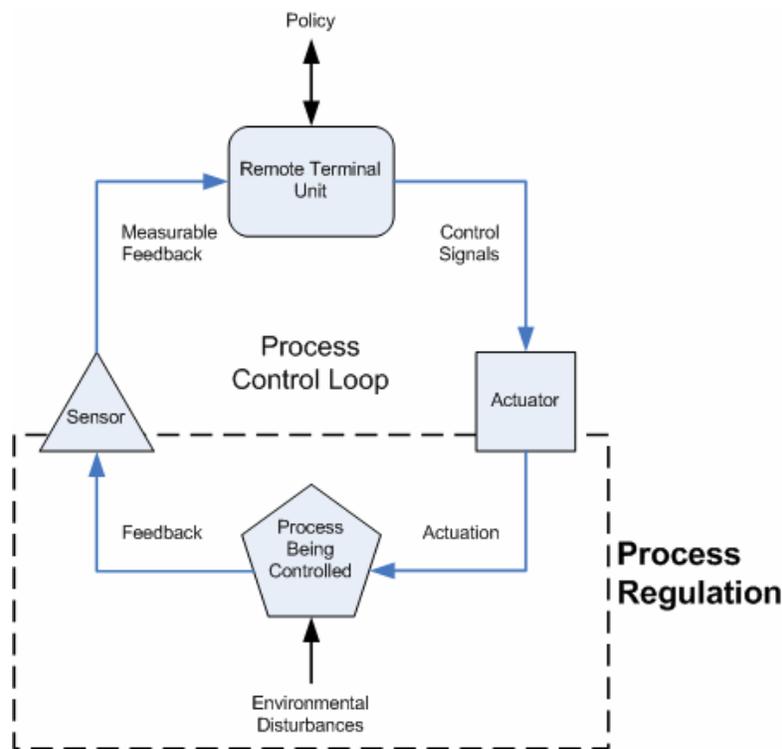


Figure 6 Process Regulation describes the effect the SCADA system's components have on a physical object. (After Ref. [20])

## B. REFINEMENT OF THE THREE LAYER MODEL

We can abstract the rather specific terms we have been using in our model so far by defining the various SCADA components by function. The Master Station exhibits properties of supervisory control. The Remote Terminal Units, responding to the supervisor, manipulate the process. The sensors and actuators, meanwhile, act on the

hardware components to carry out the specifics as directed by the process controller. A logical layering resulting from the functional breakdown, as shown in Figure 7, allows an abstraction that can be applied to systems of varying complexity and size and broadens our ability to examine requirements of each layer. The Supervisory Control layer, the Process Control layer, and the Field Instrumentation layer are each described below.

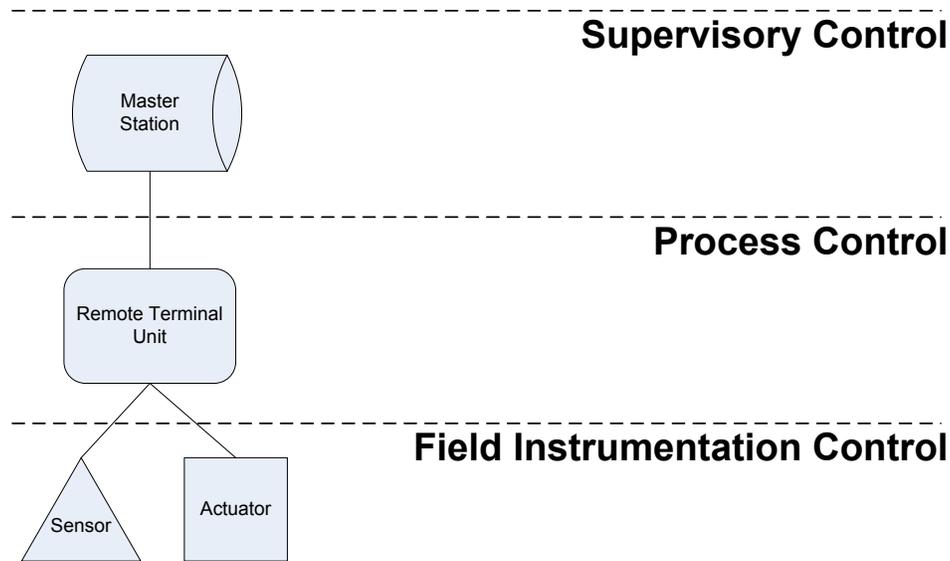


Figure 7 The SCADA system's three levels separated into a functional representation. (After Ref. [21])

## 1. Supervisory Control

The Supervisory Control layer is the *primary zone of control* for the SCADA system. Its primary functions are to:

- *Implement policy received from the Business System.* The Supervisory Control layer provides the interface for implementing business policy against the process, defining the actions the operator (whether human or automated) must take.
- *Manage the operational components of the SCADA system.* The Supervisory Control layer provides the interface for the configuration, management, programming and maintenance of the process control and field instrumentation components.
- *Provide access to the process data for archiving and analysis.* The Supervisory Control layer accepts process data (feedback) from the lower

levels and makes that data available to the Business System for its purposes.

Components on this level include the operator's console for day to day operations, and the programming console for programming, configuration, management and maintenance.

## **2. Process Control**

The Process Control layer is the *primary zone of operation* for the SCADA system. Its primary functions are to:

- *Receive directives from the Supervisory Control layer.* The Process Control layer applies logic rules dictated by policy to the directive, formats it for transmission to the field instrumentation units, and transmits it.
- *Receive feedback data from the Field Instrumentation layer.* The Process Control layer applies logic rules to the data, determines initial actions based on policy, formats data for transmission to the Supervisory Control layer, and transmits it.
- *Act on feedback data autonomously.* The Process Control layer receives feedback from Field Instrumentation units, applies pre-programmed policy-based logic rules to the data, determines actions required, and transmits directives back to Field Instrumentation units.
- *Manage day to day operations.* The Process Control layer monitors the health of Field Instrumentation units and reports anomalies to the Supervisory Control layer.

Components at this layer include RTUs, Programmable Logic Controllers (PLC), and intelligent sensors and actuators.

## **3. Field Instrumentation Control**

The Field Instrumentation layer is the *primary zone of implementation* for the SCADA system. Its functions are to:

- *Receive instructions from the Process Control.* The Field Instrumentation layer continuously monitors for Process Control layer directives and carries the directives out against the process.
- *Monitor the process for changes.* The Field Instrumentation layer continuously monitors the process components for changes and transmits those changes to the Process Control layer.

Components at this layer include sensors (such as valve flow meters or voltage meters), and actuators (such as valve flow regulators and voltage regulators). Intelligent

sensors and actuators combine capabilities of both Field Instrumentation Control and Process Control components.

#### 4. Boundaries

It is likely that different SCADA components will be segregated into logical functional or geographic boundaries that don't necessarily parallel security boundaries, as shown in Figure 8. The decision on defining boundaries is arbitrary and is largely dependant on the specific implementation.

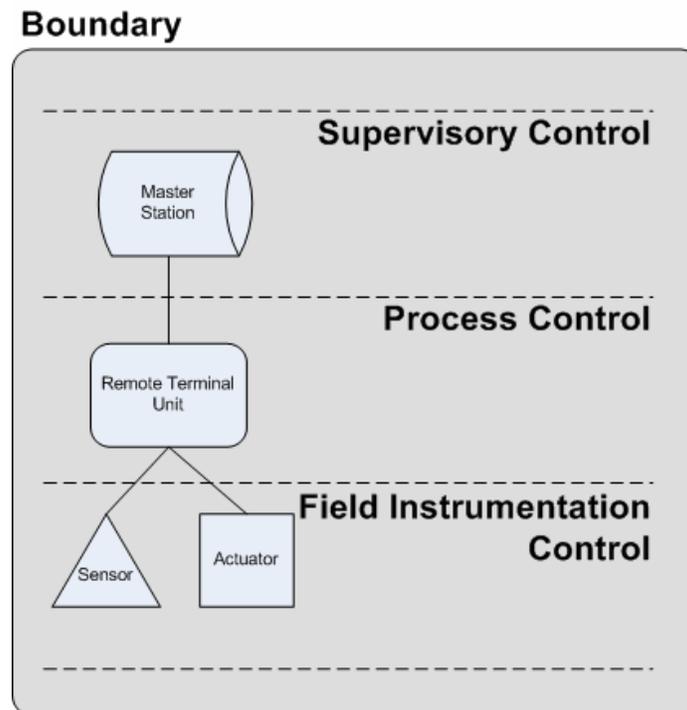


Figure 8 A boundary will normally be defined for components of a SCADA system with similar functional or geographic characteristics.

The problem of defining boundaries and synchronizing policies between boundaries is well known, and work continues on addressing this problem. The example in Figure 9 shows how a complex system consisting of a variety of boundaries can make assessment difficult. There are currently no effective methods for systematically defining or describing boundaries, nor for partitioning or assigning individual functions to specific boundaries. Until an effective method is described, explicit attention and care must be given to the effects establishing boundaries has on effecting security policy.

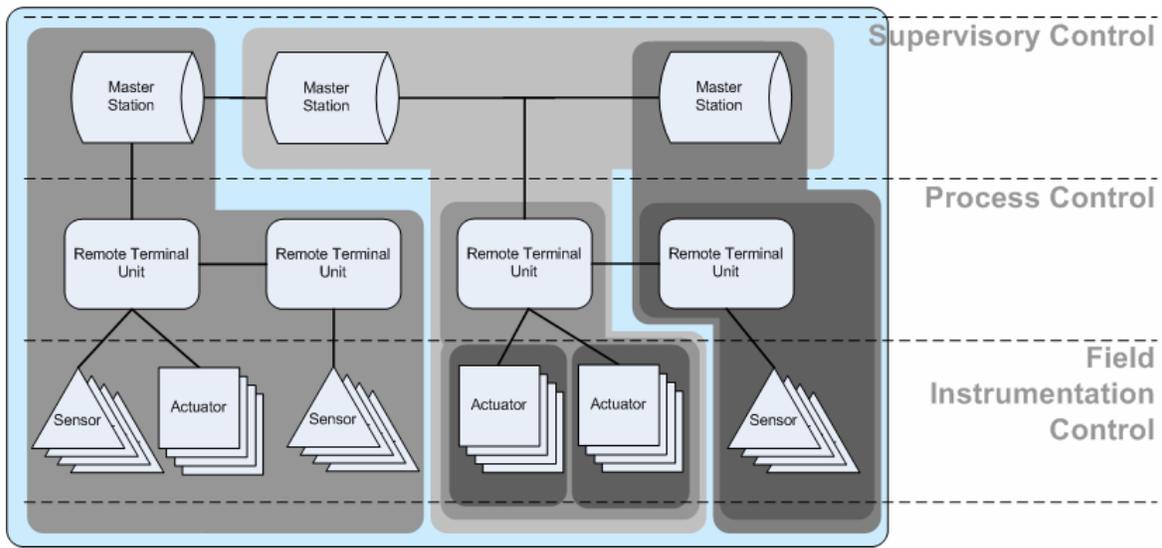


Figure 9 A complex SCADA system can define multiple boundaries among many layers, making evaluation difficult.

### C. OPERATIONAL AND MANAGEMENT FUNCTIONS OF THE THREE LAYERS

In the functional decomposition outlined above, each layer is assigned different responsibilities to perform; consequently, each layer is also managed differently. In evaluating SCADA systems for security, the distinction between management and operation plays a crucial role. SCADA systems by their nature, are designed for availability and data integrity, not confidentiality or authentication, and can reliably operate undisturbed for months or years with no human interaction or involvement. Due to the independent and heavily hardware-oriented nature of SCADA systems, security procedures used in traditional computing systems-such as forced password changes after a set period or routine account auditing by administrators-do not apply to SCADA components. An active management plan thus becomes a necessary requirement in developing a SCADA system. A separation of the operational aspects of a system from the management aspects ensures that both developers and policy makers address the unique nature of the systems. While the system may *operate* reliably on its own, it must still be *actively managed* as part of an ongoing plan to ensure correct and complete

implementation of security policy. Thoroughly examining both allows for a more comprehensive assessment of a system.

Figure 10 describes a very generic system in terms of operational and managerial functions. The operational functions and management functions can correlate with each other, but each can also independently identify requirements that need to be addressed. For example, a critical shipboard process may require that the RTU employ the shipboard local area network (LAN) to communicate with the Master Control station. In this scenario, the management functions will assess the security implications of exposing the SCADA traffic to the shipboard LAN. The operational functions will assess the impact on responsiveness and availability of a critical time sensitive system that shares network bandwidth with regular shipboard users.

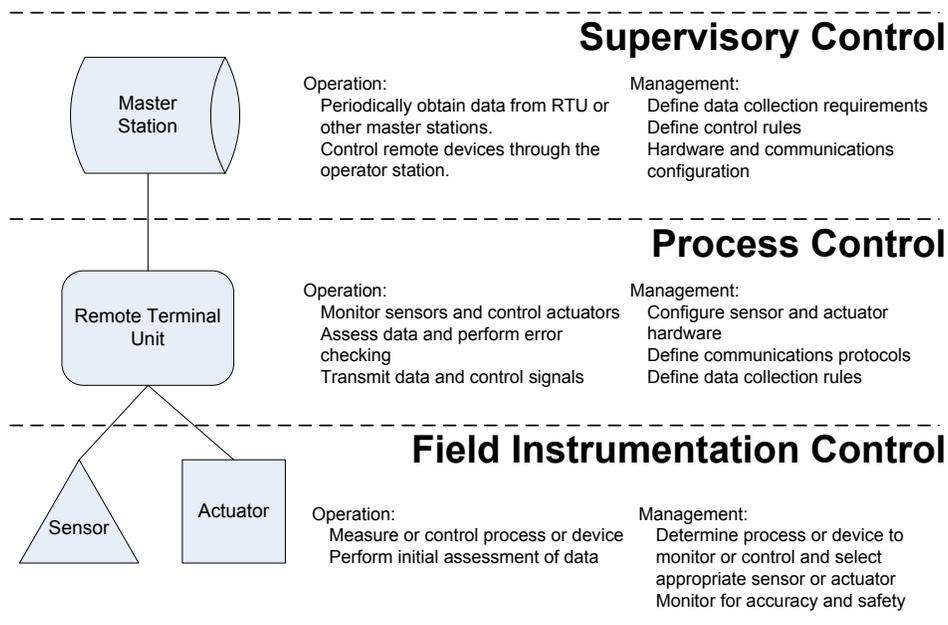


Figure 10 Each of the three layers has unique operational functions, and each is managed differently.

#### D. OPERATIONAL AND MANAGEMENT FUNCTION REFINEMENT

To better understand the variety of operational and management functions of a particular SCADA system, a refinement of the function descriptions is required. This thesis proposes nine concepts that should be evaluated: six are common to both

operational and management functions, two are relevant to operational functions only, and one is relevant to management functions only. Each of these nine functions must be examined for all components of a SCADA system, including the boundary. Table 3 summarizes the concepts, which are described in more detail in the sections that follow.

Table 3 The nine concepts that must be examined in order to define the operational and management functions of a SCADA system.

<b>Name</b>	<b>Operation Functions</b>	<b>Management Functions</b>
<b>Mission</b>	<b>X</b>	<b>X</b>
<b>Application Criticality</b>	<b>X</b>	<b>X</b>
<b>Data Sensitivity</b>	<b>X</b>	<b>X</b>
<b>Operating Environment</b>	<b>X</b>	<b>X</b>
<b>System Interfaces</b>	<b>X</b>	<b>X</b>
<b>Communications Requirements</b>	<b>X</b>	<b>X</b>
<b>Hardware</b>	<b>X</b>	
<b>Software</b>	<b>X</b>	
<b>Users and Personnel Action</b>		<b>X</b>

### 1. Mission

A clear understanding of the purpose of the component is required, in both operational and managerial terms. In addition, the boundary itself must have a clearly defined mission which will drive the design and the interaction between other boundaries and components. The individual components' missions may not be identical to the boundary's mission, but they will support it. Table 4 illustrates a sample Mission analysis for a SCADA system that monitors electrical voltage.

Table 4 Sample Mission analysis for the Electrical Demonstration Laboratory

<b>Mission - Electrical Demonstration Laboratory</b>			
		<b>Operation</b>	<b>Management</b>
<b>Boundary</b>		The Electrical Demonstration Laboratory will remotely monitor the voltage levels at a control panel, and ensure adjustments to that voltage are correctly made.	The Electrical Demonstration Laboratory will allow an operator to remotely control the operation of Process XYZ. A historical archive of past voltage readings will be maintained in the Management Information System (MIS).
<b>Component</b>	<b>Supervisory Control</b>	The operator console will provide an interface an operator to monitor the real-time voltage applied to the wires. It will allow the operator to adjust the voltage within a range, and will sound an alarm if a discrepancy exists between the actual voltage and the voltage reflected on the console, indicating that a manual override condition has occurred at the panel.	The operator console will provide an operator with an interface with which Process XYZ can be remotely controlled.
	<b>Process Control</b>	The RTU will receive voltage readings from the sensor monitoring the wires, ensure that voltage is within the acceptable range, and store the current reading for retrieval by the operator console. The RTU will receive a request from the operator console to change the voltage on the wire. After verifying that the voltage is within acceptable limits, it will direct the actuator to apply the requested voltage to the wire. If the RTU detects a discrepancy between the voltage last requested by the operator console and the voltage currently on the wire, it will raise an alarm.	The RTU will provide a means for the operator console to transmit requested commands (voltage levels) to Process XYZ in real-time, and allow for a means of returning feedback (voltage levels) from Process XYZ. It will also provide an indication if the events requested by the operator console are not reflected in Process XYZ.
	<b>Field Instrumentation Control</b>	The analog voltage sensor will monitor the voltage flowing through the wire and report that voltage to the RTU. The analog voltage actuator will receive a voltage command from the RTU and provide that voltage on the wire.	The instrumentation control components will provide real-time feedback on the status of Process XYZ to the RTU and allow for real-time adjustments to Process XYZ.

## **2. Application Criticality**

Since SCADA systems typically support vital distribution systems, an assessment of the criticality of that system will provide direction on its design and implementation, as well as the type of security measures to apply. The criticality must be examined both in terms of our system, and the individual components that make up that system. Several questions must be answered:

- Are the processes our system is monitoring of a nature that failure of the monitor functions will cause serious harm to the organization's mission?
- Are the processes our system is controlling of a nature that failure of the control functions will cause serious harm to the organization's mission?
- What type and level of failure of our system is acceptable?
- Are the individual components in our system critical to the overall success of the system?
- What type and level of failure of the components is acceptable?

This examination involves a risk assessment decision that is beyond the scope of this thesis, but a standard risk assessment methodology will likely work. The criticality of the boundary and the criticality of an individual component may not be the same, but they will support each other. Table 5 describes a sample Application Criticality examination for a SCADA system that monitors electrical voltage.

Table 5 Sample Application Criticality analysis for the Electrical Demonstration Laboratory

Application Criticality - Electrical Demonstration Laboratory			
		Operation	Management
<b>Boundary</b>		The Electrical Demonstration Laboratory provides remote status reports of voltage level readings, and allows remote control of these levels. Since there is a (local) control panel which allows the same functionality, this application is not deemed mission critical.	The Electrical Demonstration Laboratory provides remote control of process XYZ. Since there is an alternate means of locally controlling Process XYZ, this application is not deemed mission critical.
<b>Component</b>	<b>Supervisory Control</b>	The operator console provides a remote interface to RTU. This interface is a requirement, as the RTU has no inherent capability to report or adjust voltage levels to an operator without the console. The operator console is thus deemed critical to the application.	The operator console will provide an operator with an interface to control Process XYZ. Without this console, the operator cannot remotely control the process. Thus, the operator console is deemed critical to the application.
	<b>Process Control</b>	The RTU is the core component of the remote control application. It is vital to the application, as the application can neither read nor adjust voltage levels remotely without an RTU. The RTU is thus deemed critical to the application.	The RTU is the core component of the remote control application. It is vital to the application, as the application can neither control nor monitor remotely without an RTU. The RTU is thus deemed critical to the application.
	<b>Field Instrumentation Control</b>	Both the analog voltage sensor and actuator are vital components to this application. Without the sensor and actuator, remote control of voltages is impossible. The sensor and actuator are thus deemed critical to the application.	The instrumentation control components are vital to the ability to remotely monitor and control Process XYZ. They are thus deemed critical to the application.

### 3. Data Sensitivity

SCADA systems, as noted, typically control critical systems. Whether the data processed by the system is sensitive and subject to compromise or loss must be examined in order to determine how best to protect it. The assessment must be made both in terms of the system as a whole, and as individual components, and must address both operational and managerial aspects. Several questions that should be answered include:

- Is the data returned by the process of a sensitive nature such that loss, modification or compromise of the data, either intentional or unintentional, will cause serious harm to the organization's mission?
- Are the instructions transmitted to the process of a sensitive nature such that loss, modification or compromise of the instructions, either intentional or unintentional, will cause serious harm to the organization's mission?

- What type and amount of system data loss, modification or compromise is acceptable?
- Is the data retained or transmitted by the individual components subject to loss, modification or compromise, either intentionally or unintentionally, to a degree that the system will be affected?
- What level of component data loss, modification or compromise is acceptable?

The sensitivity of data at the boundary and the sensitivity of data at an individual component will not be the same, but they will support each other. Aggregation of data may be a concern. Table 6 describes a sample Data Sensitivity examination for a SCADA system that monitors electrical voltage.

