CHE 461
Process Dynamics and Control
Laboratory Manual

D.E. Rivera, Associate Professor
V.E. Sater, Professor
Department of Chemical and Materials Engineering
Arizona State University

with the assistance of
M.W. Braun
M.E. Flores
K.S. Jun, Ph.D.
M.K. Shetty
L. Yang

Copyright ©1993-2001 ASU-Control Systems Engineering Laboratory

August 25, 2001
# Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Preface</td>
<td>1</td>
</tr>
<tr>
<td>Safety Manual</td>
<td>2</td>
</tr>
<tr>
<td>Experiment A: TotalPlant Solution System Overview</td>
<td>Lab A, Page 1</td>
</tr>
<tr>
<td>Experiment B: A/B Mixing Reactor Operation</td>
<td>Lab B, Page 1</td>
</tr>
<tr>
<td>Experiment C: First-Order Systems Response Analysis</td>
<td>Lab C, Page 1</td>
</tr>
<tr>
<td>Experiment D: Gasoil Furnace System Identification</td>
<td>Lab D, Page 1</td>
</tr>
<tr>
<td>Experiment E: Mixing Tank Dynamic Modeling</td>
<td>Lab E, Page 1</td>
</tr>
<tr>
<td>Experiment F: Heat Exchanger Instrumentation</td>
<td>Lab F, Page 1</td>
</tr>
<tr>
<td>Experiment G: Heat Exchanger Control</td>
<td>Lab G, Page 1</td>
</tr>
<tr>
<td>Experiment H: Mixing Tank Conductivity Control</td>
<td>Lab H, Page 1</td>
</tr>
<tr>
<td>Experiment I: A/B Mixing Reactor Control</td>
<td>Lab I, Page 1</td>
</tr>
<tr>
<td>Experiment J: Furnace Custom Control Strategy</td>
<td>Lab J, Page 1</td>
</tr>
<tr>
<td>Experiment K: pH Reactor Control</td>
<td>Lab K, Page 1</td>
</tr>
<tr>
<td>Appendix I: Picture Editor Reference</td>
<td>Appendix I, Page 1</td>
</tr>
</tbody>
</table>
Preface

Chemical engineering students at Arizona State University are indeed privileged to learn control engineering principles in a unique, industrial-scale testbed environment. The presence of these remarkable facilities and their use in undergraduate instruction is no accident. It is an integral part of the mission of the Control Systems Engineering Laboratory, which began in September, 1990 as the result of a major collaboration between Honeywell Industrial Automation and Control, Digital Equipment Corp., and Arizona State University. Honeywell’s initial donation of a TDC 3000 Plant Information and Control System (market valued at $1.2 million) has been progressively updated over time, culminating in the migration to a Honeywell TotalPlant Solution system in early 2000. The result is a unique, state-of-the-art real-time computing environment which allows students in the laboratory to stage a wide variety of industrially-meaningful scenarios spanning the areas of batch and continuous process control.

In the years since the inauguration of the laboratory, we have been busy interfacing some of the pilot-scale equipment in our Unit Operations and Process Control Laboratories into the platform, as well as developing a number of experiments using simulated process units which display features difficult to recreate in a lab environment. The results of our efforts is summarized in this lab manual. It is our sincere hope that these experiments will help you, the student, have a deeper understanding of both fundamental and practical issues in control engineering, and the important role that process control plays towards improving operations, safety, and environmental compliance in process plants.

During the summer of 2001 we were pleased to receive a donation of a PI system from OSI Software, Inc. of San Leandro, California, to augment the functionality of the real-time system available in our laboratory. We are grateful to Dr. Pat Kennedy, President of OSI, for making this donation possible.

Last but not least, we would like to thank the following Honeywell managers and engineers (present and past) who have played a major role in the development of the Control Systems Engineering Laboratory: Rod Woods, Reed Baron, Jim McCarthy, Ed Massey, Russ Henzel, Jim Nichols, Drew Shore, Don Clark, Rob Segers, and Perry Fanzo. We look forward to continued collaboration with Honeywell in the future.

D.E. Rivera
V.E. Sater
SAFETY MANUAL

Undergraduate Laboratories
Bldg SCOB, Rms 190, 191, and 192

EMERGENCY PHONE NUMBERS

• Tempe Medical, Police, Fire.............911
• Poison Control.........................253-3334

UNIVERSITY SERVICES

• Fire Marshall.............................5-1822
• Campus Police.........................5-3456
• Student Health Service..............5-3346
Introduction

Safety in the undergraduate laboratory is everyone’s responsibility. Experiments conducted in a safe and conscientious manner will protect you and your laboratory partners. The protection of health and safety is a moral commitment. Certain practices are required by law. This manual has been prepared for your own protection in accordance with federal, state and local regulations. You are responsible for compliance with all safety regulations set forth in this manual.

General Safety Principles and Regulations

1. Safety Preparation: Know the safety procedures that apply to the experiment being performed by you and your group. Study any start-up and shut-down procedures stated in the laboratory write-ups, and inspect the equipment to be used. Determine any potential hazards, and take the appropriate precautions before beginning any work. Be sure that you know the locations of the nearest fire alarm, telephone, and building exit.

2. Eye Protection: Arizona State Law requires that eye protection be worn by all individuals within the vicinity of caustic solutions, explosive materials, heat treatment of metals or other materials, and operations involving small flying particles. In addition, eye protection is required on all experiments performed within the Process Control (SCOB 192) and Unit Operations (SCOB 190) laboratories. These include the mixing tank (192), first-order systems (192), and heat exchanger (190) experiments.

   Safety glasses with side shields are preferred. Contact lenses should not be worn when performing the experiments listed above. Standard plastic goggles can be purchased in the ASU bookstore, and are available in most hardware stores.

3. Protective Clothing: For your personal protection, long pants must be worn in the laboratory, particularly around the first-order systems, heat exchanger, and mixing tank experiments. Shirts must cover the shoulders and midriffs - no tank tops or halter tops allowed.

   Open-toed shoes (e.g., sandals) or bare feet are strictly prohibited in all laboratory areas. Unprotected feet are subject to injury from falling objects, caustic solution spills, or exposure to hot condensate and steam discharges. Shoes with uppers made of solid material with non-skid soles should be worn during the laboratory period.

   When mixing caustic solutions, be sure to wear protective rubber gloves (obtain from the laboratory TA or faculty in charge). Insulating gloves are also available for handling hot materials or operating steam valves.
4. **Food and Beverages**: Contamination of food and drinks is a potential route for exposure to toxic substances. No food is to be stored or consumed in the laboratory. This also applies to coffee and beverages. Laboratory glassware and utensils should never be used to prepare food or beverages. Smoking is strictly prohibited in all laboratory areas.