Table 6 Sample Data Sensitivity analysis for the Electrical Demonstration Laboratory

<i>Data Sensitivity - Electrical Demonstration Laboratory</i>			
		<b>Operation</b>	<b>Management</b>
<b>Boundary</b>		The Electrical Demonstration Laboratory transmits voltage level readings in two directions (to the sensors and to the operator console, via a network). This data is sensitive in that an incorrect reading, whether intentional or unintentional, can cause serious bodily harm to individuals or damage sensitive electrical equipment.	The Electrical Demonstration Laboratory provides sensitive data; a loss or compromise of the data could significantly affect the stability of Process XYZ. Further, decisions outside the boundary are made based on the application's data.
<b>Component</b>	<b>Supervisory Control</b>	The operator console must accurately reflect the current voltage state. It must also correctly indicate when an override condition has occurred. The data at the console must thus be accurate and timely.	The operator console must accurately reflect the current state of Process XYZ. A high level of confidence is required in the data displayed on the console.
	<b>Process Control</b>	Data received from the sensors must be stored in the RTU accurately and in a timely manner. Requests from the operator console must be passed accurately and quickly to the actuators. Further, the data must be accurate in order for an override condition to be detected.	The RTU must accurately and quickly report the status of Process XYZ to the operator console. It must also control Process XYZ as the operator intended with minimal variance.
	<b>Field Instrumentation Control</b>	Both the analog voltage sensor and actuator must be reliable, accurate and well calibrated. Incorrect readings can cause a chain reaction of bad information transmitted throughout the entire application.	The instrumentation control components must accurately report the status of Process XYZ, and must provide accurate control.

#### 4. Operating Environment

SCADA system hardware components are designed for industrial environments, and thus offer robust features for operation in austere environments. These features, however, do not address the management related concerns of security professionals such as data protection and controlled access to components. Understanding how a SCADA system is designed requires understanding the environment it operates in, both for operations functions and management functions. Some questions that should be answered include:

- What environmental factors will affect the process, either negatively or positively?
- What environmental factors will affect the system components, either negatively or positively?
- What is an acceptable level of interference by environmental factors?
- How should these factors be mitigated?

Table 7 describes a sample Operating Environment examination for a SCADA system that monitors electrical voltage.

Table 7 Sample Operating Environment analysis for the Electrical Demonstration Laboratory

<b>Operating Environment - Electrical Demonstration Laboratory</b>			
		<b>Operation</b>	<b>Management</b>
<b>Boundary</b>		The Electrical Demonstration Laboratory operates in a variety of environmental conditions. The operator console is in an environmentally controlled room. The RTU, sensors and actuators are in an industrial space. The network used to communicate between the operator console and the RTU is the facility LAN.	The Electrical Demonstration Laboratory operates within a larger limited access facility. The operator console is in a locked room, and the RTU, sensors and actuators are in an open industrial space accessible by all personnel in the facility. The facility LAN is used to connect the operator console with the RTU.
<b>Component</b>	<b>Supervisory Control</b>	The operator console is in an environmentally controlled room.	The operator console is in a locked room, with access to the room limited to a screened subset of the facility's personnel.
	<b>Process Control</b>	The RTU is in an industrial space subject to extremes in temperature and humidity. The network used to communicate between the operator console and the RTU is within a larger enclosed physical area but operates within a logically open network.	The RTU is in an open space accessible by all personnel in the facility, including personnel untrained and uncleared for access to the data. The facility LAN is used to connect the operator console with the RTU; access to the network is available to all facility personnel.
	<b>Field Instrumentation Control</b>	The sensors and actuators are in an industrial space subject to extremes in temperature and humidity. The cables used to monitor and control the voltage levels are subject to industrial spillage and salt water corrosion.	The sensors and actuators are in an open space accessible by all personnel in the facility, including personnel untrained and uncleared for access to the data.

## 5. System Interfaces

A complex system will likely have many interfaces, each of which may become an avenue of attack. All interfaces must be closely examined and evaluated in order to understand how it must be protected, both system wide and at the individual components.

Some questions that should be answered include:

- What interfaces exist for data to flow out of the system?
- What interfaces exist for instructions to flow into the system?
- What level of access is required to the feedback data returned by the process? Who requires access to the data?
- What level of access is required to send instructions to carry out commands against the process? Who requires the capability to transmit instructions to the process?
- What protections exist or can be applied to minimize the exposure of vulnerable interfaces by the system?
- What interfaces exist on the components for data or instructions to flow into or out of the component?
- What interfaces exist within the components for data or instructions to flow between components?
- What protections exist or can be applied to minimize the exposure of vulnerable interfaces by the components?

Table 8 describes a sample System Interfaces examination for a SCADA system that monitors electrical voltage

Table 8 Sample System Interfaces analysis for the Electrical Demonstration Laboratory

<b>System Interfaces - Electrical Demonstration Laboratory</b>		
	<b>Operation</b>	<b>Management</b>
<b>Boundary</b>	The Electrical Demonstration Laboratory provides an interface to the Management Information System through a Microsoft Open DataBase Connectivity (ODBC) server residing on the operator console. The ODBC server allows queries from a variety of servers (web, SQL server) to return data. Instructions to the process are executed through a Graphical User Interface (GUI) on the operator console.	The Electrical Demonstration Laboratory provides the capability for external servers to query the data residing on a database on the operator console. This data is queried every two minutes by a SQL server for archiving and by a web server for real-time status updates. No modification of Process XYZ is performed by these queries. (The operator console is the only interface which allows modification of Process XYZ.) The SQL server applies permissions to the queries it allows based on the organizational rules. The web server is available to all on the intranet.
<b>Component</b>	<b>Supervisory Control</b>	The operator console runs the RSView32 implementation of ODBC, and is configured to allow queries from two specific servers; all other servers are denied access to the ODBC connection. Access is through the Ethernet port on the operator console, which is connected to the facility LAN. The operator console is connected to the RTU through the same Ethernet port.
	<b>Process Control</b>	The RTU is connected to the facility LAN via an Ethernet connection with a static IP address. There are no protections on the port for restricting access, but only the RSView32 software is believed to have the capability to address the RTU correctly. The RTU is connected to the sensors and actuators through pairs of wires from the Input/Output cards. These wires run through conduits from the RTU to the sensors and actuators.
	<b>Field Instrumentation Control</b>	The sensors and actuators are directly connected to the RTU through wires running through conduits.
		The operator console runs the RSView32 implementation of ODBC, and is configured to allow remote LAN-based queries from two specific servers; all other servers are denied remote access. The operator console is connected to the RTU through the same Ethernet port. The operator console uses the built-in Microsoft Windows XP firewall.
		The RTU is connected to the facility LAN via an Ethernet connection with a static IP address. There are no protections on the port for restricting access, but only the RSView32 software is believed to have the capability to address the RTU correctly. The RTU is connected to the sensors and actuators through pairs of wires from the Input/Output cards.
		The sensors and actuators are connected to the RTU through wires running through conduits. The conduits are shielded but several junctions are open.

## **6. Communications Requirements**

Since SCADA systems are designed for reliability, availability and data integrity, extra consideration must be given to confidentiality and authentication. Other issues to consider include protocols employed, types of interfaces required, hardware configuration, and budget. Some questions to answer include:

- What degree of reliability is required?
- What degree of availability is required?
- What degree of data integrity is required?
- What degree of confidentiality is required?
- What overhead and latency in transmission is acceptable?
- What is the environment the communications links must traverse?

Answering these questions will provide a guideline as to what the tradeoff between performance and security will be. Table 9 describes a sample System Interfaces examination for a SCADA system that monitors electrical voltage

Table 9 Sample Communications Requirements analysis for the Electrical Demonstration Laboratory

<b>Communications Requirements - Electrical Demonstration Laboratory</b>			
	<b>Operation</b>	<b>Management</b>	
<b>Boundary</b>	The Electrical Demonstration Laboratory needs to be accessible to the facility LAN via Ethernet and TCP/IP to allow remote control of the voltage and allow archiving of data in the MIS. The connections to the sensors and actuators are through wire. Accuracy and integrity are more important than confidentiality. Availability is a concern, but the local panel provides an alternate means of reading the data.	The Electrical Demonstration Laboratory must allow for archival data to be accessible to decision makers, in real time and over TCP/IP. Accuracy and integrity of the data are more important than the confidentiality. Availability is a concern, but the local panel provides an alternate means of reading the data. Both the operator console and the RTU have dedicated IP addresses.	
<b>Component</b>	<b>Supervisory Control</b>	The operator console requires a connection to the TCP/IP network to communicate with both the RTU and the MIS; this capability is built in to the operating system. Authentication of the ODBC query is accomplished using the ODBC connection parameters. Integrity is accomplished through the combined error-correction and detection capabilities of TCP/IP. There is no encryption because the overhead would significantly impact the timeliness and increase costs since specialized hardware is required.	The operator console provides remote LAN-based access to the data using ODBC over TCP/IP, which ensures accuracy of data with minimal delay. Authentication is accomplished through the ODBC connection configuration, which only accepts connections from the IP addresses of the correct servers. No encryption is performed for the archival data, although the data is considered sensitive because it provides a status of Process XYZ.
	<b>Process Control</b>	The RTU requires a TCP/IP connection to communicate with the operator console. The data transmitted to the operator console must arrive correctly, and TCP/IP provides that reliability with minimal overhead and delay. There are currently no encryption capabilities, and doing so would require specialized hardware. The connection to the sensors and actuators is a direct wired connection which is not encrypted. The data is raw voltage readings, which must arrive in a timely and accurate manner.	The RTU employs a TCP/IP connection to communicate with the operator console. The data transmitted to the operator console must arrive correctly, and TCP/IP provides that reliability although with a delay due to the overhead; this delay is not seen as critical. There are currently no capabilities to encrypt the data and doing so would require specialized hardware. The connection to the sensors and actuators is a direct wired connection which is not encrypted.
	<b>Field Instrumentation Control</b>	The sensors and actuators are directly connected to the RTU through wires running through conduits. There is no encryption due to costs and overhead.	The sensors and actuators are connected to the RTU through wires running through conduits. There is no encryption due to costs and overhead.

## **7. Hardware and Software (Operational Functions)**

When discussing operational functions of a SCADA system, the hardware and software to be used must also be evaluated. In the context of standard SCADA systems, reliability, stability and safety are primary concerns. Adding a security perspective introduces the concept of assurance, or ensuring that the hardware and software have minimal exploitable flaws. Questions to answer include:

- What degree of reliability should the system have with respect to software and hardware?
- What degree of assurance should the system have with respect to software and hardware?
- What degree of reliability do the components require in order to effectively satisfy the system's mission?
- What degree of assurance do the components require?
- Has the hardware been tested for reliability, safety, assurance, stability?
- Has the software undergone a formal documented software development process?
- Have the software and hardware formally analyzed or evaluated by a trusted third party?
- What is the configuration management and lifecycle maintenance process for the software, and the firmware update process for the hardware?
- What maintenance is required for the hardware?

## **8. Users and Personnel Actions (Management Functions)**

When discussing management functions, the users of the system and its components must also be evaluated. In addition, the automated decision making that is programmed into the system must be evaluated in the same terms. Some questions to answer include:

- Are the users cleared for all functions of the system? Which functions require clearance?
- What degree of automation of decision making is required before human intervention is required?
- What training do the users receive?
- What are the different roles required for operating the system?

- What are the different roles required for maintaining the system?

## 9. Conclusion

The nine considerations outlined above (summarized in Figure 11) must be examined for each component of the system in addition to the system as a whole, and must be examined in terms of both operational and managerial functions. The "matrix" approach will provide the necessary framework for understanding a SCADA system, its interfaces and communications requirements, and will serve to better define the security requirements and the countermeasures that must be employed to protect it. Chapter IV employs this methodology against the NPS SCADA Technology Testing Laboratory Model to illustrate how it can be beneficial.

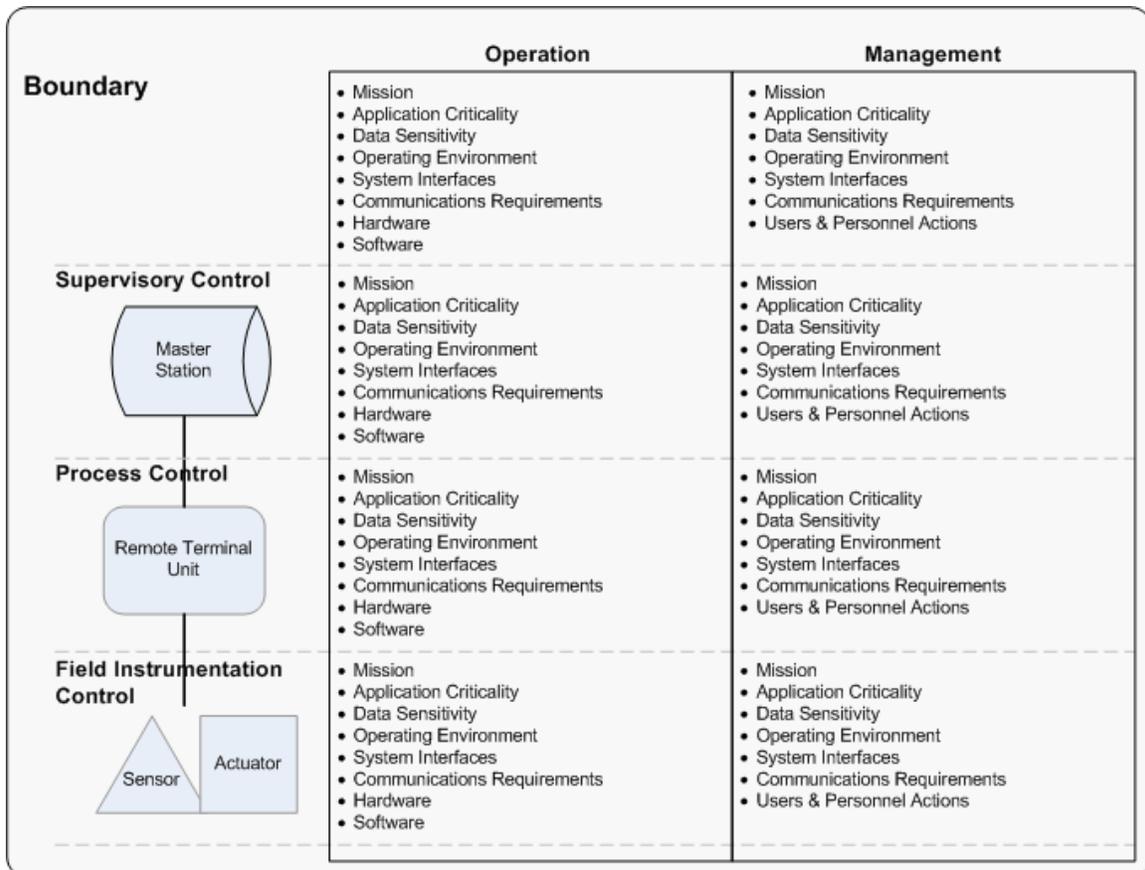


Figure 11 A SCADA system and its components must be evaluated in terms of nine considerations, for both operational functions and managerial functions.

## E. COMMUNICATIONS

The examples in the previous sections described scenarios that relied on a common communications infrastructure, mainly a TCP/IP Local Area Network. SCADA systems, by their very specialized nature, have unique communications requirements. Communications protocols (especially at the lower layers) are designed to meet the requirement that information arrive without errors in a timely manner (availability and data integrity). Confidentiality and authentication are not considerations in any of the existing SCADA protocols.

SCADA communications at the Field Instrumentation and Process Control layers will normally occur at Layer 1 (Physical) and Layer 2 (Data Link) of the standard Open System Interconnection (OSI) 7-Layer model; anything above the Data Link layer is difficult to define due to the specialized equipment available to SCADA consumers. Normally OSI Layers 3-6 (Network, Transport, Session, Presentation) are not even employed, and Layer 7 (Application) applications will normally communicate directly with the Data Link layer. Figure 12 illustrates the typical means of communications between SCADA components, which are detailed below.

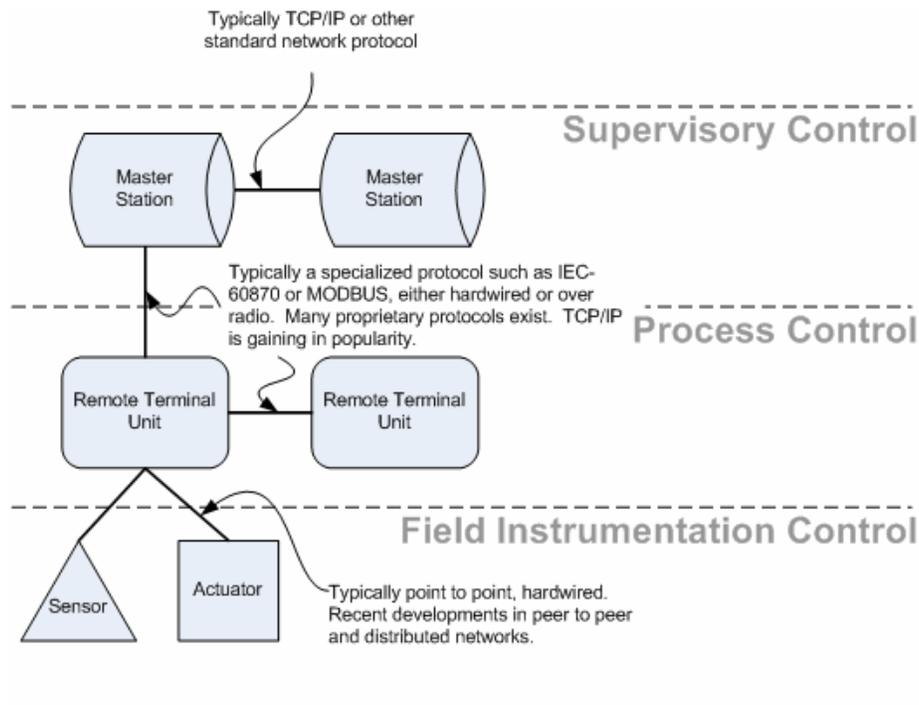


Figure 12 Typical SCADA systems will communicate through a variety of methods and protocols between components and layers.

### **1. Communications Among Supervisory Control Peers**

At the Supervisory Control layer between peers, communications are usually through a standard networking protocol such as TCP/IP or IPX over Ethernet or Token Ring. The SCADA applications involved in the communications are typically installed on standard PC systems capable of using standard operating-system provided protocols. Options exist for employing proprietary protocols between peers from the same manufacturer or SCADA-specific open source protocols between peers of dissimilar manufacturers.

### **2. Communications Between Supervisory Control and Process Control and Among Process Control Peers**

The means of communications between Supervisory Control and Process Control, as well as among Process Control peers, vary greatly and are usually dependent on hardware manufacturer. If a suite of RTUs, sensors and actuators is purchased from a single vendor, the protocol may be proprietary. Recent trends indicate a move to the option of using open source protocols in order to incorporate a variety of manufacturer's equipment, but the primary means of communication is typically a proprietary protocol. The protocols are designed for assurance and speed of delivery, requirements for a typical SCADA system managing a critical process. Interfaces can be proprietary, although many manufacturers support RS-232, and radio modems are often used to communicate with remote locations over microwave or Radio Frequency (RF) signals. Dedicated Ethernet networks are gaining wide acceptance, primarily due to the reliability of high-speed Ethernet where collisions are minimized. Encryption is rarely employed in these communications links, primarily due to the latency that encryption and decryption will cause.

### **3. Communications between Process Control and Field Instrumentation Control**

Communications between Process Control and Field Instrumentation Control is typically a point to point running over wire pairs, transmitting voltage pulses that are interpreted by the RTU based on how it is programmed. Recent trends in "intelligent" sensors and actuators provide more capabilities, and allow for peer to peer or distributed

networks among sensors and actuators; protocols in intelligent sensors will primarily be proprietary and vary among manufacturers.

## IV. LABORATORY CONFIGURATION

The Supervisory Control and Data Acquisition (SCADA) Technology Testing and Demonstration Model (SCADA Lab) at the Naval Postgraduate School (NPS) provides a functioning SCADA system in a laboratory setting using Rockwell Automation software and Allen-Bradley hardware. As part of this thesis, the configuration and installation of the laboratory components was initiated. The following sections outline two simple components of the lab. Follow on work will expand the lab facilities and provide an environment for further research and classroom instruction.

### A. HARDWARE CONFIGURATION

The process control equipment in the NPS SCADA Lab incorporates industry-standard software from Rockwell Automation and hardware from Allen-Bradley, a subsidiary of Rockwell Automation. The non-process control hardware and software in the laboratory is industry-standard and widely available.

#### 1. SLC-5/05

The SLC-5/05 is the latest in a sub-series of programmable process controllers developed by Allen-Bradley and marketed by Rockwell Automation. The SLC-500 series of controllers is designed for modularity and expandability, and provides flexibility in meeting feedback, control and communications requirements. They can be employed as standalone systems or as a component of a distributed control system, in either a local or remote locations, and can employ a wide range of communications for either management or process control. The modularity, expandability and reliability of the SLC-500 series accounts for its popularity and explains why the series is one of the more pervasive in the industry.

##### *a. Processor*

The SLC-5/05 model is characterized by a built-in Ethernet port to provide a means of communication with the controller for management functions. The Ethernet port can provide transport for TCP/IP and other protocols. Table 10 outlines the capabilities of the SLC-5/05, model number 1747-L551, employed in the NPS SCADA Lab.

Table 10 Features of the SLC-5/05 Programmable Controller [22]

Feature	SLC-5/05 (Model 1747-L551)	Comments
Memory Size (words)	16k	Other versions offer 32k or 64k
Power Supply Loading	1.0mA at 5v dc 200mA at 24v dc	Depending on availability of power at remote locations, end users can select either 5v dc or 24 dc to power sensors and other devices.
Max I/O Capacity	4096 discrete inputs and outputs	Over four thousand sensors or control systems can be managed from one CPU
Max local chassis/slots	3/30	Up to three chassis may be connected to one processor, with no more than 30 slots (for modules) employed at once.
On-board communications	Ethernet and RS-232	Other models offer DH+ (Data Highway Plus) or DH-485 protocols. The Ethernet protocol allows for TCP/IP communications. <sup>3</sup>
Programming Instruction Set	107 instructions	The programming is performed through programming software IDEs, usually installed on a PC and connected to the processor through Ethernet or RS-232.
Typical Scan Time	0.9 ms/K	The processor will scan through all of the input and output ports on the modules in this time period.

**b. Chassis & Power Supply**

The modular nature of the SLC-500 series implies a chassis-based design. Depending on the user requirements, Allen-Bradley offers either a 4-, 7-, 10- or 13-slot chassis [22]; the SLC-5/05 occupies one slot, freeing up the remaining slots for modules that provide input and output capabilities to sensors and actuators. If more than one chassis is to be employed locally (up to 3, with a maximum total of 30 slots), a ribbon cable connects the chassis. If the second or third chassis is to be remoted, one slot in the local chassis is reserved for a scanner module that communicates with an adapter module in the remote chassis.

The NPS SCADA Lab employs a 7-slot chassis, with an extra 3-slot chassis in reserve or for future expansion.

---

<sup>3</sup> Allen-Bradley and Rockwell Automation offer a variety of communications options for their products, both proprietary and open source. DH+ and DH-485 are proprietary. More information about the communications options available can be found at <http://www.ab.com/en/epub/catalogs/12762/2181376/214372/> (18 September 2004)

A variety of power supplies are available depending on requirements. The SLC-5/05 and chassis can be powered either through AC (85v to 265v ac) or DC (10v to 140v dc). The power supplies are attached to an external edge of the chassis.

The NPS SCADA Lab uses the Allen-Bradley Model 1746-P1 power supply. Table 11 describes the capabilities of the 1746-P1 power supply.

Table 11 1747-P1 Power Supply Characteristics [22]

Feature	1746-P1 Power Supply
Line Voltage	85 to 132V ac 170 to 265V ac (47 to 63Hz)
Internal Current Capacity	2A at 5V dc 0.46A at 24V dc
Maximum Inrush Current	20A
24V dc User Power Current Capacity	200mA
24V dc User Power Voltage Range	18-30V dc

*c. Analog Input/Output Modules*

(1) Input. The NPS SCADA Lab is configured for the following analog input modules:

- Allen-Bradley Model 1746-NI8. It provides eight analog input ports addressable as either voltage (-10vDC to +10vDC) or current (-20mA to +20mA). [22]
- Allen-Bradley Model 1746-NIO4V. It is a dual function module that provides two analog input ports and two analog output ports. The input ports are addressable as either voltage (-10vDC to +10vDC) or current (-20mA to +20mA). The output ports are addressable only as voltage (-10vDC to +10vDC). [22]

(2) Output. The NPS SCADA Lab is configured for the following analog output modules:

- Allen-Bradley Model 1746-NO4V. It provides four analog output ports addressable as voltage (-10vDC to +10vDC). [22]
- Allen-Bradley Model 1746-NIO4V. It is a dual function module that provides two analog input ports and two analog output ports. The input ports are addressable as either voltage (-10vDC to +10vDC) or current (-20mA to +20mA). The output ports are addressable only as voltage (-10vDC to +10vDC). [22]

**d. Digital Input/Output Modules**

The NPS SCADA Lab employs the Allen-Bradley Model 1746-IO12DC, a dual-function digital module that provides six input and six output ports with an operating ("on") range of 10vDC to 30vDC and an "off" range of 0vDC to 5vDC. [22]

**2. Programming Console**

**a. Hardware Configuration Baseline**

The minimum requirements, per the Rockwell Automation literature, are as follows:

- Intel Pentium II or greater microprocessor (Recommended: 500-MHz Pentium)
- 128MB of RAM for WinNT, Win2K or WinXP, 54MB RAM for Win98
- 45MB of HDD, or 115MB for additional options
- 256-color SVGA graphics adapter with 800x600 resolution
- CD-ROM Drive
- 3.5-inch 1.44 MB Disk Drive (required for software activation)
- Windows-compatible pointing device [23]

The systems in the NPS SCADA Lab meet or exceed these requirements and have the following properties:

- Intel Pentium 4 2.6GHz
- 1GB of RAM
- 120GB of HDD
- 32 bit NVIDIA GeForce4 MX graphics adapter with 1280x1024 resolution
- CD-ROM Drive
- 3.5-inch 1.44 MB Disk Drive
- Windows-compatible pointing device
- Broadcom 440X 10/100 MBit Ethernet Adapter

**b. Software Configuration Baseline**

The minimum requirements, per the Rockwell Automation literature, are one of either Windows 98, Windows 2000, Windows NT (Service Pack 6 or greater), or Windows XP [23]. The NPS SCADA Lab systems meet the Windows XP requirement.

### **3. Operator Console**

#### ***a. Hardware Configuration Baseline***

The requirements outlined in the Rockwell Automation literature vary depending on the complexity of the projects that the operator is intended to control. On the low end, Rockwell Automation recommends a Pentium 100 MHz CPU with 24MB RAM system. A more complex system may require a system similar to the one outlined above in the "Programming Console" section. [24]

The NPS SCADA Lab systems meets or exceeds these requirements.

#### ***b. Software Configuration Baseline***

The minimum requirements, per the Rockwell Automation literature, are one of either Windows 98, Windows 2000, Windows NT (Service Pack 6 or greater), or Windows XP [24]. The NPS SCADA Lab systems meet the Windows XP requirement.

### **4. Laboratory Configuration**

#### ***a. Simple Electrical Lab***

(1) Overview. A standard electrical panel, approximately 8"x8"x3", is used to demonstrate a simple SCADA electrical feedback and control system. A volt meter, a dial knob, and a switch are installed on the panel, and the panel is powered from the SLC-5/05 through the Analog Output module.

The volt meter measures and displays the voltage passing through the panel components; this same reading is passed to the SLC-5/05 through an analog Input/Output module to display on the operator console. The operator console provides a means for adjusting the voltage up or down remotely.

A switch allows for manual "override" control of the voltage passing through the panel. When this switch is closed, the voltage can be controlled at the panel, illustrating a "Manual Override" capability common in many applications. The voltage control knob on the panel now has full control over adjustment of the voltage. With the switch closed, if the voltage on the panel is adjusted, an alarm on the operator console indicates that a manual override condition has occurred.

The SCADA control features in this example are illustrated by the ability to control the voltage passing through the panel from the operator console. The

SLC-5/05 CPU will interpret the operator's desired direction of adjustment and transmit the required analog signals to the panel.

Two mechanisms are employed to illustrate the SCADA feedback features. In the first case, the voltage passing through the panel is relayed to the SLC-5/05 through an analog Input/Output module. In the second case, when the voltage on the panel is adjusted by an operator while in manual override mode, an alarm is raised on the operator console. The operator, after investigation, can reset the status of the override indicator.

(2) Physical Configuration - The SLC-5/05 is configured with an analog Input module that reads the voltage from the panel. This reading is processed and stored in a register in the SLC-5/05's Central Processing Unit (CPU). The SLC-5/05 is also configured with an analog Output module that supplies voltage to the panel, ranging from 0vDC to 10vDC. This voltage is controlled by adjusting the value in a register in the SLC-5/05 CPU.

The Input/Output modules are connected to the panel via two sets of wire pairs, configured as shown in Figure 13. The switch, manual override dial, and meter all receive voltage supplied from the Analog Output module in the SLC-5/05.

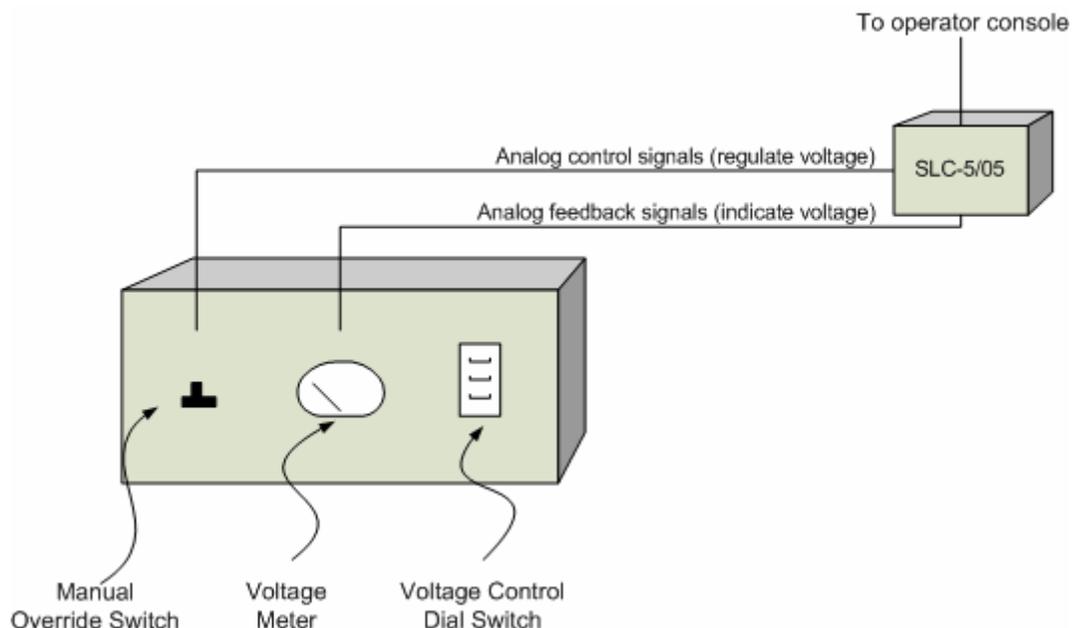


Figure 13 Physical Overview of Simple Electrical Lab

(3) Electrical Configuration - The electrical diagram in Figure 14 illustrates the basic electrical configuration for the lab. Power is supplied to the lab through the Analog Output module.

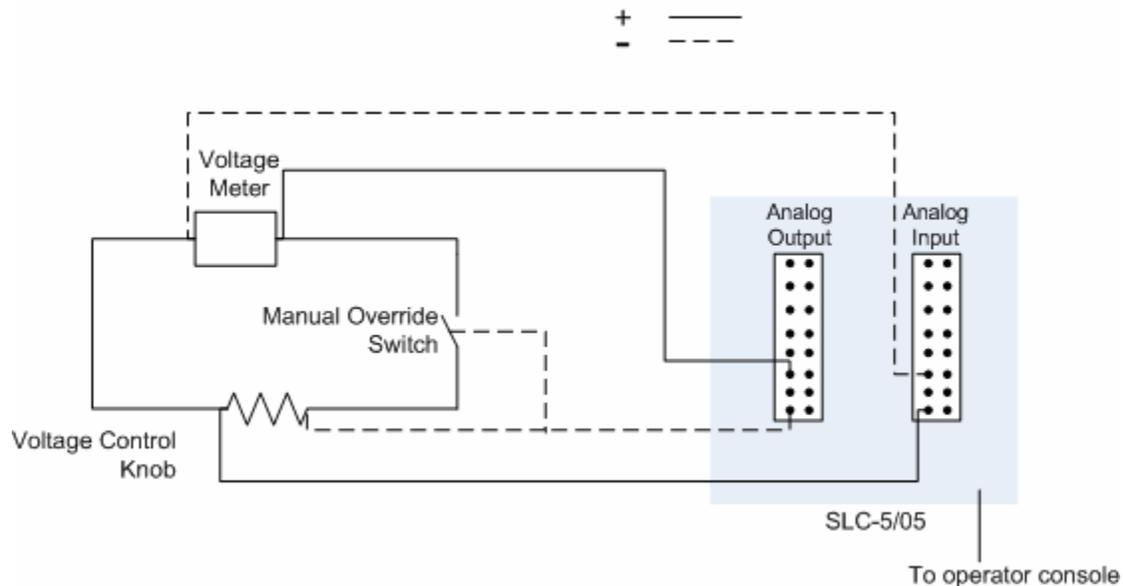


Figure 14 Electrical Overview of Simple Electrical Lab

**b. Simple Mechanical Lab**

(1) Overview. A wood block, approximately 2"x4"x7", serves as the housing for the components to illustrate the control and feedback mechanisms of this demonstration SCADA system. The robotic arm is attached to the housing along the top-most side. The housing is divided into three sections, labeled A, B and C. Embedded photo-electric sensors are found in each of the sections.

The operator console is programmed to receive the current location of a simulated barrel of hazardous materials from one of the three photo-electric sensors. Based on this information, the operator will have the option of moving the barrel to a different section of the housing using the robotic arm. The operator console also has an “Emergency Stop” option. There are no local controls as in the Simple Electrical Lab.

The SCADA control features in this example are illustrated by the robot arm, powered by a 9v battery (for the CPU) and 120v AC (for the servos).

The SLC-5/05 processor controls the robotic arm via the built-in RS-232 port, passing a series of three bytes through pins 3 (signal data) and 5 (signal ground) to the controller on the robot. The three bytes determine which servo to move (0 through 4) and the relative location across a 90 degree span. The operator can instruct the robot arm to perform any of a pre-determined set of actions.

The SCADA feedback features in this example are illustrated by three photo-electric sensors embedded in the housing. Each sensor is fed into a port pair in the analog Input/Output module of the SLC-5/05.

Since each sensor is independent of the other, the SLC-5/05 controller can determine when the barrel is covering any of the sensors, and will provide that feedback information to the operator console.

(2) Physical Configuration. An analog Input module receives voltage signals from the sensors via a pair of wires for each sensor, and stores the values in a register in the SLC-5/05 CPU. Logic programming in the CPU assesses the voltage reading from a sensor and determines if the sensor is currently covered or open, storing a bit value in another CPU register. This value is queried by the operator console regularly to determine the location of the barrel.

The robotic arm is connected to the SLC-5/05 by a standard 4-wire telephone cord (only two wires of which are actually used), plugged into a RJ-11 modular jack on the robot and a RS-232 connector in the RS-232 port of the SLC-5/05.

The robot arm is mounted on a housing as shown in Figure 15 and Figure 16. The sensors are embedded, evenly spaced in the housing, perpendicular to the center point of the robot arm.

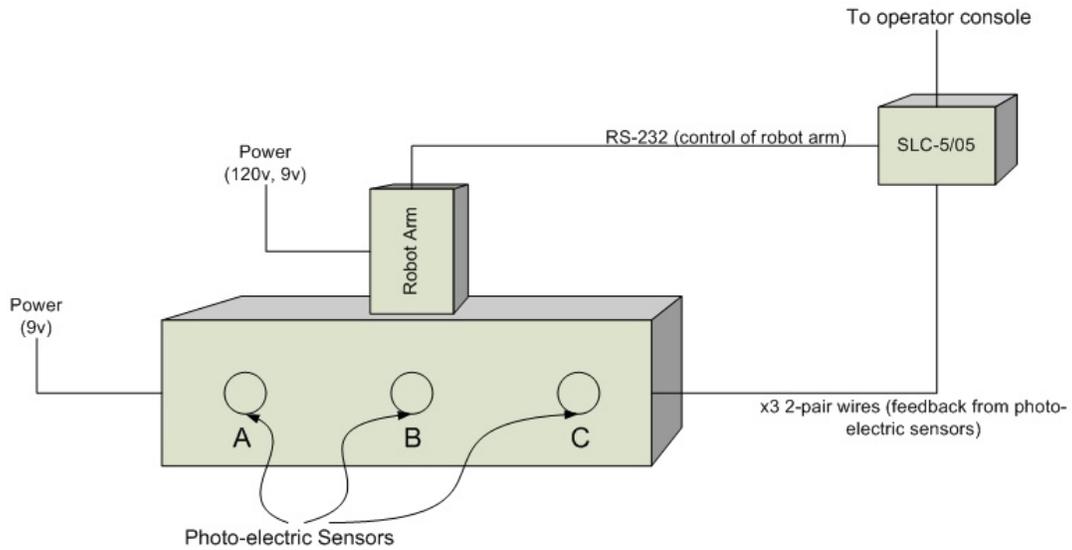


Figure 15 Physical Overview of Simple Mechanical Lab

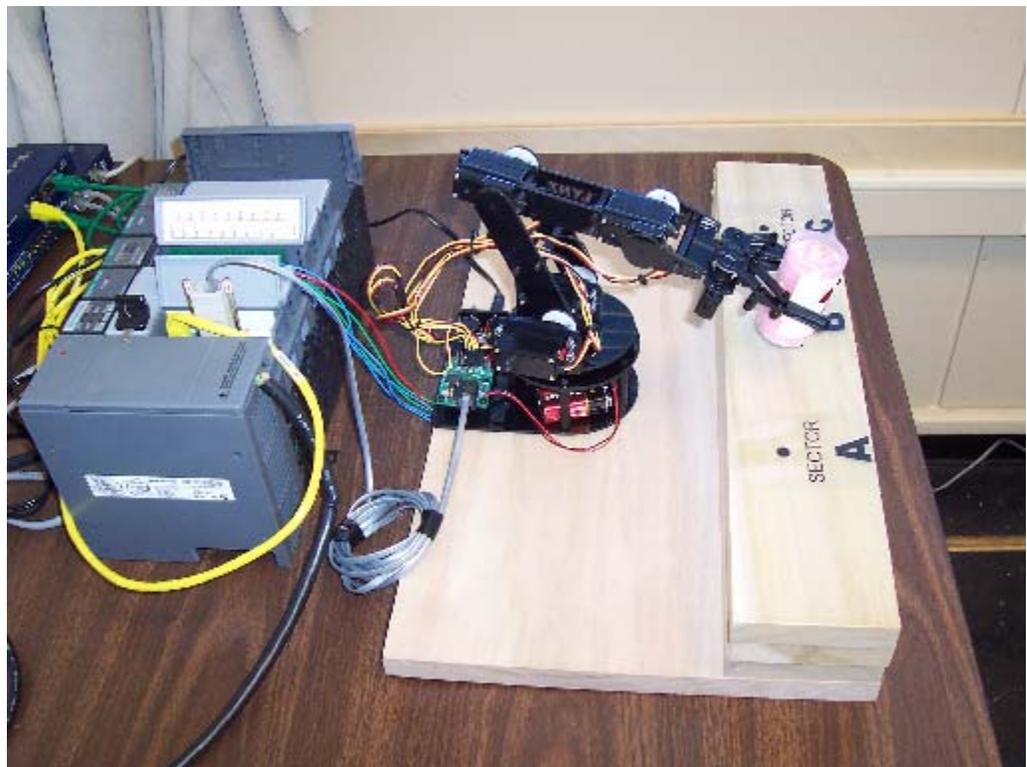


Figure 16 Photograph of Simple Mechanical Lab

(3) Electrical Configuration. Figure 17 illustrates the electrical configuration of the Simple Mechanical Lab. Power is supplied to the robotic arm CPU through a 9-volt battery. Power is supplied to the robotic arm servos through a

manufacturer-supplied 120vAC power adapter. The sensors are powered by a 9-volt battery.

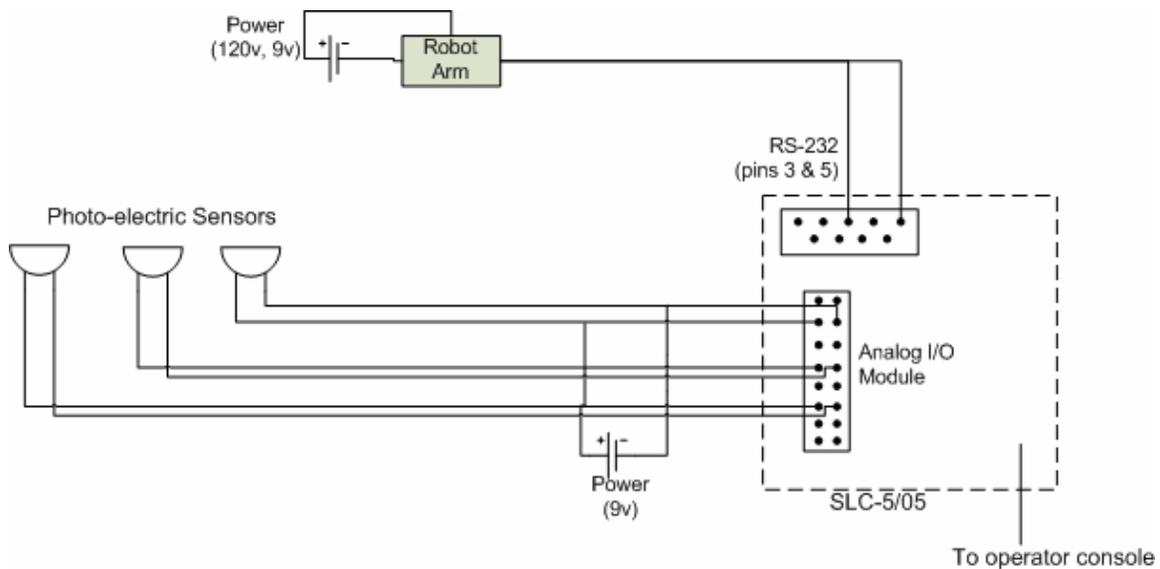


Figure 17 Electrical Overview of Simple Mechanical Lab

#### 4. Network and Communications

Since the NPS SCADA Lab simulates the real-world employment of SCADA systems, namely Internet-based management and control, the laboratory network has been designed to allow for notional access to the SLC-5/05 hardware through the NPS Local Area Network (LAN). The SLC-5/05 has been assigned a dedicated IP address by the NPS Information Technology and Communications Services (ITACS) office, and is currently accessible from anywhere on the NPS network. A series of hubs and switches co-located with the SLC-5/05 allow for physical separation from the NPS network if desired, allowing for attacks against the system to be performed without impacting users of the regular NPS LAN. This configuration, illustrated in Figure 18, allows for future research on incorporating firewalls, or other security protection mechanisms, into a SCADA protection plan.