5. **Housekeeping**: Spilled materials must be cleaned promptly with paper towels, rags or mop. Liquid spills are liable to consist of caustic solutions, dilute salt solutions, hot water and tap water. Report all spills, floods, and observed leaks to the lab TA or faculty. Exercise caution if spills occur in your area.

Do not block pathways with personal items such as books and backpacks. Bicycles are prohibited from the laboratory. Be sure that your work area has a clear path to exits and safety equipment.

Acid and base reactants used in CHE 461 are stored under the sink in SCOB 192. Be sure the containers are clearly labeled, and observe any health hazards posted prior to handling.

Inspect electrical connections prior to performing and experiment. All VAC power cords should contain a grounding wire (indicated by a three-prong plug). Avoid crossing walk ways and work areas with electrical cords. If connections appear unsafe, report these conditions to the laboratory instructor.

6. **Emergency Procedures**: Familiarize yourself with the emergency equipment in your area. Be sure you know how to operate the fire extinguishers, eye wash station, and shower station. Report the use of any fire extinguishers to the instructor.

Never work alone on any given experiment. If one or more of your laboratory partners is absent, notify the lab TA or course instructor.

All injuries (and near misses), no matter how slight should be reported to the course instructor. Burns and contact with caustic solutions should be flushed immediately with cold water. Injuries should be attended to at University Health Services, SHS Building (located near Palm Walk, across from the Physical Sciences complex).
TotalPlant Solution (TPS) System Overview

Objective

The TotalPlant Solution (TPS) system (formerly known as the TDC3000) is a state-of-the-art plant information and control system manufactured by Honeywell Industrial Automation and Control right here in the Phoenix metropolitan area. The majority of the lab experiments in ChE 461 revolve around the TPS platform. The purpose of this chapter is to familiarize you with some of its main components and the architecture of the system that is present on the ASU campus.

Procedure

At the start of the lab prep session, you will be required to watch a video that describes the basic architecture of the TPS system. A lecture based on the material in Experiments A and B (A/B Mixing Reactor Operation) will follow.

Lab Report

No lab report is required for this experiment. There will be a quiz at the beginning of lecture on Tuesday, September 4 which will cover the video, lecture and Lab Manual material for both Labs A and B. *Please note that students who fail to attend the lab prep session on Tuesday, August 28 will not be allowed to take the September 4 quiz.*

Important Note: The TPS system is a specialized computer platform with a proprietary operating system that is not always intuitive to operate. Please follow instructions carefully and refer to your TA whenever in doubt about something. If you accidentally press a key and can not recognize what is being displayed, do not improvise - ask your TA to guide you back to your working display.
Overview Of TPS Network

The TPS Network consists of the Local Control Network (LCN) and at least one process network. There are two possible process-connected networks, the Data Hiway and the Universal Control Network (UCN). The TPS system at ASU is a UCN-based system.

Each of the process-connected networks has an interface (gateway) that allows it to communicate with the LCN. The Data Hiway interface is called the Hiway Gateway (HG); the Universal Control Network interface is called the Network Interface Module (NIM).

The process networks (Data Hiway and Universal Control Network) transmit process information from data acquisition devices (for measuring temperature, flow, level etc.) through their gateways to the LCN. The visualization of the state of the plant is done by Universal Stations and Global User Stations, which reside on the LCN.

Modern installations of TPS networks include nodes that allow connections to Plant Control Networks (PCNs) based on ethernet or other standard protocols. These connections allow plant information and control decisions to be exchanged between the TPS system and computers carrying out higher-level system and entreprise-level functions, such as Enterprise Resource Planning (ERP).

Figure A.1 summarizes the architecture of the TPS system present at ASU. The TPS system in the undergraduate control laboratories is composed of the following nodes:

- 3 Process Managers (PM)
- 1 Network Interface Module (NIM)
- 6 Universal Stations (US)
- 1 History Module (HM)
- 2 Application Modules (AM)
- 3 Global User Stations (GUS)
- 1 Application Processing Platform (APP)

The sections that follow describe these nodes in relation to their TPS network connections. LCN-connected nodes are described first, followed by UCN-connected nodes. Last but not least, the more modern TPS nodes which have both LCN and PCN connections are presented.

LCN Nodes

Universal Station (US)

The Universal Station (US) is the primary TPS system human/machine interface. It provides a single window to the entire system, whether the data is resident in one of the LCN modules or in one of the process connected devices. The same workstation can be used to accomplish different tasks; it can be used by an operator, a process engineer, and by a maintenance technician to accomplish each of their different tasks.
Figure A.1: TotalPlant Solution System Architecture at ASU
History Module (HM)
The History Module (HM) provides mass storage of data on hard disk media. It is available with redundant WREN drives and allow storage and quick access to large blocks of data. Some examples of the types of data that can be stored and accessed are:

- History of process alarms
- Operator changes
- Operator messages
- System status changes
- System errors
- System maintenance recommendations
- Continuous process history to support logs and trends
- System files of all types, load images and other data required any time modules are loaded or personalities are changed
- Checkpoint data for maintaining up-to-date box and module settings in the event the device is taken out of service
- On-process maintenance information and analysis
- User-written application programs and custom data segments.

Application Module (AM)
The Application Module (AM) permits the implementation of more complex control calculations and strategies than are possible when using only process-connected devices. A set of standard advanced control algorithms is included. Custom algorithms and control strategies can be developed by using a process-engineer-oriented Control Language (CL/AM).

Network Interface Module (NIM)
The Network Interface Module (NIM) is a module on the LCN that interconnects the UCN with the LCN. It converts the transmission technique and protocol of the LCN to the transmission technique and protocol of the UCN. A NIM is almost always redundant (coupled with a second NIM) for enhanced security.
UCN Nodes

Process Manager (PM)

The Process Manager (PM) provides a complete range of data acquisition and control capabilities, including digital inputs and outputs, analog inputs and outputs, and up to 160 regulatory control loops. The number and types of control functions to be implemented, along with the PM processing rate, are configurable by the user. Custom control strategies can be developed by using a process engineer oriented control language (CL/PM). Peer-to-peer communications with other devices on the UCN is possible.

Figure F.1 shows that the PM consists of the Process Manager Module (PMM) and the Input/Output (I/O) subsystem. The PMM consists of

- Communication Processor and Modem: It provides high performance network communications such as network data access and peer-to-peer communications.
- I/O Link Interface Processor: It is the PMM interface to the I/O subsystem and provides high speed I/O access for communications and control functions.
- Control Processor: It executes regulatory, logic and sequence functions and includes a powerful user programming facility.

The PMM is partitioned into slots. A tagged slot is referred to as a data point in the TDC3000 system. There are eight types of data points configured into the PMM slots.