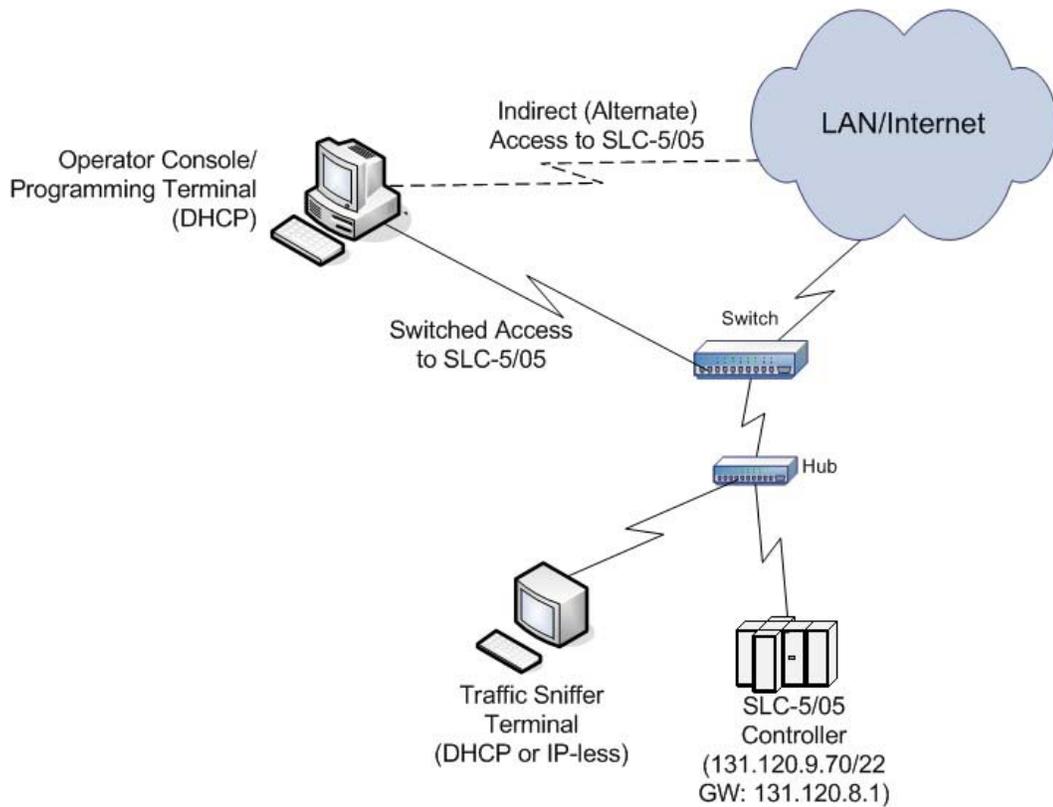


Figure 18 Network Topology

## B. SOFTWARE CONFIGURATION

### 1. Programming Console

#### a. *RSLinx (Communications)*

RSLinx runs on both the Programming Console and the Operator Console, and provides the communication subsystem for links between the Windows-based computer and the programmable controller. As shown in Figure 19, RSLinx provides an interface for Rockwell Automation software (RSLogix500, RSVIEW) to communicate with Rockwell Automation hardware products. It has an Application Programming Interface (API) which supports custom applications, and can serve as an Open Process

Control (OPC) Data Access Compliant Server<sup>4</sup> or as a Dynamic Data Exchange (DDE) Server.

RSLinx maintains a database of available Rockwell and Allen-Bradley hardware and software that can be used to easily configure the communications link between the two. In addition to a selection of hardware available, RSLinx also maintains a list of protocols that can be used to communicate with the hardware. RSLinx runs in the background and is started automatically upon boot-up of the Windows machine.

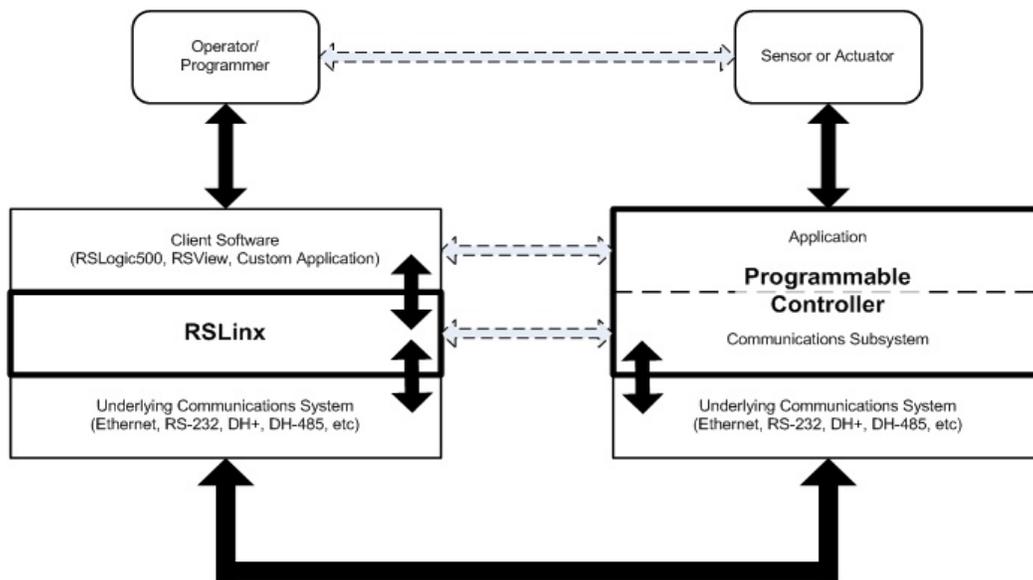


Figure 19 The RSLinx Software package from Rockwell Software provides an interface between the client application and the underlying process control protocol.

RSLinx in the NPS SCADA Lab is configured for Ethernet Access to the SLC-5/05, and provides interfaces for monitoring the status of the connection, as shown in Figure 20. It is also configured to employ TCP/IP over Ethernet.

<sup>4</sup> Open Process Control (OPC), also known as Object Linking and Embedding (OLE) for Process Control, is a series of specifications for supporting open connectivity in industrial automation. OPC uses Microsoft Distributed Component Object Model (DCOM) technology to provide a communication link between OPC servers and OPC clients.

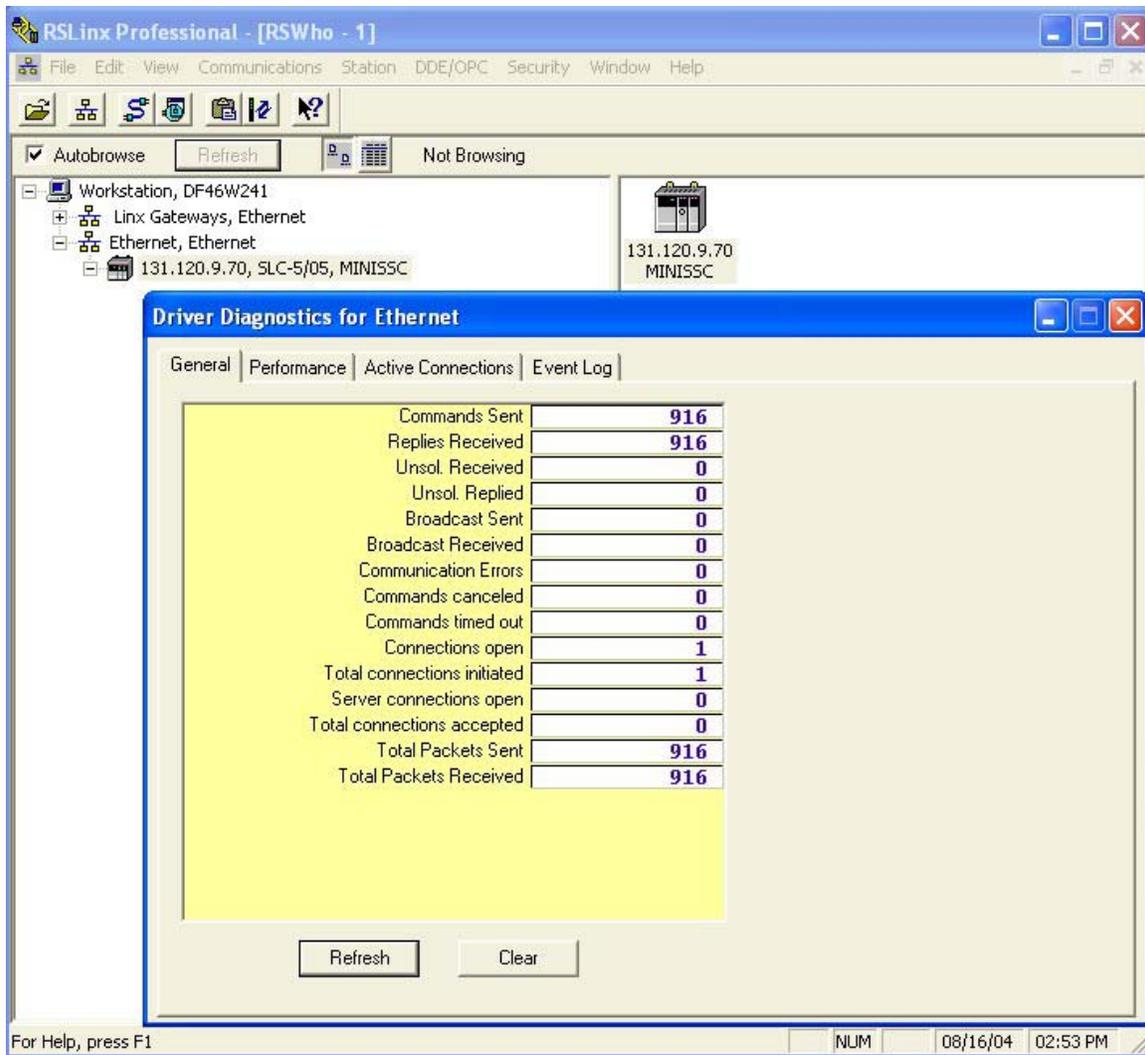


Figure 20 This screenshot from RSLinx illustrates the main screen, showing the Ethernet driver configured to communicate with a SLC-5/05 programmable controller.

***b. RSLogix500 (Programming IDE)***

RSLogix500 operates on the Programmer Console, and provides the integrated development environment (IDE) for programming the Rockwell or Allen-Bradley controllers. Programming can be conducted off-line and downloaded to the controller, or can be conducted on-line with changes taking effect immediately. Figure 21 illustrates a screenshot of the RSLogix500 main screen, and has been annotated to illustrate the basic components and actions available to a programmer.

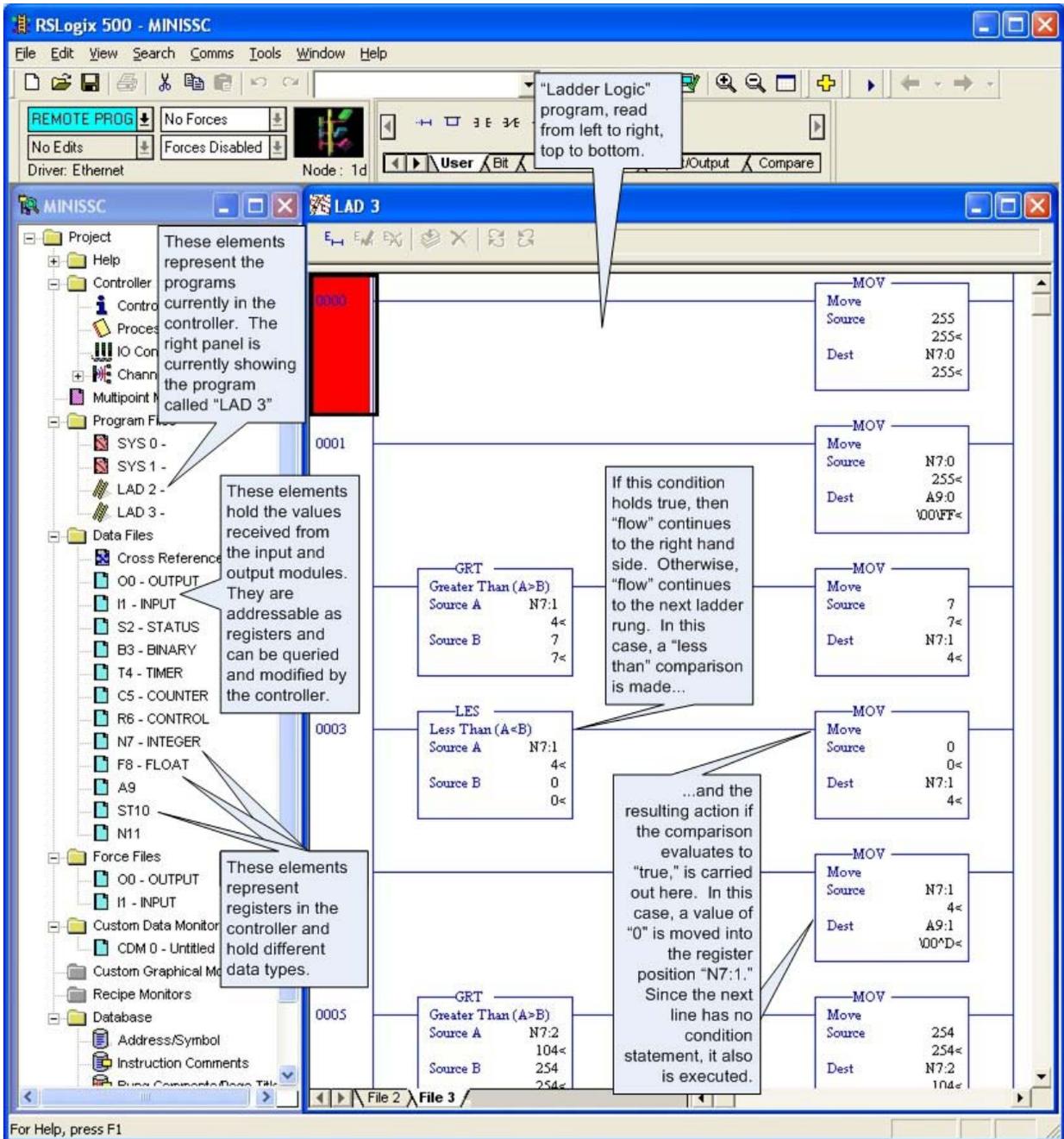


Figure 21 A screenshot of the RSView32 IDE main screen, annotated to show the different SLC-5/05 control functions available to a programmer.

*c. RSView32 Works (Operator Interface IDE)*

RSView32 Works provides an IDE for development of operator console applications. It provides libraries and pre-built components that can be used to design

sophisticated graphical user interfaces (GUI) that can interface with Rockwell or Allen-Bradley controllers directly. While RSLogix 500 provides an interface to program the controller, RSVIEW32 Works provides an interface on which to build custom applications that can interact with a controller. RSVIEW32 Works, for example, can develop the application that a shipboard operator would use to monitor pressure levels in a fuel line. Figure 22 shows a sample RSVIEW32 Works application, presenting an attractive interface to an operator who is responsible for monitoring a process.

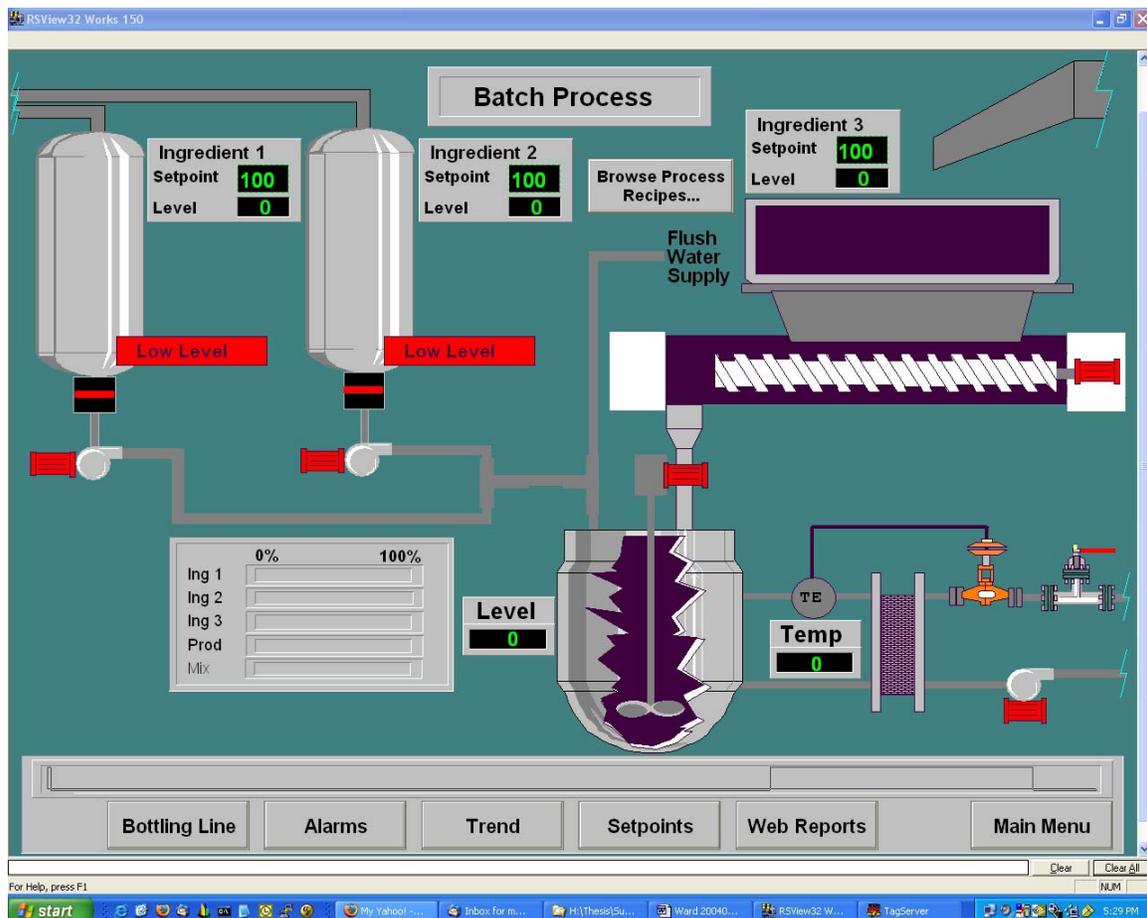


Figure 22 A screenshot of a RSVIEW32 Works sample application showing an operator interface to a process monitoring system. [19]

2. **Operator Console**  
 a. *RSLinx (Communications)*

RSLinx runs on both the Programming Console and the Operator Console, and provides the communication subsystem for links between the Windows-based

computer and the programmable controller. RSLinx provides an interface for Rockwell Automation software (RSLogix500, RSView) to communicate with Rockwell Automation hardware products. It has an Application Programming Interface (API) which supports custom applications, and can serve as an Open Process Control (OPC) Data Access Compliant Server or as a Dynamic Data Exchange (DDE) Server.

RSLinx maintains a database of available Rockwell and Allen-Bradley hardware and software that can be used to easily configure the communications link between the two. In addition to a selection of hardware available, RSLinx also maintains a list of protocols that can be used to communicate with the hardware. RSLinx runs in the background and is started automatically upon boot-up of the Windows machine.

***b. RSView32 Runtime (Operator Interface)***

RSView32 Works provides an IDE for development of operator console applications. It provides libraries and pre-built components that can be used to design sophisticated graphical user interfaces (GUI) that can interface with Rockwell or Allen-Bradley controllers directly. While RSLogix 500 provides an interface to program the controller, RSView32 Works provides an interface on which to build custom applications that can interact with a controller. RSView32 Works, for example, can develop the application that a shipboard operator would use to monitor pressure levels in a fuel line. Figure 22 shows a sample RSView32 Works application, presenting an attractive interface to an operator who is responsible for monitoring a process.

**3. SLC-5/05**

***a. Simple Electrical Lab***

(1) Overview. The simple electrical lab consists of two software components, the operator console and the SLC-5/05 ladder logic program. The operator interface, shown in Figure 23, provides several ways for the operator to adjust the voltage of the system: pushbuttons in .5vDC increments; "maximum voltage" and a "minimum voltage" pushbuttons; and a slider bar for incremental adjustments. The operator interface provides four means of displaying voltage: a vertical bar; an analog dial meter; a digital voltage readout, and a percentage of maximum readout. The interface also has an alarm indicating that a manual override condition has occurred, along with a reset button for the operator to use once the override condition has been

corrected. The binary RSView32 Works file that defines the interface is found on the operator console terminal's hard disk drive in the NPS SCADA Lab at C:\LABFILES\ELECTRICAL\. This file should be opened using the RSView32 application.

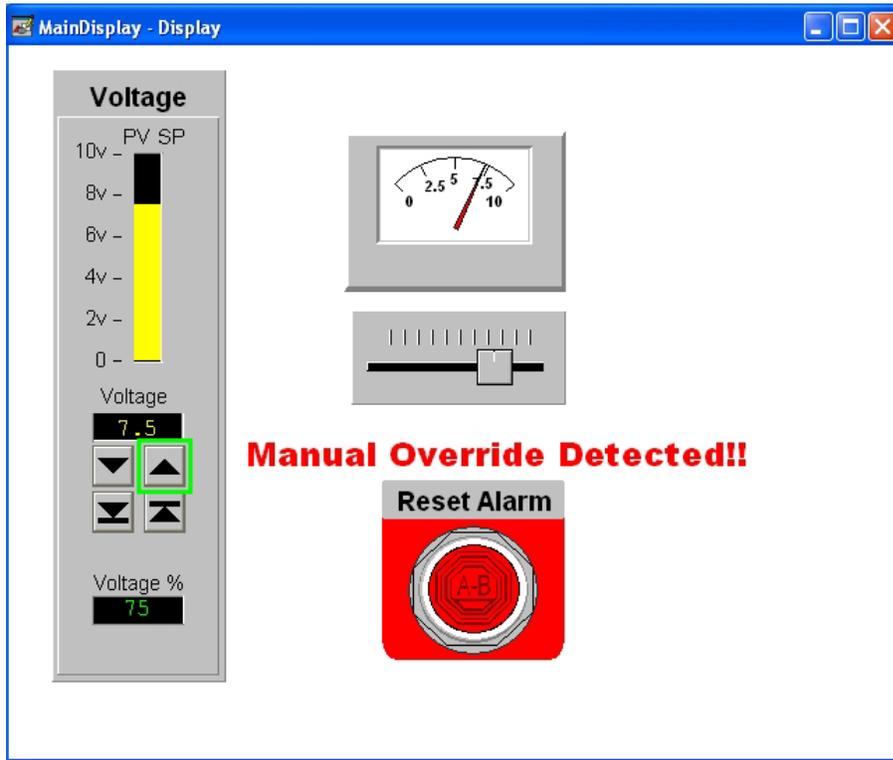


Figure 23 The main operator interface for the Simple Electrical Lab, showing the manual override alarm engaged.

(2) Ladder Logic Structure. The ladder logic program for the Simple Electrical Lab is located in Appendix B. Conceptually, the program follows the steps illustrated in Figure 24

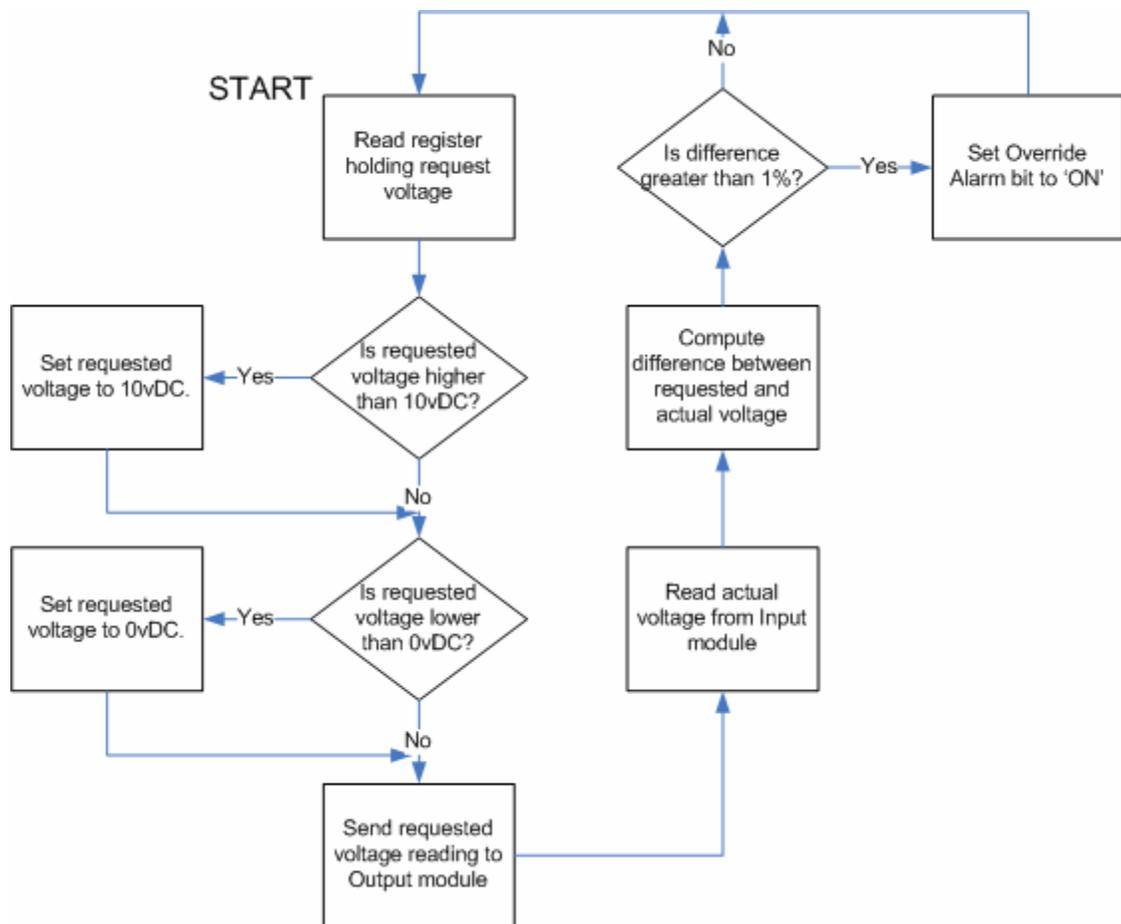


Figure 24 The logical flow of the Simple Electrical Lab Ladder Logic program.

The ladder logic program, reproduced in Appendix B, is loaded in the SLC-5/05 CPU, with the original file located on the operator console terminal's hard disk drive in the NPS SCADA Lab at C:\LABFILES\RSLOGIX\. The Simple Electrical Lab ladder logic is one portion of the SLC-5/05 ladder logic program called SCADALAB.RSS; it is the tab within the program called "Electrical." This file should be opened with RSLogix 500 application.

***b. Simple Mechanical Lab***

(1) Overview. The Simple Mechanical Lab consists of two software components, the operator interface and the ladder logic. The operator interface, illustrated in Figure 25, shows the current position of the simulated barrel of hazardous materials based on feedback from the sensors. The position of the barrel determines which of the three allowable actions (seen in the three buttons on the right side of the

interface in Figure 25) are available to the operator. The Emergency Stop feature will halt the robot movement after centering.

The data that determines the movement of the robot is coded in a set of three text files located on the operator console. Each file contains a series of instructions that will move the robot arm in one direction, and each instruction is sent individually to the SLC-5/05 for processing. The format of the commands is found in Appendix C, and the text files are found on the operator console terminal's hard disk drive in the NPS SCADA Lab at C:\LABFILES\ROBOT\MOVEMENT\. The binary RSView32 Works file that defines the interface is found in C:\LABFILES\ROBOT\.

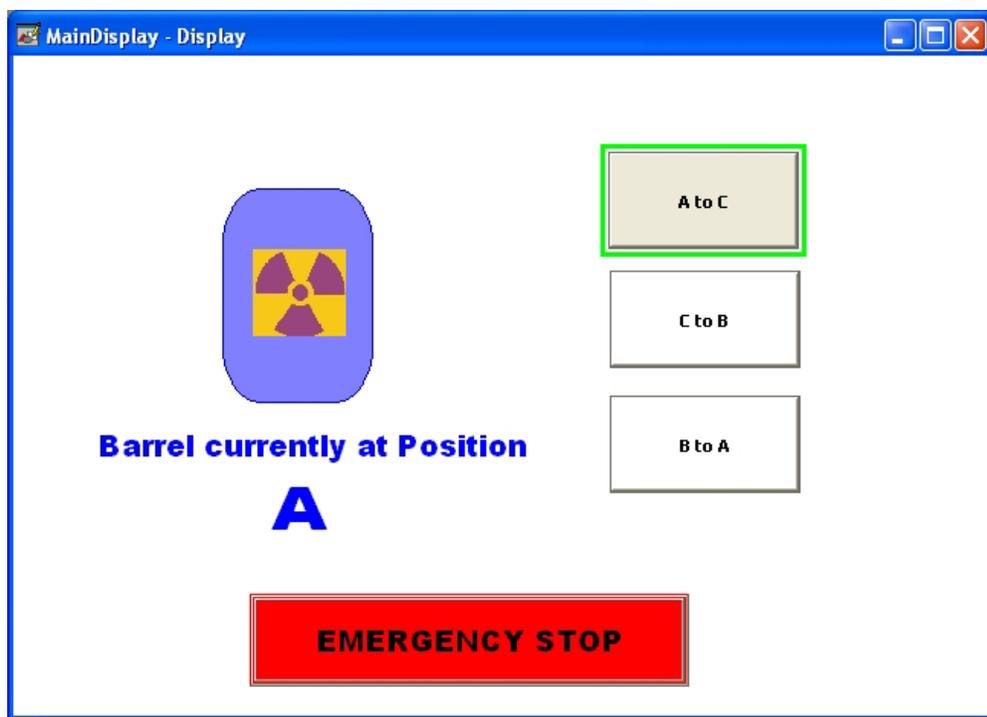


Figure 25 The main operator interface for the Simple Mechanical Lab, showing the current position of the simulated barrel at Position A.

(2) Ladder Logic Structure. The ladder logic program for the Simple Electrical Lab is located in Appendix B. Conceptually, the program follows the steps illustrated in Figure 26.

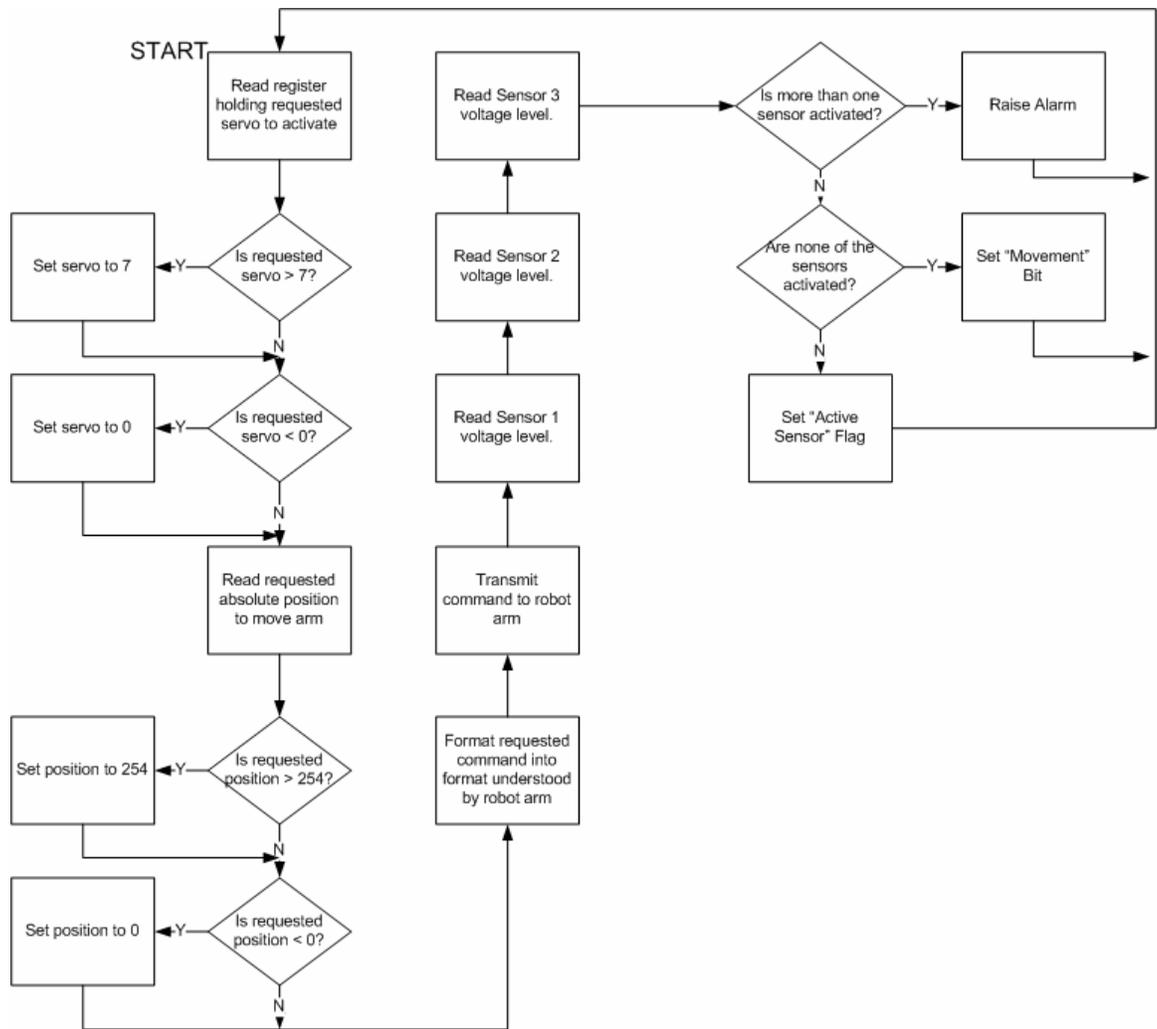


Figure 26 The logical flow of the Simple Mechanical Lab Ladder Logic program.

The ladder logic program, reproduced in Appendix B, is loaded in the SLC-5/05 CPU, with the original file located on the operator console terminal's hard disk drive in the NPS SCADA Lab at C:\LABFILES\RSLOGIX\. The Simple Mechanical Lab ladder logic is one portion of the SLC-5/05 ladder logic program called SCADALAB.RSS; it is the tab within the program called "Robot." This file should be opened with RSLogix 500 application.

## C. APPLICABILITY OF ARCHITECTURAL ANALYSIS TO LABORATORY

The NPS SCADA Lab can be described in terms of the Architectural Analysis outlined in Chapter III. The following sections describe both the Simple Electrical Lab and the Simple Mechanical Lab in terms of the steps given in the Architectural Analysis.

### 1. Simple Electrical Laboratory

#### a. Boundary

The boundary of the Simple Electrical Lab is illustrated in Figure 27. The Supervisory Control Layer contains the Personal Computer that serves as both the operator console and the programming console, running RSLinx, RSLogic 500, and RSView32 Works.. The SLC-5/05 (minus the input/output modules) is in the Process Control Layer, and the input/output modules, while physically located within the SLC-5/05, are logically located in the Field Instrumentation Control layer.

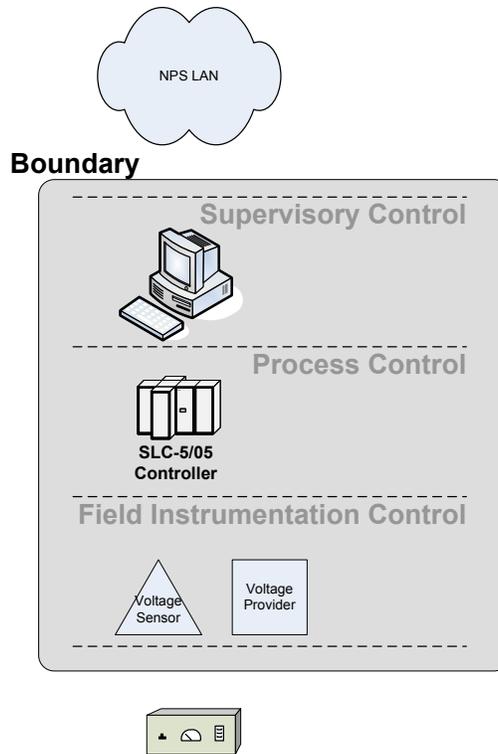


Figure 27 The boundary for the Simple Electrical Lab.

#### b. Operational and Management Functions Analysis

Table 12 contains the Operational Functions analysis of the Simple Electrical Lab, while Table 13 contains the Management Functions analysis.

Table 12 The Operational Functions analysis for the Simple Electrical Lab

OPERATIONAL FUNCTIONS	Boundary	Supervisory Control	Process Control	Field Instrumentation Control
<b>Mission</b>	Demonstrate basic feedback and control mechanisms of a SCADA System using a low voltage electrical system.	Provide a user interface for an operator to adjust a voltage remotely. Provide an interface to program the controller.	Apply the correct voltage to a simple electrical circuit. Read the voltage from the circuit. Monitor for a discrepancy between requested voltage and actual voltage.	Apply correct voltage to a circuit. Read voltage from a circuit.
<b>Application Criticality</b>	This application is a demonstration. Not considered critical.	The interface is not critical in terms of monitoring and controlling the voltage, as the voltage can be adjusted locally. The interface is critical for programming the controller.	The controller is critical because it supplies the required voltage to the circuit.	The modules are critical because they supply the required voltage to the circuit.
<b>Data Sensitivity</b>	The data is not considered sensitive since this is a demonstration.	The data is not considered sensitive since this is a demonstration.	The data is not considered sensitive since this is a demonstration.	The data is not considered sensitive since this is a demonstration.
<b>Operating Environment</b>	The system operates in a controlled access facility with adequate environmental controls. It is connected to the NPS intranet.	The interface is in a controlled access facility with adequate environmental controls. It is connected to the NPS intranet via a DHCP IP address allocation.	The controller is in a controlled access facility with adequate environmental controls. It is connected to the NPS intranet with a static IP address. It is bolted to a panel.	The system operates in a controlled access facility with adequate environmental controls. The modules are attached to the controller, which is bolted to a panel.
<b>System Interfaces</b>	To demonstrate the networking capabilities of the system, it is connected to the NPS intranet. No other interfaces are required.	To demonstrate the networking capabilities of the system, it is connected to the NPS intranet. No other interfaces are required.	To demonstrate the networking capabilities of the system, it is connected to the NPS intranet. The modules (Field Instruments) are directly attached to the controller.	The modules are directly attached to the controller. The modules are also connected to the circuit via 2 pairs of wires, one for input and one for output.
<b>Communications Requirements</b>	To demonstrate the networking capabilities of the system, it is connected to the NPS intranet via (a) an Ethernet port on the personal computer, and (b) an Ethernet port on the SLC-5/05. The system uses TCP/IP.	To demonstrate the networking capabilities of the system, it is connected to the NPS intranet via an Ethernet port on the personal computer. The system uses TCP/IP, which is built in to both the application and the operating system.	To demonstrate the networking capabilities of the system, it is connected to the NPS intranet via an Ethernet port on the controller. The system uses TCP/IP, which is built into firmware of the operating system.	The modules communicate with the controller via the backplane. The modules communicate with the circuit directly through 2 pairs of wires.
<b>Hardware</b>	The system requires a personal computer running Windows XP for the operator interface, a SLC-5/05 controller, an analog input module, and an analog output module.	The interface uses a personal computer running Windows XP for the operator interface.	The controller uses a SLC-5/05 controller.	The modules required are the Allen-Bradley model numbers 1746-NI8 and 1746-NO4V in a 7-slot chassis.
<b>Software</b>	Requires Windows XP, RSView32 Works, RSLinx, and RSLogix 500. The latest firmware for the SLC-5/05 is desired.	Requires Windows XP, RSView32 Works, RSLinx, and RSLogix 500. C.	Firmware version is OS501 Series C.	Firmware is current as of 9/2004.

Table 13 The Management Functions Analysis for the Simple Electrical Lab

MANAGEMENT FUNCTIONS	Boundary	Supervisory Control	Process Control	Field Instrumentation Control
<b>Mission</b>	Demonstrate basic feedback and control mechanisms of a SCADA System using a low voltage electrical system.	Provide a user interface for an operator to adjust a voltage remotely. Provide an interface to program the controller.	Apply the correct voltage to a simple electrical circuit. Read the voltage from the circuit. Monitor for a discrepancy between requested voltage and actual voltage.	Apply correct voltage to a circuit. Read voltage from a circuit.
<b>Application Criticality</b>	This application is a demonstration. Not considered critical.	The interface is critical for a proper demonstration.	The controller is critical because it is a vital component of the demonstration.	The modules are critical because they are vital components of the demonstration.
<b>Data Sensitivity</b>	The data is not considered sensitive since this is a demonstration.	The data is not considered sensitive since this is a demonstration.	The data is not considered sensitive since this is a demonstration.	The data is not considered sensitive since this is a demonstration.
<b>Operating Environment</b>	The system operates in a controlled access facility with adequate access controls. It is connected to the NPS intranet.	The interface is in a controlled access facility with adequate access controls. It is connected to the NPS intranet via a DHCP IP address allocation. The interface resides on a computer with password controlled access.	The controller is in a controlled access facility with adequate access controls. It is connected to the NPS intranet with a static IP address. It is bolted to a panel.	The system operates in a controlled access facility with adequate access controls. The modules are attached to the controller, which is bolted to a panel.
<b>System Interfaces</b>	To demonstrate the networking capabilities of the system, it is connected to the NPS intranet. No other interfaces are required.	To demonstrate the networking capabilities of the system, it is connected to the NPS intranet. No other interfaces are required.	To demonstrate the networking capabilities of the system, it is connected to the NPS intranet. The modules (Field Instruments) are directly attached to the controller.	The modules are directly attached to the controller. The modules are also connected to the circuit via 2 pairs of wires.
<b>Communications Requirements</b>	To demonstrate the networking capabilities of the system, it is connected to the NPS intranet via (a) an Ethernet port on the personal computer, and (b) an Ethernet port on the SLC-5/05. The system uses TCP/IP.	To demonstrate the networking capabilities of the system, it is connected to the NPS intranet via an Ethernet port on the personal computer. The system uses TCP/IP, which is built in to both the application and the operating system.	To demonstrate the networking capabilities of the system, it is connected to the NPS intranet via an Ethernet port on the controller. The system uses TCP/IP, which is built into firmware of the operating system. There are minimal protections for accessing the controller remotely.	The modules communicate with the controller via the backplane. The modules communicate with the circuit directly through 2 pairs of wires.
<b>Users and Personnel Actions</b>	Several personnel will be trained in demonstrating the system. These personnel must have a basic understanding of personal computer usage. Changes to the system will require personnel experienced in controller programming and basic electrical circuitry.	A shared account on the personal computer allows access to the demonstration. Several personnel will require access to the password. Users must have a familiarity with starting Windows applications. A programmer must have knowledge of RSLogix 500 and ladder logic programming.	Normal users should require little or no interaction with the controller. Programmers will require knowledge of ladder logic and the RSLogix-500 software.	Normal users should require little or no interaction with the field instrumentation units. Programmers will require knowledge of ladder logic, the RSLogix 500 software, and basic electrical concepts.

After completing the analysis, a security professional can better evaluate the security requirements for this system. In the Simple Electrical Lab, it is immediately obvious that the system is a low criticality system that does not require high assurance against compromise. However, several issues arise based on the results of this analysis:

- The operating system on the personal computer will require regular configuration management and patch application.
- The personal computer is connected to the NPS intranet, and thus must be treated at the same level of assurance as any other machine connected to the intranet.
- Some of the software on the personal computer is esoteric, and thus could be compromised by an intruder without any obvious outward signs to inexperienced users of the computer.
- The SLC-5/05 controller is connected to the NPS intranet, and its flaws are not well known. An intruder who manages to bypass the NPS physical or network security controls could compromise the controller and at a best disable the laboratory, and at worse the intruder could use the controller as a vector of attack against other targets.
- While the lab is a low voltage circuit with minimal current, there is still potential for mild electric shock that could result in damage to sensitive electrical components. While the controller and the modules within it are designed to take precautions to avoid this possibility, the system still needs to be monitored for potential safety issues.
- While the SLC-5/05 processor and the modules are not vital for the proper operation of the lab (the voltage can be controlled at the panel), they are considered vital for a proper demonstration of a SCADA system. This highlights well the distinction between operational and management analyses, and could affect how a security professional protects the system.

## **2. Simple Mechanical Laboratory**

### ***a. Boundary***

The boundary of the Simple Mechanical Lab is illustrated in Figure 28. The Supervisory Control Layer contains the Personal Computer that serves as both the operator console and the programming console, running RSLinx, RSLogic 500, and RSView32 Works.. The SLC-5/05 (minus the input/output modules and the RS-232 port) is in the Process Control Layer. The RS-232 port and the input/output modules, while

physically located within the SLC-5/05, are logically located in the Field Instrumentation Control layer. The light-sensitive sensors are in the Field Instrumentation Control layer.