The I/O subsystem performs input and output scanning and processing on all field signals. The following 8 I/O processors are available for the PM:

- Analog Input - High Level (16 pts)
- Analog Input - Low Level (8 pts)
- Analog Input - Low Level Multiplexer (32 pts)
- Smart Transmitter Interface (16 pts)
- Analog Output (8 pts)
- Pulse Input (8 pts)
- Digital Input (32 pts)
- Digital Output (16 pts)

Each of these processors is briefly described below:

Analog Input. The main function of the analog input processors is signal conversion and conditioning of analog (i.e., continuous-time) signals. Typical examples of analog signals in a process system are temperature and concentration. The analog input processor carries out the following tasks:
Figure A.2: Outline of the Process Manager
TPS System Overview

- PV (Process Value) Source
- PV Clamping
- EU (Engineering Unit) Conversion
- PV value status
- PV filtering
- Software calibration

The low-level multiplexer processor provides an economical way to bring in a large number of data acquisition signals. The number of points handled by the multiplexer varies on the type of analog input.

Smart Transmitter Interface. The smart transmitter interface processor is the process manager’s interface to Honeywell’s advanced series of smart transmitters. It supports the functions for PV processing, EU conversion, and alarming supported by the other analog input processors. It also provides bad PV and bad database protection for added security.

Analog Output. Analog output processor achieves D/A (Digital-to-Analog) conversion in the TDC system. It provides the following functions:

- Readback check of actual output current.
- Output characterization.
- Output default action on failure.
- Modes and associated functions to support
  - Manual loader station
  - Direct Digital Control (DDC)
- Software calibration.

It also provides separate D/A converters and power regulator per channel for maximum output security.

Pulse Input. Precise control using high accuracy pulsing-type sensing devices is possible with the pulse input processor.

Digital Input. Digital signals are characterized by a finite number of states (e.g., ON/OFF states in a pump; OPEN/CLOSED states in a valve). The digital input processor provides the following functions

- Event counting (accumulation)
- Push button and status type inputs
• Time deadband on alarms for status inputs
• Input direct/reverse
• PV source selection
• State alarming for status inputs

*Digital Output.* It provides the following functions

• Output types
  – Latched
  – Pulsed
  – Pulse - Width modulated
• Output default action on failure
• Output readback checking

**Logic Manager (LM) - Not Present at ASU**

The Logic Manager (LM) is a process-connected controller that provides high-speed logic control that is typical of a programmable logic controller (PLC). Logic Managers are supported by TDC3000 systems running R300 software or later. Peer-to-Peer communications to other devices on the UCN is possible.
Modern TotalPlant System Nodes

The most recent additions to the TotalPlant System architecture are TPS nodes that contain connections to both the LCN and the PCN. These nodes consist of commercial Intel-based workstations running under the Windows NT operating system. An LCN co-processor (LCNP) printed circuit board allows the data connection between the TPS node and other LCN devices. The TPS nodes present at ASU that fit within this category are the following:

- Global User Station
- Application Processing Platform

Global User Station (GUS)

The Global User Station (GUS) is a modern human interface to the TPS system which enables operators and engineers to monitor and control the process. It consists of:

- **Native Window**: The Native Window provides all original TPS Network Universal Station operating and engineering displays in a window on the Global User Station.

- **Display Runtime**: The Display Runtime component executes GUS displays built by the Display Builder or translated from TPS Network schematics by the Display Translator.

- **SafeView**: SafeView is a window manager that allows a user to define where types of windows can appear, where they can be moved, and whether they can be resized or may overlap other windows. SafeView can be configured to ensure that critical windows are never hidden.

Application Processing Platform (APP)

The APP is a TPS node that mimics the function of the Application Module; however, since it resides in a PC-based workstation, the computing speed and flexibility of the APP node are much greater than those of a traditional AM. The APP node’s principal role is in integrating advanced control and information management applications.

References

Much of the material in this chapter has been adapted from the Users and Training manuals developed by Honeywell for the TPS system. General information on control implementation environments is available in Chapter 2 of Ogunnaike and Ray (“Introduction to Control System Implementation”).
SAMPLE QUIZ QUESTIONS

1. Name the two principal networks that comprise the TPS system present at the ASU campus.

2. Name the TPS system node(s) that

   a) provides graphic displays that can be used by process operators.
   b) connects the two networks that comprise the TDC3000.
   c) serves as the file server for the system.
   d) allows, within the security of the TPS system, the implementation of custom algorithms and control strategies more complex than those available from the PM.
   e) allows visualization of the system via the Native Window application.
   f) receives and sends analog signals to the process.
   g) provides logic control similar to that of a Programmable Logic Controller (PLC).
   h) saves user-written application programs and custom data segments.

3. Name the processor card in the Process Manager that accomplishes the following task:

   a) provides precise control using high-accuracy pulsing-type sensing devices.
   b) reads fields signals such as temperature and concentration
   c) enables communication with the NIM and other UCN nodes.
   d) implements regulatory control action.
   e) sends the instructions that allow a pump to be turned on or off.

4. What principal features of the GUS and APP nodes distinguish them from other LCN-connected devices (name at least two)?
Experiment B: A/B Mixing Reactor Operation

Objective

In this experiment, you are asked to operate a simulated stirred batch reactor (Figure B.1) using the TDC3000. The basic process involves the addition of two reactants (A and B), mixing and heating of the reactants, and removal of the final product through the drain of the reactor. Your objective is to perform the following sequence of operations on this reactor:

1. Starting with a reactor that is clean, empty, and ready, fill the reactor with 100 gallons of Solution A.
2. Add 50 gallons of Solution B.
3. Stir the contents for 60 seconds.
4. Heat the contents to 40 deg C.
5. Drain the contents.

You will accomplish these operations using both manual entry of the sequence commands and with the help of an automated sequence program.

Background

The reactor has three general areas of concern: the Feed System, the Heating System, and the Drain System (Figure B.2). These are described below:

*Feed System.* The two feeds to the reactor are Solution A and Solution B (Tanks A and B on the diagram). Each feed has an independent feed system that consists of a feed block valve, a flow indicator, and a flow totalizer. Each of the feed systems is used to meter in a predetermined amount of each solution.

*Heating System.* The reactor is a jacketed vessel, with steam supplied to the jacket to heat the reactants. Reactor temperature control is accomplished by a reactor temperature controller (TIC2190#) cascaded to a steam flow controller (FIC2190#). To prevent overheating of the reactor, an interlock has been implemented in the logic block...
Figure B.1: A/B Mixing Reactor Schematic
Figure B.2: A/B Mixing Reactor Points Diagram
<table>
<thead>
<tr>
<th>Tag Name</th>
<th>Tag Type</th>
<th>Tag Name</th>
<th>Tag Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>TIC2190#</td>
<td>RC-1</td>
<td>INGA90#</td>
<td>NU-1</td>
</tr>
<tr>
<td>FIC2190#</td>
<td>RC-2</td>
<td>INGB90#</td>
<td>NU-2</td>
</tr>
<tr>
<td>FY2190#</td>
<td>RP-1</td>
<td>STATE90#</td>
<td>FL-1</td>
</tr>
<tr>
<td>FY2290#</td>
<td>RP-2</td>
<td>FULMT90#</td>
<td>FL-2</td>
</tr>
<tr>
<td>FY2390#</td>
<td>RP-3</td>
<td>CLNDT90#</td>
<td>FL-3</td>
</tr>
<tr>
<td>LI2490#</td>
<td>RP-4</td>
<td>AGTIM90#</td>
<td>TM</td>
</tr>
<tr>
<td>FVL2190#</td>
<td>DC-1</td>
<td>SETNK90#</td>
<td>LB</td>
</tr>
<tr>
<td>FVL2290#</td>
<td>DC-2</td>
<td>REACT90#</td>
<td>PMS-2</td>
</tr>
<tr>
<td>DVL2390#</td>
<td>DC-3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AG2490#</td>
<td>DC-4</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

| DC       | Digital Composite |
| FL       | Flag              |
| NU       | Numeric           |
| RC       | Regulatory Control|
| RP       | Regulatory PV     |
| TM       | Timer             |
| PMS      | Process Module Slot|
| LB       | Logic Block       |

Table B.1: Points for A/B Reactor Control System; # refers to last digit of the partition number.

<table>
<thead>
<tr>
<th>Partition</th>
<th>Group Displays</th>
</tr>
</thead>
<tbody>
<tr>
<td>901</td>
<td>1–5</td>
</tr>
<tr>
<td>902</td>
<td>7–11</td>
</tr>
<tr>
<td>903</td>
<td>13–17</td>
</tr>
<tr>
<td>905</td>
<td>101–105</td>
</tr>
<tr>
<td>906</td>
<td>107–112</td>
</tr>
</tbody>
</table>

Table B.2: Associated Group Displays
Figure B.3: Group Display 1

Figure B.4: Group Display 2
Figure B.5: Group Display 3

Figure B.6: Group Display 4
Figure B.7: Group Display 5

Figure B.8: Group Display 5 indicating TIC2190# and FIC2190# trends
Figure B.9: First page detail display for TIC2190#.

Figure B.10: First page detail display for SETNK90#.
Figure B.11: Block Diagram Representation of the Temperature-Flow Cascade Control Structure for the A/B Reactor.
SETNK90# that forces the steam flow controller to manual mode and the output to a specified value (FIC2190#.SAFEOP) should the reactor temperature exceed a predetermined trip point.

Drain System. The drain system is similar to the feed system. It consists of a drain block valve, a flow indicator, and a totalizer. The function of the drain system is to remove the product from the reactor after the reaction has been completed. An interlock is included to prevent the drain block valve from being opened if either feed valve is open.

In Lab A you were introduced to the component devices which comprise the TPS system. In this lab, you will be introduced to system “points”, that is, the software entities in the TDC system which accomplish the operation of the feed, heating and drain systems. A collection of points working in a well-defined sequence defines a plant control system. A list of the points that control the operation of the the A/B reactor control system is found in Table B.1. The points are organized into five group displays; the first two are shown in Figures B.3 and B.4. Table B.2 lists the corresponding group numbers for each partition. All the points for this control system have been built on the Process Manager (PM). A brief description of each of these point classes follows:

Regulatory Control Point. Regulatory Control points provide both open-loop and closed-loop control of process values (PV) such as flow, level, temperature, etc. The regulatory control point uses a PV value provided by an analog input point on the Process Manager, compares it to the desired setpoint (SP), and uses a mathematical control law to translate the difference into a corresponding output (OP) which is fed to an analog output point in the PM. The most common type of control law is the PID (Proportional-Integral-Derivative) control law; understanding and tuning this equation will be a major focus of this course. Often times the OP of a controller will correspond to the SP of another; this arrangement is referred to as cascade control. In the A/B reactor control system, TIC2190# is a temperature control point which cascades into FIC2190#, the steam flow controller, as seen in Figures B.1, B.2, and B.3. A block-diagram representation of the closed-loop system is shown in Figure B.11.

Digital Composite Point. Digital composite points use digital input and digital output information from the PM to provide an interface to discrete devices such as motors, pumps, solenoid valves, and motor-operated valves. This point provides built-in structures for handling interlocks, and supports display of the interlock conditions in group, detail, and graphic displays. Examples are FVL21901, FVL22901, DVL23901, and AG24901 in Figure B.4.

Numeric Data Point. Numeric points store values that can be used for batch/recipe operations, or it can be used as a scratch pad to store the intermediate results of calculations. The values in a numeric point are real numbers that have been entered by the operator, or by a sequence program, or other system element. Numeric points in the control system include INGA90# and INGB90# (points which store how much of ingredient A and B needs to be charged in the reactor).
Regulatory PV Point. This point provides the flexibility to implement various types of PV calculations and compensation functions. Among the choices of selectable algorithms include mass flow, totalization, and variable dead-time compensation. In this lab, points FY2190#, FY2290#, and FY2390#, seen in Figure B.4, use the totalizer algorithm, which provide a *time-scaled accumulation* of the flow of A, B, and A/B mixture that is entering/leaving the system.

Timer Data Point. It allows the operator and the sequence program the ability to time process events, as required. It keeps track of the elapsed time after the timer has been started and provides an indication when the elapsed time has reached the predefined limit. Example: AGTIM901 in Figure B.4.

Flag Data Point. This is a 2-state point that is used for storing a Boolean value. The value can be supplied by the operator, by the sequence program, by an output connection from another PM point, by any PM box on the same UCN, or by a node on the LCN. Examples are FULMT901, and CLNDT901 in Figure B.4, which indicate the full/empty and clean/dirty status of the reactor, respectively.

Logic Block. To provide safety, ease of use for the operator, and to prevent waste, several *interlocks* are included that inhibit certain actions when conditions warrant. In this experiment, the interlocks are configured via a logic block (SETNK90#). In the feed system, an interlock has been added to the solution addition valves to prevent excess Solution A or Solution B from being added to the reactor. That is, when the solution totalizer point reaches its setpoint, the associated block valve automatically closes. Furthermore, neither solution feed block valve can be opened if the reactor level exceeds a specified trip point or if the reactor drain valve is open. Another important function of the logic block is to implement the shutdown of steam flow to the reactor jacket in the event that a high temperature trip point is exceeded.

Process Module Slot. Because manual operation of the reactor can be tedious (and error-prone), the PM is equipped with a process module slot point which runs a sequence program that automates the entire batch control system. A listing of the sequence program for Partition 901 is presented at the end of the write-up. The sequence program (ABMX290#) is written in Honeywell’s Control Language (CL), which combines features of languages such a PASCAL and FORTRAN. The sequence program is loaded in the PMS point REACT90#.