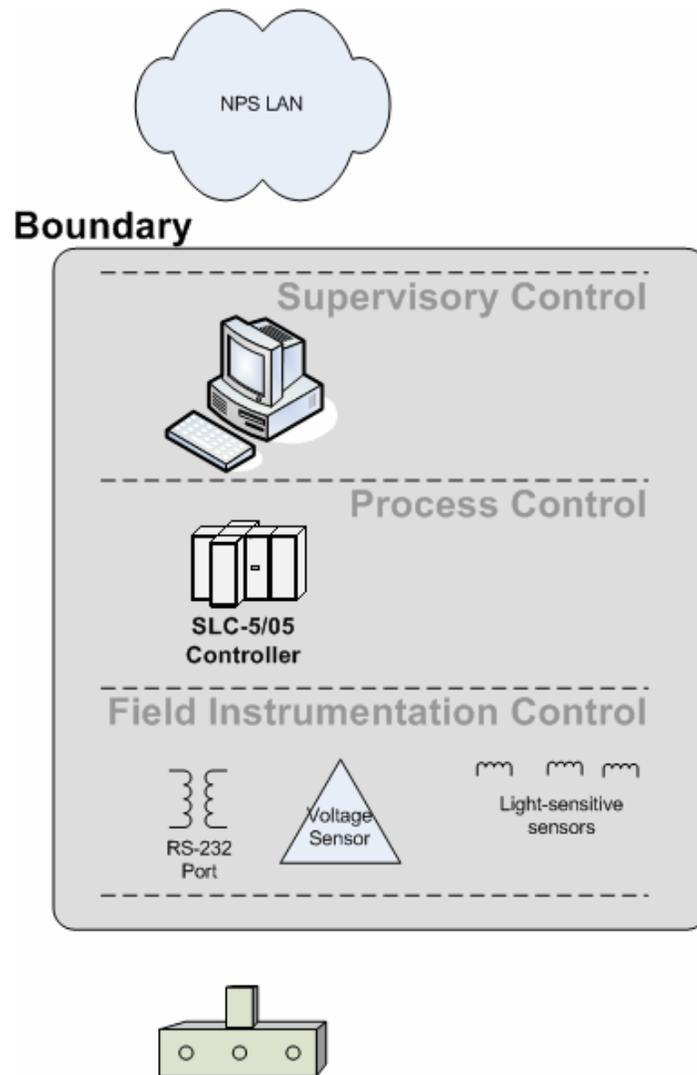


Figure 28 The boundary for the Simple Mechanical Lab.

***b. Operational and Management Functions Analysis***

Table 14 contains the Operational Functions analysis of the Simple Mechanical Lab, while Table 15 contains the Management Functions analysis.

Table 14 The Operational Functions analysis for the Simple Mechanical Lab

OPERATIONAL FUNCTIONS	Boundary	Supervisory Control	Process Control	Field Instrumentation Control
<b>Mission</b>	Demonstrate basic feedback and control mechanisms of a SCADA System using a robotic arm and light-sensitive sensors.	Provide a user interface for an operator to remotely operate a robotic arm. Provide an interface to program the controller.	Apply the correct commands to correctly move a robotic arm. Read and analyze the feedback from light sensitive sensors.	Transmit correct commands to the robotic arm through the RS-232 port. Read correct voltage levels from three light-sensitive sensors.
<b>Application Criticality</b>	This application is a demonstration. It is not considered critical.	The interface critical to monitoring and controlling the robotic arm. The interface is critical for programming the controller.	The controller is critical because it supplies the required commands to operate the robotic arm and receive feedback from the light-sensitive sensors.	The module is critical because it receives the voltage levels returned by the light-sensitive sensors. The RS-232 port is critical to the correct operation of the robotic arm.
<b>Data Sensitivity</b>	The data is not considered sensitive since this is a demonstration.	The data is not considered sensitive since this is a demonstration.	The data is not considered sensitive since this is a demonstration.	The data is not considered sensitive since this is a demonstration.
<b>Operating Environment</b>	The system operates in a controlled access facility with adequate environmental controls. It is connected to the NPS intranet.	The interface is in a controlled access facility with adequate environmental controls. It is connected to the NPS intranet via a DHCP IP address allocation.	The controller is in a controlled access facility with adequate environmental controls. It is connected to the NPS intranet with a static IP address. It is bolted to a panel.	The system operates in a controlled access facility with adequate environmental controls. The module and RS-232 port are attached to the controller, which is bolted to a panel.
<b>System Interfaces</b>	To demonstrate the networking capabilities of the system, it is connected to the NPS intranet. No other interfaces are required.	To demonstrate the networking capabilities of the system, it is connected to the NPS intranet. No other interfaces are required.	To demonstrate the networking capabilities of the system, it is connected to the NPS intranet. The module is directly attached to the controller. The robotic arm is connected through the RS-232 port.	The module is directly attached to the controller, and the light-sensitive sensors are connected via 3 pairs of wires. The robotic arm is connected via a cable from the RS-232 port.
<b>Communications Requirements</b>	To demonstrate the networking capabilities of the system, it is connected to the NPS intranet via (a) an Ethernet port on the personal computer, and (b) an Ethernet port on the SLC-5/05. The system uses TCP/IP.	To demonstrate the networking capabilities of the system, it is connected to the NPS intranet via an Ethernet port on the personal computer. The system uses TCP/IP, which is built in to both the application and the operating system.	To demonstrate the networking capabilities of the system, it is connected to the NPS intranet via an Ethernet port on the controller. The system uses TCP/IP, which is built into the firmware of the operating system.	The module communicates with the controller via the backplane. The module communicates with the sensors directly through wires. The RS-232 port communicates directly with the robotic arm through a cable.
<b>Hardware</b>	The system requires a personal computer running Windows XP for the operator interface, a SLC-5/05 controller, an analog input module, and an analog output module.	The interface uses a personal computer running Windows XP for the operator interface.	The controller uses a SLC-5/05 controller.	The module required is the Allen-Bradley model number 1746-N18 in a 7-slot chassis. The RS-232 port is built in to the SLC-5/05 controller. The cable for the RS-232 port is provided with the robotic arm.
<b>Software</b>	Requires Windows XP, RSView32 Works, RSLinx, and RSLogix 500. The latest firmware for the SLC-5/05 is desired.	Requires Windows XP, RSView32 Works, RSLinx, and RSLogix 500.	Firmware version is OS501 Series C.	Firmware is current as of 9/2004.

Table 15 The Management Functions analysis for the Simple Mechanical Lab

MANAGEMENT FUNCTIONS	Boundary	Supervisory Control	Process Control	Field Instrumentation Control
<b>Mission</b>	Demonstrate basic feedback and control mechanisms of a SCADA System using a robotic arm and light-sensitive sensors.	Provide a user interface for an operator to remotely operate a robotic arm. Provide an interface to program the controller.	Apply the correct commands to correctly move a robotic arm. Read and analyze the feedback from light sensitive sensors.	Transmit correct commands to the robotic arm. Read correct status from three light-sensitive sensors.
<b>Application Criticality</b>	This application is a demonstration. Not considered critical.	The interface is critical for a proper demonstration.	The controller is critical because it is a vital component of the demonstration.	The module is critical because it is a vital component of the demonstration.
<b>Data Sensitivity</b>	The data is not considered sensitive since this is a demonstration.	The data is not considered sensitive since this is a demonstration.	The data is not considered sensitive since this is a demonstration.	The data is not considered sensitive since this is a demonstration.
<b>Operating Environment</b>	The system operates in a controlled access facility with adequate access controls. It is connected to the NPS intranet.	The interface is in a controlled access facility with adequate access controls. It is connected to the NPS intranet via a DHCP IP address allocation. The interface resides on a computer with password controlled access.	The controller is in a controlled access facility with adequate access controls. It is connected to the NPS intranet with a static IP address. It is bolted to a panel.	The system operates in a controlled access facility with adequate access controls. The module is attached to the controller, which is bolted to a panel. The light-sensitive sensors are embedded in a block which is free-standing on the adjacent table.
<b>System Interfaces</b>	To demonstrate the networking capabilities of the system, it is connected to the NPS intranet. No other interfaces are required.	To demonstrate the networking capabilities of the system, it is connected to the NPS intranet. No other interfaces are required.	To demonstrate the networking capabilities of the system, it is connected to the NPS intranet. The Field Instruments are either directly attached to the controller or embedded in a wood block.	The module is directly attached to the controller. The RS-232 port is built in to the RS-232 controller. The light-sensitive sensors are connected to the sensor by 3 pairs of wires.
<b>Communications Requirements</b>	To demonstrate the networking capabilities of the system, it is connected to the NPS intranet via (a) an Ethernet port on the personal computer, and (b) an Ethernet port on the SLC-5/05. The system uses TCP/IP.	To demonstrate the networking capabilities of the system, it is connected to the NPS intranet via an Ethernet port on the personal computer. The system uses TCP/IP, which is built in to both the application and the operating system.	To demonstrate the networking capabilities of the system, it is connected to the NPS intranet via an Ethernet port on the controller. The system uses TCP/IP, which is built in to firmware operating system. There are minimal protections for accessing the controller remotely.	The module and RS-232 port communicate with the controller via the backplane. The light-sensitive sensors communicate with the module directly through 3 pairs of wires.
<b>Users and Personnel Actions</b>	Several personnel will be trained in demonstrating the system. These personnel must have a basic understanding of personal computer usage. Changes to the system will require personnel experienced in controller programming, basic electrical circuitry, and the robotic arm's instruction set.	A shared account on the personal computer allows access to the demonstration. Several personnel will require access to the password. Users must have a familiarity with starting Windows applications. A programmer must have knowledge of RSLogix 500 and ladder logic programming.	Normal users should require little or no interaction with the controller. Programmers will require knowledge of ladder logic and the RSLogix 500 software.	Normal users should require little or no interaction with the field instrumentation units. Programmers will require knowledge of ladder logic, the RSLogix 500 software, basic electrical concepts, and the robotic arm's instruction set.

After completing the analysis, a security professional can better evaluate the security requirements for this system. In the Simple Mechanical Lab, it is immediately obvious that it is a low criticality system that does not require high assurance against compromise. However, several issues arise based on the results of this analysis:

- The operating system on the personal computer will require regular configuration management and patch application.
- The personal computer is connected to the NPS intranet, and thus must be treated at the same level of assurance as any other machine connected to the intranet.
- Some of the software on the personal computer is esoteric, and thus could be compromised by an intruder without any obvious outward signs to inexperienced users of the computer.
- The SLC-5/05 controller is connected to the NPS intranet, and its flaws are not well known. An intruder who manages to bypass the NPS physical or network security controls could compromise the controller and at a best disable the laboratory, and at worst the intruder could use the controller as a vector of attack against other targets.
- While the lab is a low voltage circuit, there is still potential for electric shock that could result in damage to sensitive electrical components. While the controller and the modules within it are designed to take precautions to avoid this possibility, the system still needs to be monitored for potential safety issues.

## V. SUMMARY AND RECOMMENDATIONS

### A. SUMMARY

Control systems are prevalent through almost every sector of the economy, including those that comprise the critical infrastructure. Supervisory Control and Data Acquisition (SCADA) systems are one implementation of control systems, defined in this thesis as control systems involved in the distribution of a commodity. All SCADA systems have in common three layers:

- Supervisory Control
- Process Control
- Field Instrumentation Control

SCADA systems can implement a number of proprietary or open protocols to communicate across the three layers. Business systems establish policies and procedures for the system, and describe the process that is to be regulated. Boundaries are defined around components based on physical or functional characteristics, and are difficult to establish for complex or large SCADA systems.

SCADA systems can be defined by both operational and managerial functions; the unique capability of SCADA systems to operate for extended periods without human intervention requires that an active management plan be adopted to meet policy objectives, particularly with security concerns. The operational functions can be described in terms of eight concepts:

- Mission
- Application Criticality
- Data Sensitivity
- Operating Environment
- System Interfaces
- Communications Requirements
- Hardware
- Software

The management functions can be described in terms of seven concepts:

- Mission
- Application Criticality
- Data Sensitivity
- Operating Environment
- System Interfaces
- Communications Requirements
- Users and Personnel Action

Clearly defining these concepts for operational and management functions, and thoroughly understanding the implications that multiple boundaries can have on policy, helps in understanding a SCADA system's features, functions and capabilities, and thus how it can be protected.

A sample laboratory has been designed at the Naval Postgraduate School's SCADA Technology Testing Lab and Demonstration Model which can be described by the framework outlined above. This laboratory, built using industry standard hardware and software from Allen-Bradley and Rockwell, provides two working SCADA systems simulating real-world control applications.

## **B. RECOMMENDATIONS**

Based on research work conducted for this thesis, the following recommendations are made:

1. SCADA systems currently in operation should be re-evaluated against the framework presented in this thesis to determine if they meet minimum requirements for security.
2. New SCADA system designs should be evaluated with this framework as a guide to understand the security implications of the design.
3. Further research should be conducted on the implications of intersecting boundaries, and the effect on conflicting policies. Research on how best to systematically define boundaries should be conducted.
4. A formal analysis of the various open and proprietary communications protocols should be conducted to determine their suitability for high assurance SCADA systems.

5. A formal analysis of the software and hardware components of various vendor products should be conducted to determine their suitability for high assurance SCADA systems. Rockwell Automation and Allen-Bradley are leading developers and vendors of SCADA systems and their products have wide employment worldwide. Partnerships with Rockwell Automation and Allen-Bradley should be established to ensure access to source code and schematics for further research. Rockwell Automation has informally expressed an interest in partnering with NPS.

6. Partnerships with critical infrastructure agencies and organizations should be established to further our understanding of the real-world employment of SCADA systems. This partnership should include an assessment of current policies and practices.

7. An analysis of commercially available link encryption products should be conducted for SCADA systems where confidentiality, non-repudiation, or integrity of data is required. The Cisco Systems Critical Infrastructure Assurance Group ([http://www.cisco.com/security\\_services/ciag/](http://www.cisco.com/security_services/ciag/)) is currently researching and marketing such a line of products.

8. More realistic and complex SCADA systems should be developed at the NPS SCADA Technology Testing Lab. These systems should then be studied for vulnerabilities that could also be present in real-world systems.

THIS PAGE INTENTIONALLY LEFT BLANK

## LIST OF REFERENCES

- [1] Graham C. Goodwin, Stefan F. Graebe, and Mario E. Salgado, Companion Slides for *Control system design*, pp. 15 of Slides for Chapter 1, available from <http://csd.newcastle.edu.au/control/index.html> (24 May 2004), Prentice Hall, 2001.
- [2] David Bailey and Edwin Wright, *Practical SCADA for industry*, pp. 17-37, IDC Technologies, 2003.
- [3] David Bailey and Edwin Wright, *Practical SCADA for industry*, pp. 46-48, IDC Technologies, 2003.
- [4] Robert F. Dacey, "Critical Infrastructure Protection: Challenges in Securing Control Systems," Testimony before the Subcommittee on Technology, Information Policy, Intergovernmental Relations, and the Census, House Committee on Government Reform, GAO-04-140T, October 2003.
- [5] Perry Sink, "A comprehensive guide to industrial networks, Part 1: Why Use an Embedded Network or Fieldbus, and What Are the Most Popular Standards?" *Sensors Magazine*, available at <http://www.sensorsmag.com/articles/0601/28/main.shtml> (12 April 2004).
- [6] Steve Mackay, *Tutorials on Industrial Data Communications*, available at <http://www.idc-online.com/html/tutorial.html> (12 April 2004), I.D.C. Technologies (no date).
- [7] *Understanding SCADA System Security Vulnerabilities*, Ripstech White Paper, January 2001, Available from <http://itpapers.zdnet.com/abstract.aspx?scid=278&tag=tu.sc.ont.dir3&sortby=titled&docid=14905> (19 September 2004).
- [8] William J. Ackerman and Wayne R. Block, "Understanding supervisory systems," *IEEE Computer Applications in Power*, Vol. 5, Issue 4, pp 37-40, October 1992.
- [9] Lead Automation Analyst and Information Technology Specialist, private conversation at a major oil company drilling facility, Bakersfield, CA, 22 April 2004.

- [10] Tomas E. Dy-Liacco, "Modern Control Centers and Computer Networking," IEEE Computer Applications in Power, Vol. 7, Issue 4, pp. 17-22, October 1994.
- [11] website: <http://www.iec.ch/index.html> (26 April 2004)
- [12] website: [http://www.plcopen.org/pages/fr\\_tc1.htm](http://www.plcopen.org/pages/fr_tc1.htm) (26 April 2004)
- [13] Gordon Clarke and Deon Reynders, *Practical Modern SCADA Protocols*, pp. 63-311, IDC Technologies, 2004.
- [14] Gordon Clarke and Deon Reynders, *Practical Modern SCADA Protocols*, pp. 42-56, IDC Technologies, 2004.
- [15] Gordon Clarke and Deon Reynders, *Practical Modern SCADA Protocols*, pp. 349-361, IDC Technologies, 2004.
- [16] Gordon Clarke and Deon Reynders, *Practical Modern SCADA Protocols*, pp. 362-373, IDC Technologies, 2004.
- [17] website: <http://ethernet.industrial-networking.com/default.asp> (17 September 2004)
- [18] Rockwell Automation, *Netlinx*, Publication NETS-BR001A-EN-P, August 2000.
- [19] Rockwell Automation, "Demo, 'Process'," Sample program included in software package *RSView32 Works 150*, Release 7.0 Build 10, 2004.
- [20] The Instrumentation, Systems, and Automation Society (ISA), draft working copy of *Manufacturing and Control Systems Security, Part 1: Models and Terminology, Draft 1, Edit 2 (ISA-dS99.00.01)*, Figure 9, 2 August 2004.
- [21] The Instrumentation, Systems, and Automation Society (ISA), draft working copy of *Manufacturing and Control Systems Security, Part 1: Models and Terminology, Draft 1, Edit 2 (ISA-dS99.00.01)*, Figure 2, 2 August 2004
- [22] Rockwell Automation, *SLC-500 Programmable Controllers and I/O Modules*, Publication 1747-SO001B-EN-P, January 2001.

[23] Rockwell Automation, *RSLogix 500 Getting Results Guide*, Doc ID LG500-GR0001A-EN-P, 2002.

[24] Rockwell Automation, *RSView32 Users Guide*, Doc ID VW32-UM001A-EN-E, September 2003.

[25] Gordon Clarke and Deon Reynders, *Practical Modern SCADA Protocols*, pp. 177-285, IDC Technologies, 2004.

[26] Gordon Clarke and Deon Reynders, *Practical Modern SCADA Protocols*, pp. 66-142, IDC Technologies, 2004.

[27] Gordon Clarke and Deon Reynders, *Practical Modern SCADA Protocols*, pp. 42-45, IDC Technologies, 2004.

[28] Gordon Clarke and Deon Reynders, *Practical Modern SCADA Protocols*, pp. 45-56, IDC Technologies, 2004.

[29] Gordon Clarke and Deon Reynders, *Practical Modern SCADA Protocols*, pp. 349-355, IDC Technologies, 2004.

[30] Gordon Clarke and Deon Reynders, *Practical Modern SCADA Protocols*, pp. 355-361, IDC Technologies, 2004.

[31] Gordon Clarke and Deon Reynders, *Practical Modern SCADA Protocols*, pp. 45-56, IDC Technologies, 2004.

THIS PAGE INTENTIONALLY LEFT BLANK

## APPENDIX A - SCADA PROTOCOLS

### 1. IEC 60870

IEC-60870-5-101 (or T101) defines message structure for bit-serial communications over low bandwidth channels using existing widely used standards covering exchange between data terminal equipment (DTE) and data communication equipment (DCE). T101 defines link layer and application layer message format, as shown in Figure 29. This is a variable length frame; however, two fixed length frames are defined (which carry no data) for link layer control and acknowledgement messages. [25]

The maximum frame size is 261 bytes. The address field may be either 1 or 2 bytes, determined by a fixed system parameter and defined in advance of transmission, and defines the "secondary" recipient link layer address (for a multi-station network, a "primary" station is defined, and all other stations are secondary. If a primary station sends a message, the Address field will contain the secondary stations address. If a secondary station sends a message, the Address field will contain its own address so that the master station can identify who the message is from.) [25]

In IEC 60870-5-104 (or T104), the link layer definitions are not defined, as it is assumed that the transport and networking is performed with TCP/IP. The Application Service Data Unit (ASDU) as shown in the right hand side of Figure 29 is preserved.

In both T101 and T104, the number of ASDUs is limited to one for each message, and only one data type ("Type ID") is permitted for each ASDU, and the Type ID is the first field in the ASDU. The Type ID defines the general content of the message, whether it is process information, system information, parameters, or a file transfer. The Type ID will also define the direction the message is flowing, whether to the monitor station or to the control station. Reference [25] contains a table listing the 127 defined Type IDs.