**Procedure**

Two lab sessions will be devoted to this experiment. Your job in the first lab session is to become familiarized with the components of the A/B reactor control system and learn how to operate them in both manual and fully automated fashion. In the second lab session, you will review what was accomplished in the first session and demonstrate your proficiency to the instructor by successfully completing the A/B Reactor Operation Competency Test.
Getting Familiarized

Reactor Schematic and Group Displays. Before you begin any work, take some time to look over your assigned schematic display and group displays. Hit the Group key and follow that with the number of the first assigned group number in your partition sheet. Your first assigned group is called “Set up Reactor Valves”. From this group display, use the Detail key to get detailed displays for each point; use the Trend key to get data histories for the flow and temperature controllers. Other groups can be similarly accessed via the Group key or by using the Disp Fwd and Disp Back keys.

Feed/Drain System. Here you want to familiarize yourself with the required steps for feeding components A and B and draining product from the reactor. You must set the correct target on the totalizer and start the flow. Note that just because the totalizer is running, it does not mean that contents are entering or leaving the system - you must open the corresponding valve using the appropriate digital composite point. The logic block SETNK90# takes care of shutting the flow automatically once the totalizer target is reached.

Do not reset a totalizer point once it has been started; you will lose your material balance if you do.

Heating System. Now that you have entered some contents in the reactor, proceed to warm them up. Here you want to get familiarized with the different controllers (temperature and flow) and their different modes: manual (MAN), automatic (AUTO), and cascade (CASC). Start first with the flow controller (FIC2190#) in manual mode; change the OP parameter and see what happens to the PV value of both TIC2190# and FIC2190#. Set FIC2190# to AUTO and then change the SP (setpoint) values. Then set FIC2190# to CASCade, and change the OP parameter on TIC2190#. Finally, set TIC2190# on AUTO and change the temperature setpoint. This last setting (AUTO for TIC2190#, CASC for FIC2190#) represents the desired settings for your controllers.

The detailed displays of TIC2190# and FIC2190# display both alarm and range limits for the controllers (Figure B.9). Familiarize yourself with what the limits on these points, and be able to explain to the instructor or your TA how these affect the group displays when running your reactor sequence.

Timer and Agitator. Familiarize yourself with the use of the agitator and timer. There is no interlock that links the timer with the agitator, so coordinate with your partners so that the agitator is turned off as soon as the timer has expired.

Logic Block. We have alluded to the logic block before, but take some time to learn how the logic slot is structured. To learn more about this point, examine Figure B.12 and get the detailed display for the point SETNK90# (Figure B.10). Look through all the pages, but in particular look for the high temperature trip point in the first page that causes the steam controller shutdown. Be able to explain the interlock logic to your TA and demonstrate how it works.
Figure B.12: Logic Block Algorithm - A/B Reactor. ### refers to the student partition number, 901-905. Please note that the schematic display LOGIC### can be used to visualize the actions of the logic block on your assigned partition.
Manual Operation Summary

There are five groups where your main points are displayed; these five groups follow the natural sequence of reactor operation. The steps you need to take to operate the system manually are:

**Group 1: Set up Reactor Valves (Figure B.3).** Make sure that all valves are closed; set the TIC2190# and FIC2190# to their proper modes.

**Group 2: Set up Accumulators and Flags (Figure B.4).** Set all totalizer points to RUNNING; do not start the timer point, however. Set the PV values in the FLAG points to FULL and DIRTY (the logic block point will indicate that the reactor is busy). If you wish, you may set the totalizer feed targets and timer setpoint in this display.

**Group 3: Load Solution A (Figure B.5).** Make sure that the totalizer setpoint for ingredient A is correctly set; open the valve to feed the ingredient. Please note that you do not have to wait until the tank has been charged with ingredient A in order to begin loading ingredient B (the next group display).

**Group 4: Load Solution B (Figure B.6).** As with solution A, make sure the correct totalizer setpoint for ingredient B is specified and open the corresponding valve.

**Group 5: Run Agitator and Drain (Figure B.7).** Turn the agitator on, then set the timer to RUNNING (make sure the correct setpoint has been specified first). Once the timer has expired, turn off the agitator. The temperature and flow controllers should already be in the appropriate modes; enter the desired setpoint and trend the SP and PV values for both the TIC2190# and FIC2190# points (as seen in (Figure B.8). Once the temperature reaches setpoint, proceed to drain the reactor contents (use LI2490#.PV value to correctly set the drain totalizer setpoint). As a final touch, return to Group Display 2 and set the FULL/EMPTY flag to “Empty”.

Make sure to gain familiarity with the range and limit alarms, particularly how these appear in the group displays and how they are configured in the detailed point displays (Figure B.9). Be able to define the temperature trip point for reactor shutdown from the detailed display for SETNK90# (Figure B.10).

Sequence Program Operation

At this stage you should be able to go manually through the entire sequence using group displays. A sequence program, however, can allow the operator to conduct all required operations almost automatically. ABMX290#, the Control Language (CL) program that we will use in this lab, executes from the “process module slot” point REACT90#. Call up this point from the reactor schematic, and show your TA that you can run the entire sequence from the process module display without resorting to the group displays. **Note:** The PM Control Language program used by the Process Module Slot can be found at the end of the Lab B description. During the TPS Competency Test, you will be asked to operate the sequence program using the Native Window in a Global User Station.
Sample Quiz Questions

1. What are the five main steps in the A/B Reactor Operating Sequence (be specific, and please distinguish these from the titles of the Group Displays)?

2. Name the PM point types that accomplish the following:

   a) provide both open-loop and closed-loop control of process values such as flow, level, and temperature.

   b) implement system interlocks that inhibit certain actions or cause overrides when conditions warrant.

   c) executes a sequence program which replaces tedious (and potentially error-prone) manual operation.

   d) provide flexibility for implementing various types of process value computations, such as the totalizer algorithm.

   e) gives the operator or sequence point the ability to clock process events, as required.

   f) a 2-state point type that stores a Boolean value.

   g) uses digital input and digital output information from the PM to regulate the behavior of discrete devices such as motors or pumps.

3. Examining Figures B.10 and B.12, what is the current value for the temperature trip point on the logic block point for the reactor?

4. Examine the PM Control Language code used to implement the automated batch reactor sequence. Answer the following questions:

   a) What is the maximum amount of ingredient A which can be fed into the reactor?

   b) Is the agitator operation coordinated with the timer in the sequence program? Does the sequence program allow the user to enter the agitator timer setpoint?

   c) What temperature difference must exist between the reactor contents and its setpoint before the sequence program begins to drain the tank contents?
A/B Reactor Operation Competency Test

As with Lab A, no report is required for this lab. At the end of the second session you will need to complete a competency test which demonstrates the ability of your group to carry out the tasks described in the Objective section. The competency test entails successfully completing the tasks listed below. Note: no written notes will be allowed during the competency test. Students will be assigned at random by the instructor to complete a portion of the assigned tasks. Please note that the score received will be a group grade.