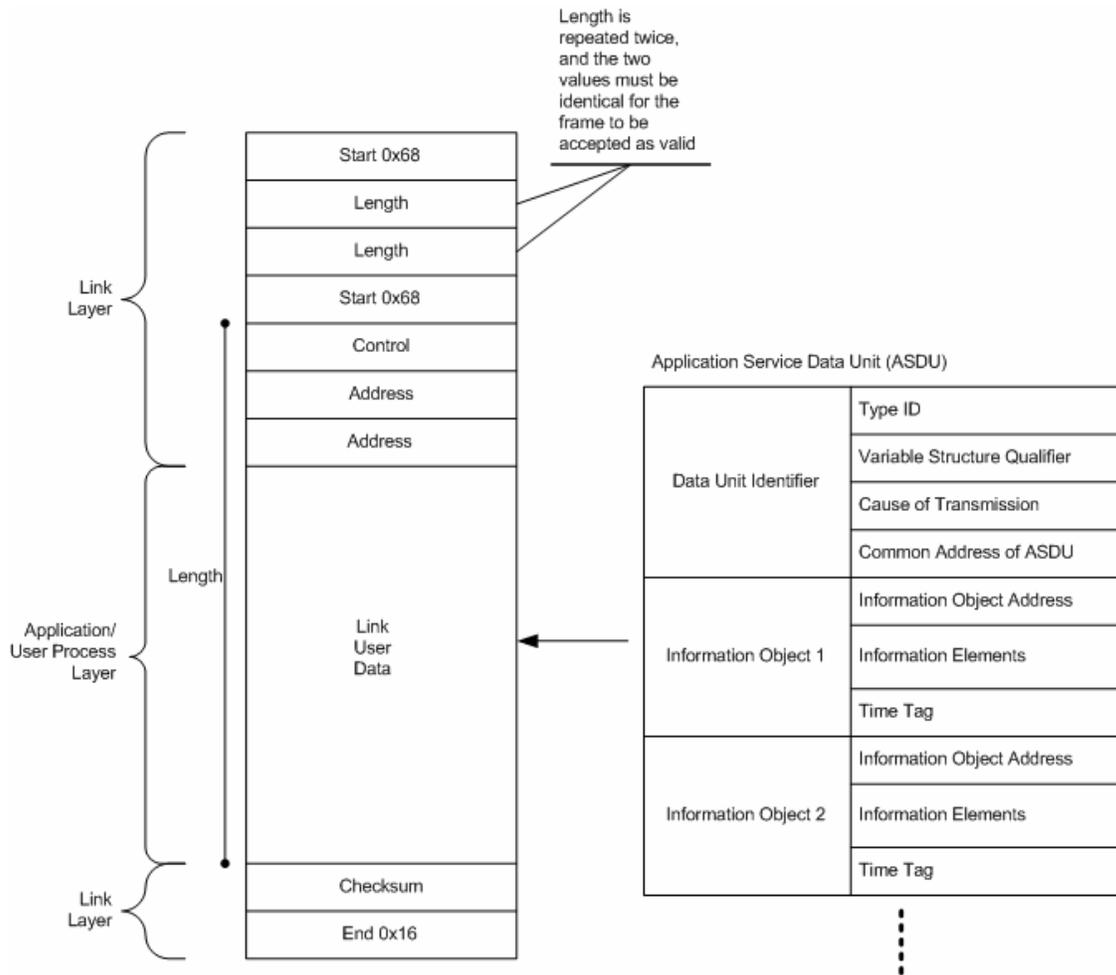


Figure 29 Message Structure under IEC 60870-5-101 (bit serial links). (After Ref. [25])

The ASDU Variable Structure Qualifier field specifies the number of information objects and how they are addressed in the message. The Cause of Transmission field is used at the recipient station to determine the correct task or program for processing the information object. The Common Address field is either one or two bytes and defines the recipient that will process the information objects. (It is called "Common Address" because all of the data contained in the message is destined for the same recipient.)

The Information Objects contain the actual data that is of interest to the SCADA process. There may be multiple information objects of different types in one message, or there may be one information object with multiple data values of the same type. The forty different types of information objects are divided into seven broad categories: Process, Protection, Commands, Time, Qualifiers, File Transfer, and Miscellaneous.

Clark and Reynders provide a list of information object types, and a detailed description of each. [25]

## **2. DNP3**

Distributed Network Protocol Version 3.3 (DNP3) is similar in function and purpose to IEC-60870-5-101, with which it shares a common origin. It is designed for reliably transmitting small packets of data in a deterministic, in-order sequence, which differentiates it from TCP/IP protocols and thus makes it more suitable for SCADA applications.

DNP3's link layer functionality allows for link initialization and error recovery and provides status indications to higher levels. Clark and Reynders describe the functions available in DNP3's link layer [26]. DNP3 introduces a Transport Layer (see Figure 30), which functions solely to disassemble large Application Layer data blocks into small enough for the Link Layer to transport. Because of this limited functionality, it is referred to as a pseudo-transport layer. The Transport Layer's one byte header provides disassembly and reassembly information so that 2048 byte data blocks from the application layer can be broken up into 250 byte blocks prior to delivery to the Link Layer and reassembled at the receiving end.

The Application Layer data unit header, as seen in Figure 30, is either 2 or 4 bytes long, depending on whether the message is sent as a request or a response. The Application Control field is primarily used for flow control, while the Function Code field defines the function that must be performed. A request message will offer eight types of functions: transfer (confirm, read or write), control, freeze, application control, configuration, or time synchronization). A response message offers three: confirm, read, or write. Clark and Reynders describe the functions available. [26] The Internal Indicators field in a response packet is a two byte field that describes the status of the responding station.

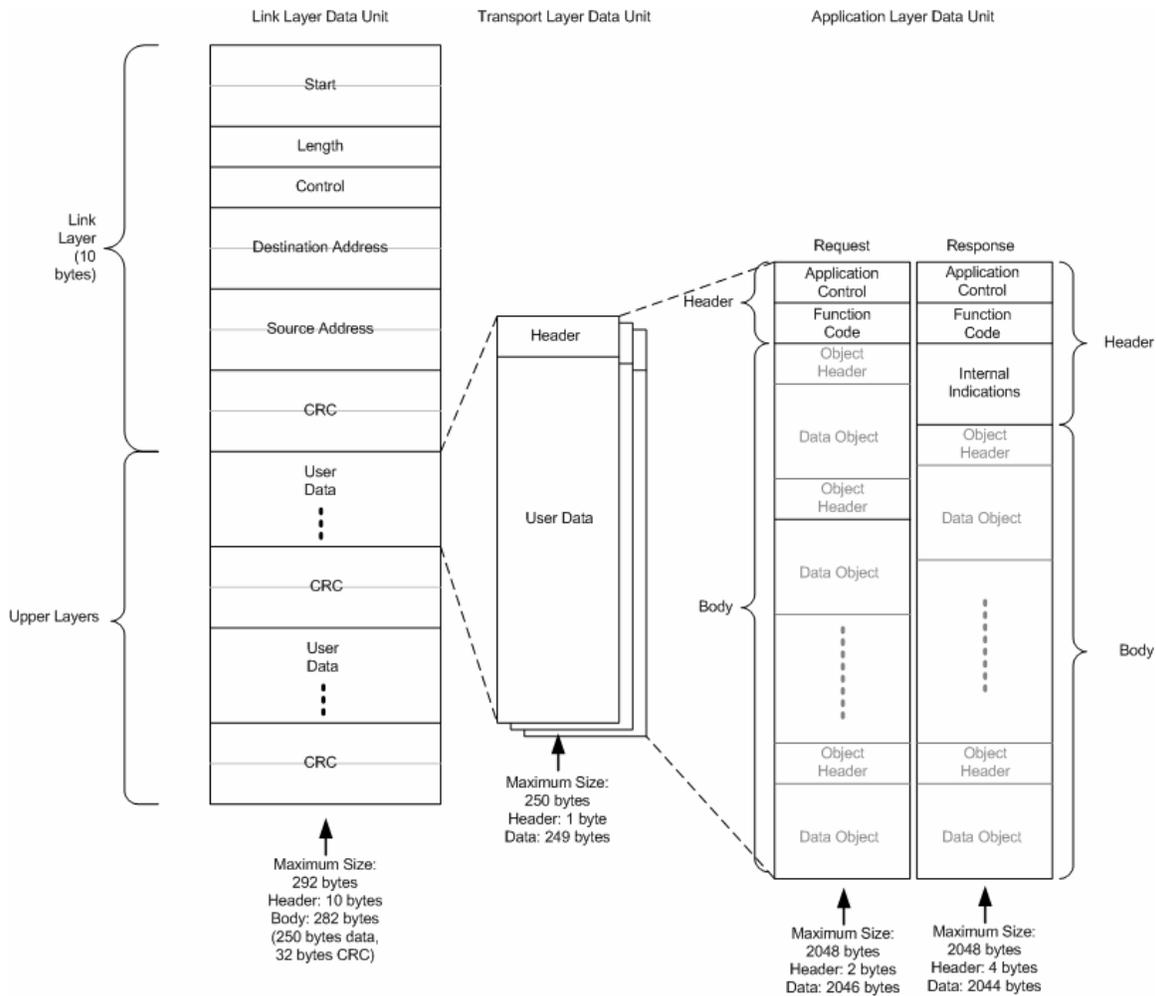


Figure 30 A simplified breakdown of the DNP3 message format, showing encapsulation and assembly across layers.

The data in the Application layer consists of one or more Data Units. There are a wide variety of Data Units available with header sizes ranging from three to eleven bytes. Clark and Reynders describe many of the message formats available. [26]

### 3. HDLC

High Level Data Link Control (HDLC) is a link layer protocol defined by the International Standards Organization (ISO) for point-to-point and multi-point links. It is a predecessor to the modern Ethernet protocol. HDLC operates either in unbalanced response mode (NRM) for environments where a master station initiates all transactions, and Asynchronous balanced mode (ABM) for environments where any node can initiate a transaction. The format of a HDLC frame is shown in Figure 31.

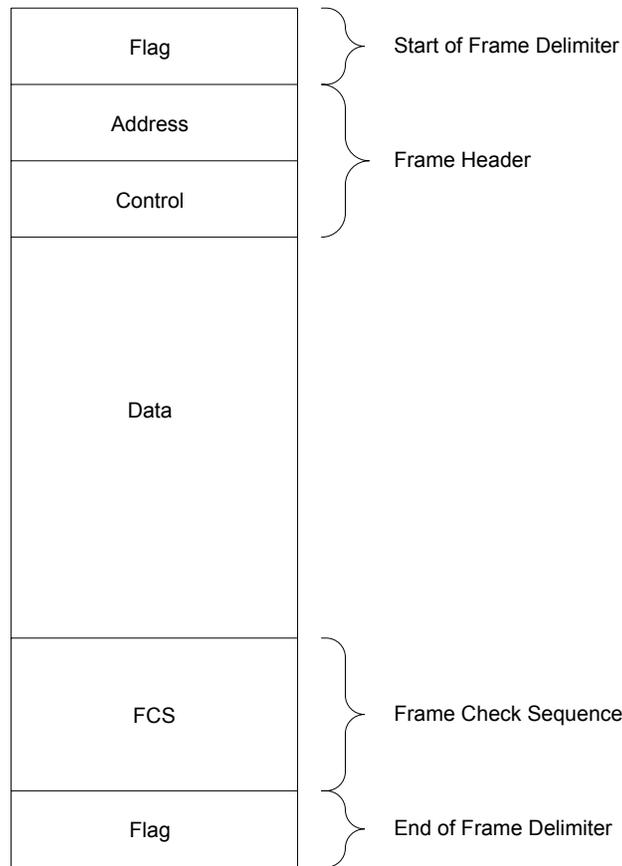


Figure 31 High Level Data Link Control Frame Structure. The Control field is used to determine which class of message is being used.

HDLC offers three classes of frames: unnumbered, information, and supervisory. Unnumbered frames are used for establishing the link or connection, and to define which mode (NRM or ABM) is used. Information frames are used to convey data from one node to another. Supervisory frames are used for flow and error control, and to provide acknowledgement or request re-transmission. Clark and Reynders describe the protocol operation and the error and flow control functions. [27]

#### 4. MODBUS

Modbus is a slow, yet widely accepted, protocol for process control systems. It defines no interface, so almost any transmission path is suitable for the slow speeds with which Modbus operates. Modbus operates on the Master/Slave principle, allowing for up to 247 slaves per master. Only masters initiate transactions, and a broadcast capability is defined when no responses from the slaves are required. A timeout function ensures that if the master does not receive a response to the query, it can resend its query. Data can be

sent in either ASCII or hex formats, with the hex format (often called Modbus-B for Binary) offering increased speed and reduced size of messages. Modbus offers error checking via character framing, a parity check and a Cyclic Redundancy Check (CRC).

Modbus offers seven functions:

- Coil (digital bits that can be read and written to) control commands for reading and writing a single coil or group of coils
- Input (digital bits that can be read) control commands for reading input status of a group of inputs
- Register (16 bit integers that can be read and written to) control commands for reading and writing one or more holding registers
- Diagnostic tests and reports
- Program functions
- Polling control functions
- Reset [28]

The format of a Modbus frame is shown in Figure 32. . Clark and Reynders provide examples of addressing and implementation of Modbus. [28]

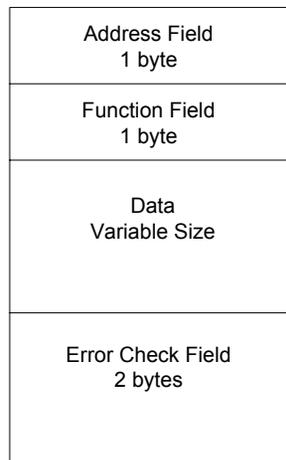


Figure 32 The Modbus frame format.

## 5. PROFIBUS

The Profibus protocol is based on a token bus/floating master system. The three types of Profibus are:

- Fieldbus Message Specification (FMS) is used for general data acquisition systems. At the physical layer, FMS requires the EIA-485 voltage standard for transmission speeds up to 187.5 Kbps..
- Decentralized Peripherals (DP) for use when fast communication is required. At the physical layer, DP requires the EIA-485 voltage standard for transmission speeds up to 12 Mbps.
- Process Automation (PA) for use when highly reliable and safe communication is required. At the physical layer, PA requires the IEC 1158-2 voltage standard for transmission speeds up to 31.25 Kbps. [29]

The significant difference between FMS, DP and PA is at the physical layer, which allows all three versions of Profibus to be employed in one SCADA system due to the similarities at the application layer. One application, for example, would be to use cheaper FMS devices in most field instrumentation devices, DP devices where speed is important, and PA devices where safety and reliability is required.

At the data link layer, Profibus employs a hybrid token passing and master/slave operation. A token is passed to all masters in a precisely defined time period and in a defined sequence, which allows the master with the token to read and write to slave devices. [29]

## **6. FOUNDATION FIELDBUS**

Foundation Fieldbus uses IEC 1158-2 and ISA S50.02-1992 standards at the physical layer which allows for intrinsic safety options, bus power, and a standardized interface to the communications link (Physical layer as described in Figure 33).

Foundation Fieldbus supports communications at 31.25 Kbps.

Figure 33 shows the message formats of Foundation Fieldbus for each the layers defined in the standard. The Data Link layer provides a scheduler called the Link Active Scheduler (LAS) that assigns permissions to devices for communicating based on a pre-defined schedule; no device communicates without permission from the scheduler. The LAS employs a "publisher/subscriber" model: when a device is permitted to transmit, it broadcasts its message on the bus (publisher), and the recipient device reads the message into memory (subscriber); all other devices ignore the message. Details regarding the

responsibilities of the various layers and sublayers of the specification are found in Reference [30].

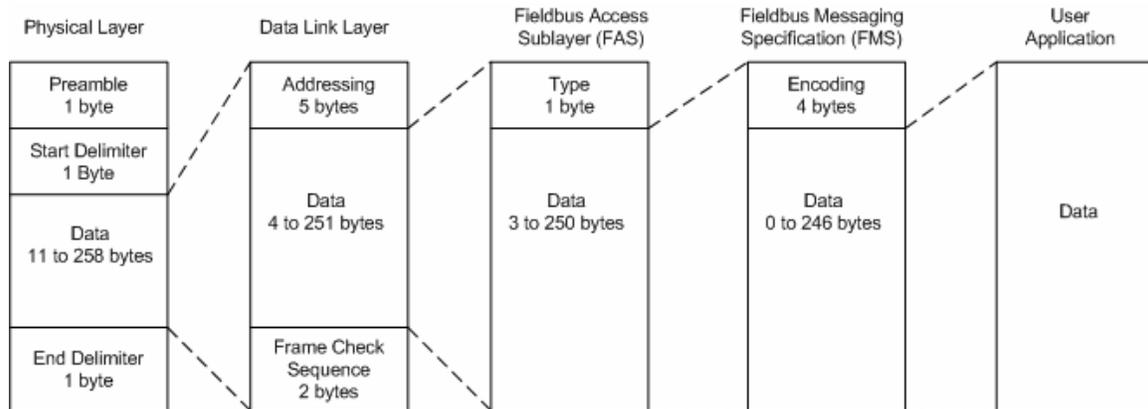


Figure 33 The Foundation Fieldbus message format as it travels up the stack from the physical to the application layer. (After Ref [30])

## 7. UCA

UCA is a relatively new standard employing an object-oriented approach with three primary goals:

- A uniform communications infrastructure
- A uniform application interface
- A uniform data model [31]

The three goals serve to provide vendor and platform independence, allowing disparate SCADA systems to integrate seamlessly.

The uniform communications interface is based on three defined layers called the L, T and A-Profiles, corresponding roughly to the Physical/Data Link, Network/Transport, and Session/Presentation/Application layers, respectively, of the OSI model. The modularity allows for almost any transmission medium and almost any network and transport standard. [31]

The uniform application interface is designed to allow manufacturers to design their devices using standard services models, in theory allowing any UCA-compliant device to be connected to any other UCA compliant device. It defines components (with

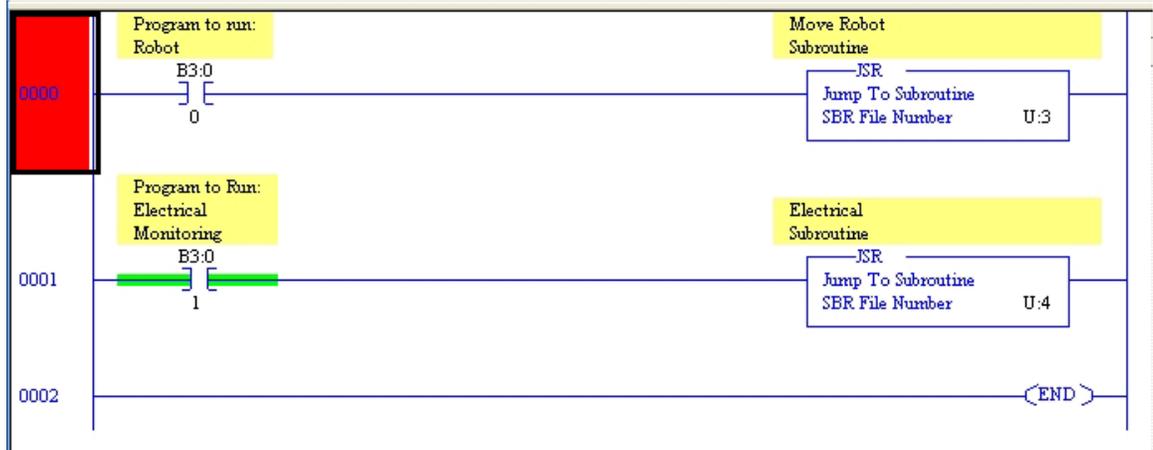
appropriate interfaces) such as a Data Acquisition and Control (DAC) Client, DAC Server, and Data Repository. [31]

The uniform data model defines an object model, through four levels of abstraction, which serves to standardize the approach to handling SCADA data. Each level in the abstraction provides attributes that are inherited by lower levels. [31]

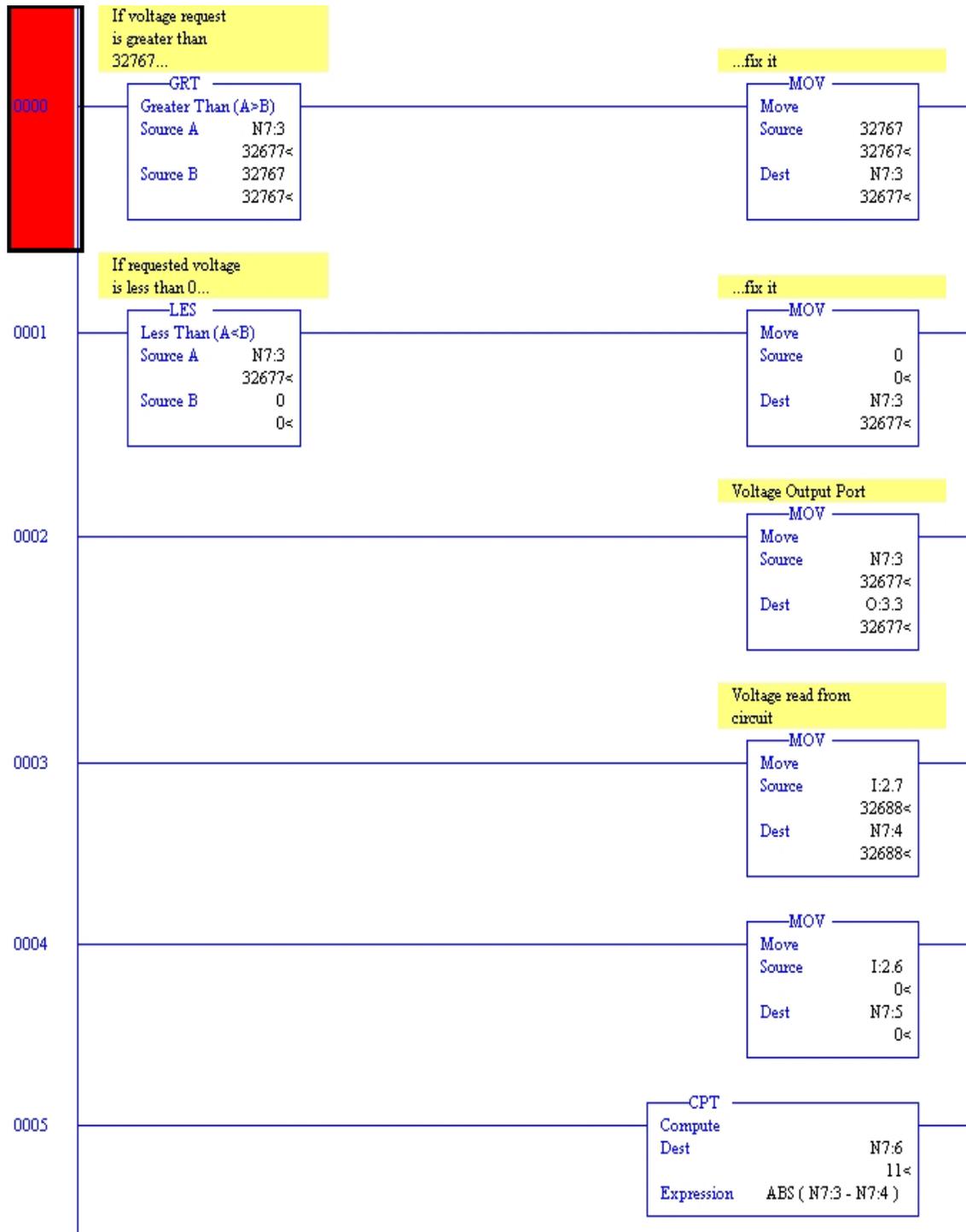
THIS PAGE INTENTIONALLY LEFT BLANK

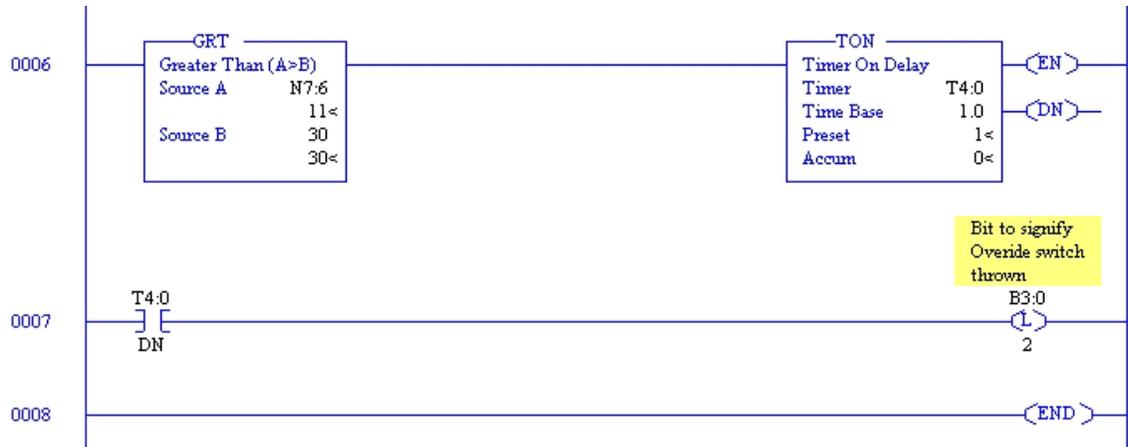
# APPENDIX B - LADDER LOGIC FOR SCADA TECHNOLOGY TESTING LABORATORY DEMONSTRATION MODEL