Manual (Group Display) Operation: TA will start simulator sequence program

1. Set totalizers, controllers, and flags to their proper modes (5 pts).
2. Filled tank with soln A and B correctly (5 pts)
3. Turned agitator on and coordinated with timer (3 pts)
4. Set correct setpoint and heated contents. (5 pts)
5. Can create and rescale trends, and call detailed point displays (6 pts)
6. Can define and explain alarm and range limits in a regulatory control point (3 pts)
7. Drained contents and set flags to their proper final modes (3 pts)

Process Module Slot (Automatic) Operation: This part of the competency check must be done using a GUS station.

1. Properly started sequence program (4 pts).
2. Entered targets, acknowledged, and confirmed all messages correctly (9 pts).
3. Can relate/interpret sequence operation to its corresponding PM-CL code commands (2 pts).

Logic Block Operation:

The TA or instructor will provide you with a temperature trip point - please demonstrate the shutdown procedure (5 pts).
PM Sequence Program Listing

ABMX2901.CL
07/06/92 09:30:09

SEQUENCE ABMX2901 (PM; POINT REACT901)
EXTERNAL FIC21901,TIC21901,FY21901,FY22901,FY23901,LI24901,FVL21901,
& FVL22901,DVL23901,AG24901,STATE901,
& INGA901,INGB901,FULMT901,CLNDT901,AGTIM901,SIMLT901
-- ANY INTERNAL VARIABLE ASSIGNMENTS TO BE PLACED HERE
--
PHASE SETUP
STEP VALVE
SET FY21901.COMMAND,FY22901.COMMAND,FY23901.COMMAND=RESET
SET FVL21901.MODATTR,FVL22901.MODATTR,DVL23901.MODATTR,AG24901.MODATTR=
& PROGRAM
SET FVL21901.MODE,FVL22901.MODE,DVL23901.MODE,AG24901.MODE = MAN
STEP SHUT
CLOSED FVL21901,FVL22901,DVL23901
OFF AG24901
SET FL(001) = OFF
--
STEP FLOW
SET FIC21901.MODATTR=PROGRAM
STEP MODMAN
SET FIC21901.MODE= MAN
STEP OP
SET FIC21901.OP=0
SET FY21901.COMMAND,FY22901.COMMAND,FY23901.COMMAND=STOP
SET FY21901.COMMAND,FY22901.COMMAND,FY23901.COMMAND=RESET
INITIATE SIMLT901
--
PHASE STARTUP
STEP REDY
SET FL(001) = ON
SEND (WAIT):"READY THE REACTOR"
WAIT STATE901.PV=READY
--
STEP WARM
SET TIC21901.MODATTR=PROGRAM
SET TIC21901.MODE= AUTO
SET FIC21901.MODATTR=PROGRAM
SET FIC21901.MODE= CAS
--
STEP NEWSP
SET TIC21901.SP=25
WAIT ABS(TIC21901.PV-TIC21901.SP)<=2.0
--
PHASE LOADA
STEP AMT
SET INGA901.PV=0
SET FL(001) = ON
A: SEND (WAIT): "ENTER VALUE FOR INGA901 (RANGE= 0_200 GAL)"
IF INGA901.PV<=0.0 OR INGA901.PV>=200.1 THEN GOTO A
ENB HOLD VALVEA
--
STEP FILL
SET FY21901.AVT = INGA901.PV
SET FY21901.COMMAND=START
OPEN FVL21901 (WHEN ERROR INITIATE HOLD)
SET FULMT901.PV=FULL
SET CLNDT901.PV=DIRTY
WAIT FY21901.AVTFL
CLOSED FVL21901
SET FY21901.COMMAND=STOP
CALL EXCESS (FY21901.PVCALC,INGA901.PV,NN(80))
--
PHASE LOADB
STEP AMT
SET INGB901.PV=0
SET FL(001) = ON
A: SEND (WAIT): "ENTER VALUE FOR INGB901 (RANGE= 0_200 GAL)"
IF INGB901.PV<=0.0 OR INGB901.PV>=200.1 THEN GOTO A
ENB SHUTDOWN VALVEB
--
STEP FILL
SET FY22901.AVT = INGB901.PV
SET FY22901.COMMAND=START
OPEN FVL22901 (WHEN ERROR INITIATE SHUTDOWN)
WAIT FY22901.AVTFL
CLOSED FVL22901
SET FY22901.COMMAND=STOP
CALL EXCESS (FY22901.PVCALC,INGB901.PV,NN(79))
--
PHASE AGITATE
STEP AGT
SET AGTIM901.SP =60
SET AGTIM901.COMMAND=STOP
SET AGTIM901.COMMAND=RESTSTRT
SET FL(001) = ON
SEND :"MIXTIME WILL BE",AGTIM901.SP," SECONDS"
ON AG24901
WAIT AGTIM901.SO
OFF AG24901
--
PHASE HEATA
STEP SP
SET TIC21901.MODE= AUTO
STEP SP2
A: SET TIC21901.MODATTR=OPERATOR
SET FL(001) = ON
SEND (WAIT): "ENTER NEW TIC21901 SETPOINT (RANGE= 0_100 DEGC)"
WAIT TIC21901.SP<>25.0
IF TIC21901.SP< 25.0 THEN (set TIC21901.MODATTR=PROGRAM;
 & SET TIC21901.MODE= AUTO
& SET TIC21901.SP=25.0;GOTO A)
ENB EMERGENCY HEAT
STEP HEAT
SET TIC21901.MODATTR=PROGRAM
SET TIC21901.MODE= AUTO
SET TIC21901.PVHITP=TIC21901.SP+20.0
WAIT TIC21901.PV>TIC21901.SP-20.0
SET TIC21901.PVLOT=TIC21901.SP-20.0
WAIT ABS(TIC21901.PV-TIC21901.SP)<1.0
--
PHASE SETTLE
STEP SETTLING
WAIT ABS(TIC21901.PV-TIC21901.SP)<0.5
--
PHASE DRAIN
STEP DRN
SET FY23901.AVTv=LI24901.PV
SET FY23901.COMMAND=RESET
SET FY23901.COMMAND=START
OPEN DVL23901
WAIT FY23901.AVTvFL
CLOSED DVL23901
--
STEP FIN
SET FULMT901.PV=EMPTY
SET FY23901.COMMAND=STOP
SET FL(001) = ON
SEND :"BATCH COMPLETE"
--
STEP RSTMODTR
SET TIC21901.MODATTR,FIC21901.MODATTR,FVL21901.MODATTR,
& FVL22901.MODATTR,DVL23901.MODATTR,AG24901.MODATTR = OPERATOR
--
END ABMX2901
--
HOLD HANDLER VALVEA
CLOSED FVL21901
SEND (WAIT) :"TANK_A VALVE FAILED: REPAIR AND CONFIRM"
RESTART
SET FL(001) = ON
SEND :"PROGRAM RESTART AT PHASE LOADA"
RESUME PHASE LOADA
END VALVEA
--
SHUTDOWN HANDLER VALVEB
CLOSED FVL22901
SET TIC21901.MODATTR=PROGRAM
SET TIC21901.MODE= AUTO
SET TIC21901.SP=25
SEND (WAIT) :"TANK_B VALVE FAILED: REPAIR AND CONFIRM"
RESTART
SET FL(001) = ON
SEND :"BATCH RESUMES AT PHASE LOADB "
RESUME PHASE LOADB
END VALVEB