## 1. LADDER LOGIC - PROGRAM ENTRY

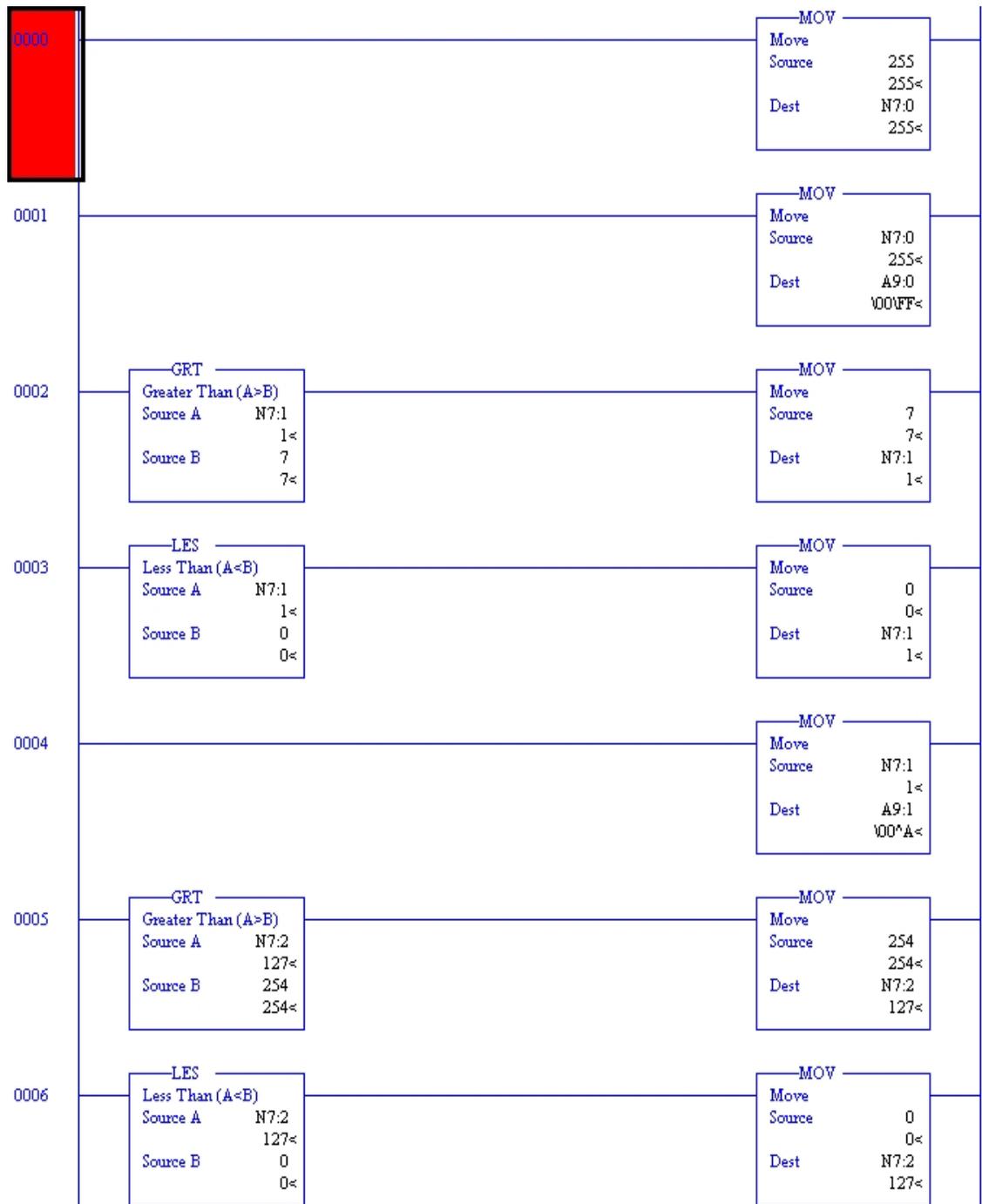


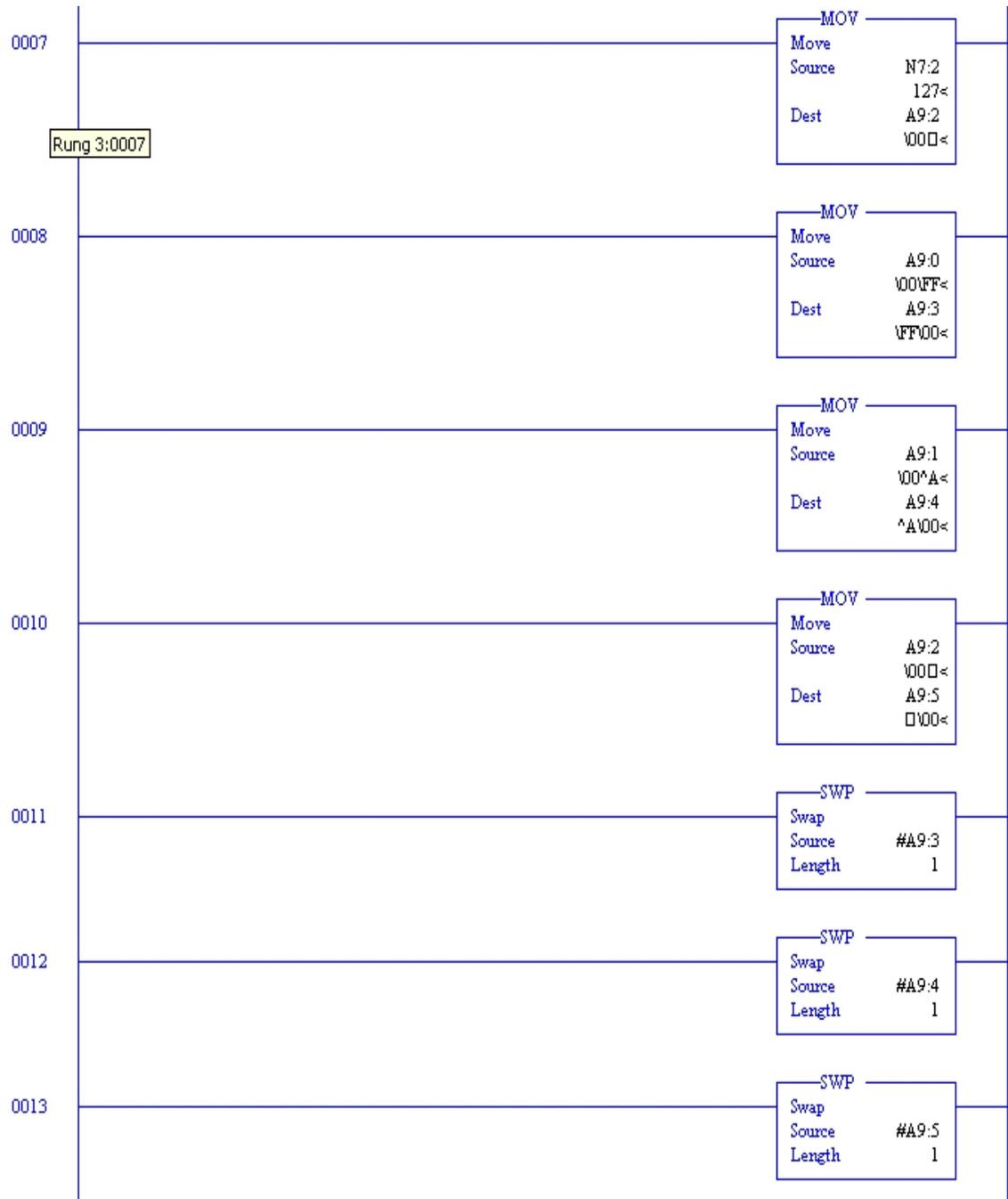
## 2. LADDER LOGIC - SIMPLE ELECTRICAL LAB

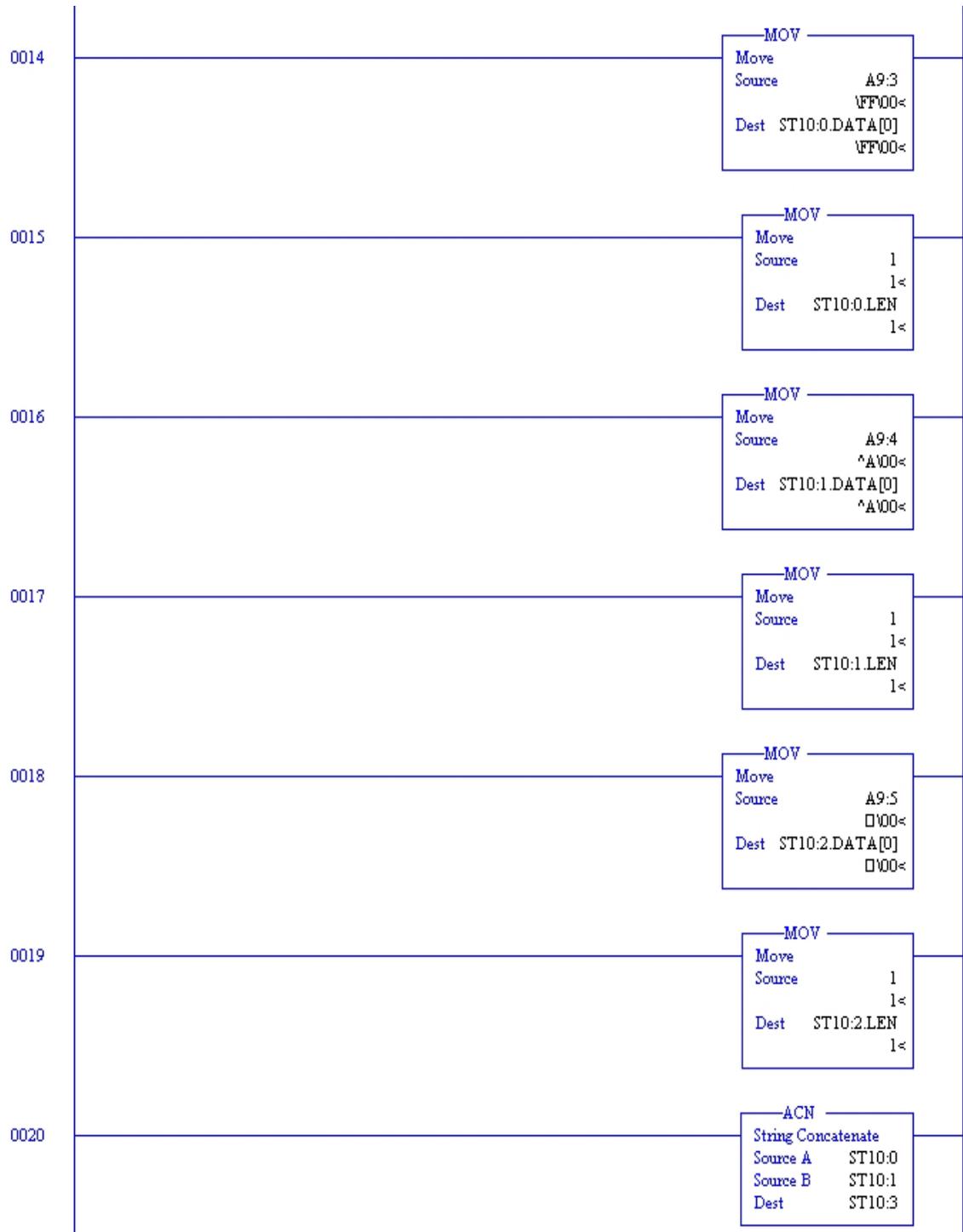


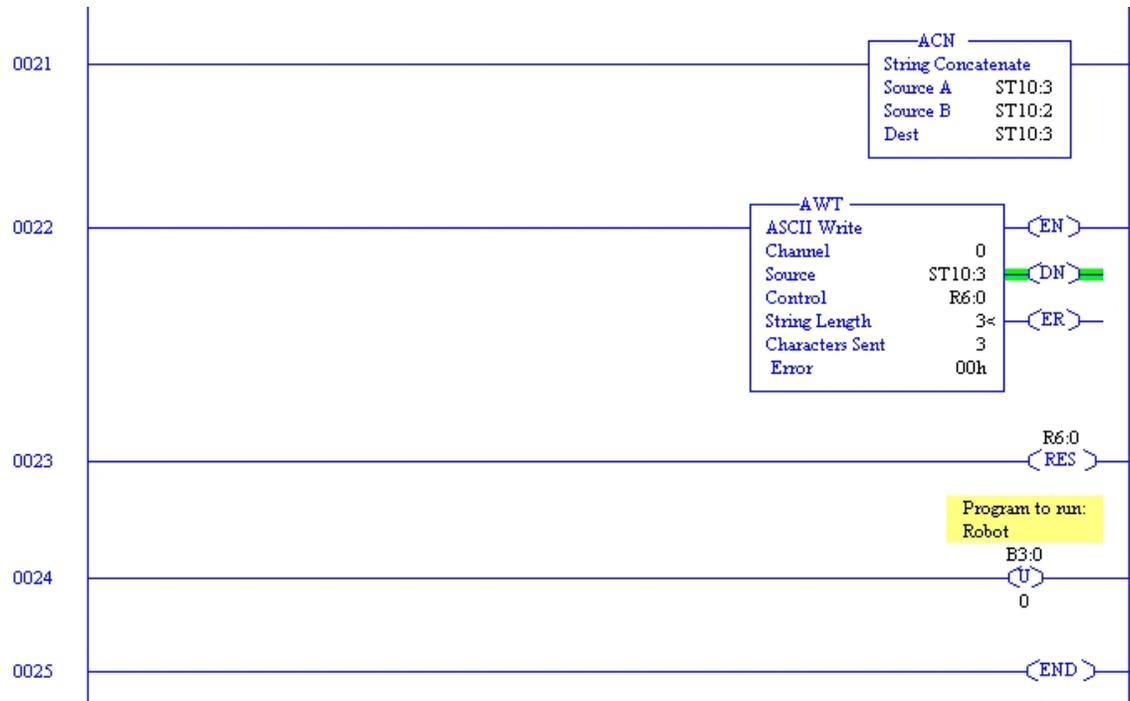


### 3. LADDER LOGIC - SIMPLE MECHANICAL LAB









THIS PAGE INTENTIONALLY LEFT BLANK

## APPENDIX C - ROBOTIC ARM MOVEMENT COMMANDS

To command the robot arm to move, a series of three bytes in the following format is sent to the robot arm through the SLC-5/05 RS-232 port:

- Byte 1: sync marker (always 255)
- Byte 2: Servo Number (between 0 and 254) In the NPS SCADA Lab, there are only 5 servos, so the appropriate range is 0-4:
  - Servo 0 controls the base
  - Servo 1 controls the shoulder
  - Servo 2 controls the elbow
  - Servo 3 controls the wrist
  - Servo 4 controls the pincers.
- Byte 3: Position (between 0 and 254. Position 0 is far left/up, position 254 is far right/down, and position 127 is center.)

The following sections describe the commands sent to the robot arm to perform the following sequence of actions:

1. Reset the arm to the center
2. Move from the center to the section indicated by "FROM"
3. Grasp the barrel
4. Move to the section indicated by "TO"
5. Release the barrel
6. Reposition at center

The filename is the file read into the RSView32 Works application's memory at the time the appropriate button is pressed on the operator's console interface. The RSView32 Works application then transmits the contents of the file, one line at a time, to the SLC-5/05 for further processing and transmission to the robotic arm.

The first byte, since it always 255, is omitted from the text file and is automatically added by the SLC-5/05 ladder logic program. The second byte (servo) is represented by the number in front of the comma, and the third byte (position) is represented by the number after the comma.

### 1. FROM: A TO: C

The file "Box\_A\_C\_01.txt" provides the commands for the robot arm to move the barrel of simulated hazardous material from position A on the housing to position B.

0,127  
1,127  
2,127  
3,127  
4,127  
0,220  
4,208  
2,150  
1,115  
1,115  
1,115  
4,65  
4,65  
4,65  
1,140  
0,70  
2,165  
1,135  
1,135  
1,135  
4,208  
4,208  
4,208  
2,127  
0,127  
1,127  
4,127

**2. FROM: B TO: A**

The file "Box\_B\_A\_01.txt" provides the commands for the robot arm to move the barrel of simulated hazardous material from position B on the housing to position A.

0,127  
1,127  
2,127  
3,127  
4,127  
0,146  
1,175  
4,208  
2,190  
2,190  
1,154  
1,154

1,154  
4,65  
4,65  
4,65  
2,150  
0,220  
1,120  
1,120  
1,120  
4,208  
4,208  
4,208  
2,127  
0,127  
1,127  
4,127

**3. FROM: C TO: B**

The file "Box\_C\_B\_01.txt" provides the commands for the robot arm to move the barrel of simulated hazardous material from position C on the housing to position B.

0,127  
1,127  
2,127  
3,127  
4,127  
0,75  
0,75  
0,75  
4,208  
2,160  
2,160  
2,160  
4,65  
4,65  
1,175  
0,146  
1,160  
2,180  
2,180  
2,180  
4,208  
4,208

2,127  
0,127  
1,127  
4,127

## INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center  
Ft. Belvoir, Virginia
2. Dudley Knox Library  
Naval Postgraduate School  
Monterey, California
3. Susan Alexander  
National Security Agency  
Fort Meade, MD
4. George Bieber  
OSD  
Washington, DC
5. RADM Joseph Burns  
Fort George Meade, MD
6. Deborah Cooper  
DC Associates, LLC  
Roslyn, VA
7. CDR Daniel L. Currie  
PMW 161  
San Diego, CA
8. LCDR James Downey  
NAVSEA  
Washington, DC
9. Dr. Diana Gant  
National Science Foundation
10. Richard Hale  
DISA  
Falls Church, VA
11. LCDR Scott D. Heller  
SPAWAR  
San Diego, CA

12. Wiley Jones  
OSD  
Washington, DC
13. Russell Jones  
N641  
Arlington, VA
14. David Ladd  
Microsoft Corporation  
Redmond, WA
15. Dr. Carl Landwehr  
National Science Foundation  
Arlington, VA
16. Steve LaFountain  
NSA  
Fort Meade, MD
17. Dr. Greg Larson  
IDA  
Alexandria, VA
18. Penny Lehtola  
NSA  
Fort Meade, MD
19. Ernest Lucier  
Federal Aviation Administration  
Washington, DC  
Ernest.Lucier@faa.gov
20. CAPT Sheila McCoy  
Headquarters U.S. Navy  
Arlington, VA
21. Dr. Vic Maconachy  
NSA  
Fort Meade, MD
22. Doug Maughan  
Department of Homeland Security  
Washington, DC

23. Dr. John Monastra  
Aerospace Corporation  
Chantilly, VA
24. John Mildner  
SPAWAR  
Charleston, SC
25. Keith Schwalm  
Good Harbor Consulting, LLC  
Washington, DC
26. Dr. Ralph Wachter  
ONR  
Arlington, VA
27. David Wennergren  
DONCIO  
Arlington, VA
28. Colleen Herrmann  
DONCIO  
Arlington, VA
29. David Wirth  
N641  
Arlington, VA
30. Daniel Wolf  
NSA  
Fort Meade, MD
31. CAPT Robert Zellmann  
CNO Staff N614  
Arlington, VA
32. Charley Robinson  
Instrumentation, Systems, and Automation Society (ISA)  
Research Triangle Park, NC
33. Marine Corps Representative  
Naval Postgraduate School  
Monterey, California

34. Director, Training and Education  
MCCDC, Code C46  
Quantico, Virginia
35. Director, Marine Corps Research Center  
MCCDC, Code C40RC  
Quantico, Virginia
36. Marine Corps Tactical Systems Support Activity (Attn: Operations Officer)  
Camp Pendleton, California
37. Dr. Cynthia E. Irvine  
Naval Postgraduate School  
Monterey, CA
38. Deborah S. Shifflett  
Naval Postgraduate School  
Monterey, CA
39. Daniel F. Warren  
Naval Postgraduate School  
Monterey, CA