--

EMERGENCY HANDLER HEAT (WHEN TIC21901.PVHIFL)

SET FIC21901.MODE=MAN
SET FIC21901.OP=0.0
SET FIC21901.MODATTR=OPERATOR
SET FL(001) = ON
SEND :"BATCH RUINED: SEEK ALTRNT EMPLOYMNT"
SET FY23901.COMMAND=START
OPEN DVL23901
WAIT LI24901.PVP<=1.0
CLOSED DVL23901
END HEAT

--

SUBROUTINE EXCESS (ACTUAL,TARGET:IN;EXCESS:OUT)

SET EXCESS=ACTUAL-TARGET
SET FL(001) = ON
SEND :"EXCESS SOLUTION",EXCESS,"GALLONS"
END EXCESS
Experiment C: First-Order Systems Response Analysis

Objective

In this experiment, you and your lab teammates are asked to record and analyze the dynamic behavior of two sizes of unshielded industrial expansion thermometers. The gain and time constants for each thermometer must be computed from step response data generated under heating and cooling scenarios. Both the graphical inflection point technique and the fraction incomplete response method will be used for this purpose. You should be able to explain the experimentally-derived results on the basis of information gained from first-principles models.

Background

The required background material for this lab is covered in Chapters 4, 5, 13, and 15 of Ogunnaike and Ray. Chapter 9 of Seborg, Edgar, and Mellichamp and Chapter 3 of the first edition of Smith and Corripio’s Principles and Practice of Automatic Process Control (in reserve at Noble Library) may be helpful.

Procedure

Note: Safety in the laboratory is paramount. Please come dressed to lab in accordance with the safety dress code described in this manual; goggles or similar eye protection (with side shields) is required.

Two thermometers must be analyzed by your team. Make sure these are distinctly different thermometers in terms of size and weight. There are four different manufacturers that you can choose from (Weston, Reotemp, Ashcroft, Taylor).

Decide as a team the most effective way of recording time and temperature measurements from the thermometer’s response. Stop watches will be available for your group, but any reliable time recording device will do.
Make sure to first record the room temperature. Proceed to quench one of the unshielded industrial expansion thermometers in boiling water and measure its response by taking time-temperature readings with stop watches. Do not hold the thermometer directly over the boiling water before quenching as it will begin to heat up. Hold to the side and quickly place in the bath. The thermometer should respond in the familiar first order fashion. Record the response until it reaches steady state. *Note: Do not limit yourself to too few data points or you will be unable to accurately estimate gain and time constants from your measurements. Also, be sure to allow sufficient response time to see the system reach steady-state, since this will make it easier to estimate $y_\infty$.*

Once the thermometer reaches steady-state in the boiling water, remove it from the beaker and hold it stationary in the air. Record the cool down temperatures, keeping in mind the suggestions made previously. Repeat the heat-up/cool-down procedure with the second thermometer.

Calculate the gain and time constant from the time-temperature data using the “fraction incomplete time response” method discussed in lecture and summarized in Ogunnaike and Ray, Section 13.3. Keep in mind the results of EPA 4, Problem 2 as you carry out your analysis. Print out a plot of your data generated from any sensible computer tool (Matlab, Excel, etc.) and apply the graphical inflection point analysis technique of Ogunnaike and Ray Section 15.4.1 to obtain a second estimate of the gain and time constant values for this system. Remember to do this analysis for the two thermometers for both the heating and cooling scenarios.

**Lab Report**

Document your results using the format described in the CHE 461 Lab Syllabus. Since the execution of this experiment is relatively simple, a strong emphasis in grading will be placed on the quality of the experimental results. The instructor strongly recommends completing EPA 4, Problem 2 prior to lab time as a means for learning of some of the potential pitfalls in applying the fraction incomplete response method. You are also recommended to plot the data during your assigned lab period, and rework any experimental runs that may have inconsistent measurements or do not appear to give the expected results.

In your discussion, use a simple first-principles model (such as the one you were asked to derive for EPA No. 3, Problem 3) to explain your experimental observations and calculations. Be able to justify the similarities and differences in the estimated gain and time constant values on the basis of a first-principles understanding.

List and prioritize the potential sources of error that may have led your experimental results to not agree with those of a first-order system, or to discrepancies between the results obtained from graphical inflection point analysis and the fraction incomplete response technique. Furthermore, based on your experience, be sure to discuss the pros and cons of these two techniques for estimating first-order system parameters from data.
Experiment D: Gasoil Furnace System Identification

Objective

This lab is the first of two CHE 461 experiments focused on a simulated gasoil furnace. The overall objective is to introduce CHE 461 students to some of the diverse tasks that are performed by practicing engineers when building, commissioning, and testing a process control system. The primary goal of this experiment is to perform model building via system identification on the furnace. Both low noise and high drift disturbance conditions will be considered, as well as the use of step tests and more sophisticated PRBS inputs for identification testing. Lab J will explore closed-loop control issues on this furnace simulation in some detail.

Background

A heavy oil stream in a refinery must be preheated before being fed to a catalytic cracking unit. Your team has been assigned the job of completing the installation of a plant information and control system for the furnace unit. The immediate task that lies ahead of you is to conduct step and Pseudo-Random Binary Sequence (PRBS) tests on the plant to determine a model that will be used to build an advanced control strategy (Labs J1 and J2: Custom Control Strategy - Furnace Control).

Your team has been made painfully aware by the refinery’s Operations Superintendent that your plant is currently in a desirable normal operating mode. As such, you must conduct your step tests in a “plant-friendly” manner that causes minimal disruption of these operating conditions. Significant deviations from these will result in off-spec product in the cracking unit and loss of profits to your company.

Eventually, the plant suffers from substantial drift in the outlet gasoil temperature as a result of ambient and other unmeasured disturbances. Under these circumstances, “plant-friendly” step testing may no longer be an option - it will be necessary to conduct alternate identification tests that are non-“plant-hostile.” However, great care must be taken to avoid implementing identification tests that are substantially disruptive on the plant.
Figure D.1: Gasoil Furnace Schematic, with change zone display

Figure D.2: Group Display for Furnace Simulation
Procedure

Before commencing any work, take some time to look over the points and group display for your assigned furnace partition. *Take note of your assigned partition number, since you will also be working with it in Lab J.* The group display for this lab is composed of the following points:

*TI418##.* This is an analog input point whose PV value contains the current temperature measurement of the hot oil leaving the furnace. This is the controlled variable for this system.

*FI418##.* This flow indicator is an analog input point which informs the operator regarding the flowrate of oil entering the furnace.

*FIC419##.* A Process Manager (PM) regulatory control point which controls the flow of fuel gas to the furnace. The FIC419## setpoint is manipulated to achieve control of the outlet gasoil temperature per TI418##.

*ID.INP##* A custom point on the Application Module (AM) which generates a PRBS input. It will be needed to generate an identification data set under high-drift noise conditions.

*DACQ##* A second custom point on the AM which writes the values of FIC419##.SP, TI418##.PV and other system measurements to an ASCII text file.

## in the tagnames refers to the partition number for your team, which will be assigned to you by the instructor. A typical GUS-based schematic is shown in Figure D.1; a sample group display is shown in Figure D.2.

**GUS Schematic**

A real-time schematic of your assigned gasoil furnace partition has been built for your team using the Honeywell TPS Display Builder. A detailed, step-by-step guide to working with the Honeywell Display Builder is posted separately in the *Lab D Materials* folder on the ChE 461 myASU website. You may consider developing your own schematic as an “enrichment” activity once the main tasks of the experiment have been completed.

**System Identification**

This lab will require teams to perform the following tasks (nearly) simultaneously: identification testing and data recording on the TDC3000, model building using the System Identification Toolbox (SITB) in MATLAB, and model building using graphical methods. Since multiple identification tests will be performed in the course of this lab, it may make sense for team members to rotate between the jobs of attending to the TPS console, computing model parameters graphically from data, and regression-based analysis using the MATLAB m-file *pIDfurn*. In addition, team members will want to get acquainted with the M-file *inputdsc2001.m* provided by the instructor as a means for effectively designing a
“plant-friendly” PRBS test under noisy drift conditions. The team leader should coordinate tasks with lab team members to insure execution of the lab in a timely and efficient manner.

Keep in mind that your beloved instructor and his teaching assistants will serve the roles of Operations Superintendents. You can expect to hear from them if you create substantial disruptions on the plant.

Figure D.3: Typical Process Reaction Curve for a Furnace Using a “Double-Pulse” Input Sequence

Step Testing, Low Noise Conditions

First, make sure that the data collection program in DACQ## is activated and writing data to a clean data file in your user partition. The TA will guide you regarding the proper operation of the data collection point and the steps needed to transfer data from TDC3000 format to ASCII text (which can be easily loaded into Matlab, Excel, etc). The process should initially be operating close to its nominal operating conditions (300 deg C and 5 MSCF/HR). Generate data for system identification by introducing a step change in the fuel flow setpoint from your assigned group display. Start with a small positive step change (≈ 3% of scale for the fuel gas flowrate). NOTE: Please examine the detail display of the FIC419## point to determine the magnitude of the change based on instrument range, as determined from the values of PVEUHI and PVEULO. You will need to change the scales in the real-time trends on the TDC3000 rather often to obtain useful plots for graphical analysis. Once ‘steady-state” is attained, print out the response and have a team member use the inflection point method to estimate gain, deadtime, and time constant. Introduce a second step change in the fuel gas flowrate; this time, however, decrease the fuel gas flow rate by twice the amount as before. Wait until steady-state, print out the response, and estimate parameters graphically. Return the plant to the original nominal value for fuel flow setpoint (5 MSCF/HR). The overall input changes will correspond to a double pulse input as discussed in the lab prep lecture (Figure D.3). Graphical analysis should be performed on the third
step change and compared with the values from the previous two steps (which step change
gives most reliable values, and why?). This last step change will also provide data useful in
subsequent regression analysis for model validation purposes. Take the average of the graphi-
cal results from all three step tests and include a record of these values in your lab notebook.

Momentarily stop data collection and proceed to transfer the data file from the TDC3000
system to floppy. The data file corresponding to the low noise conditions should have the
name ident##a.xx, with ## representing your team’s partition. Start Matlab and load the
data for use by pIDfurn. This involves assigning data columns in ident##a.xx to variables
in the Matlab workspace. Please enter the following commands in the Matlab command
window (the 01 partition used as an example):

\[
\begin{align*}
> \text{cd a:} & \quad \% \text{Make sure that you are in the right directory!} \\
> \text{load ident01a.xx} \\
> u = \text{ident01a(:,3)}; \\
> y = \text{ident01a(:,4)}; \\
> T = 10; & \quad \% \text{This is the sampling period (in seconds).}
\end{align*}
\]

In this system FIC419##.SP corresponds to the manipulated variable \(u\), while TI418##.PV
is the output variable \(y\). The second column in ident##a.xx corresponds to the sampled
values for FIC419##.PV, while column five contains the sampled values for the furnace
feedflow FI418##. The first column is a time index which is not used by pIDfurn. Having
loaded the input, output, and sampling time into the Matlab workspace, your team should
be able to use pIDfurn to estimate model parameters; compare these with values estimated
from graphical (inflection point) analysis.

**Step Testing, High Noise Conditions**

Resume data collection with a clean data file (the new data collection file will be named
ident##b.xx). The TA and/or instructor will proceed to activate high noise conditions on
your furnace simulation. Allow 20 - 30 minutes before introducing new step changes (this
is time that can be well spent applying pIDfurn on the low noise data set you have just
collected, or learning how to use inputdsc2001.m, the PRBS design and training program).
Introduce a “double pulse” input using the same exact magnitudes as before. As with the
low noise case, generate plots of the responses and attempt graphical analysis of the data;
likewise, transfer the data file from the TDC3000 and use pIDfurn to obtain models from
regression analysis.

**PRBS Testing, High Noise Conditions**

The identification test design program inputdsc2001.m will allow your team to appropriately
design a Pseudo-Random Binary Sequence (PRBS) test based on a priori knowledge gained
from the model identified under low noise identification conditions. Please keep in mind the
following when executing the program:
• Please enter the gain in engineering units, and the deadtime and time constant in seconds (as displayed by pIDfurn).

• Select the PRBS input and “guidelines based on time constant information” options from the menus that initially appear in the program.

• A sensible estimate of the dominant time constant is obtained from the equation:

\[ \tau_{\text{dom}} \approx \tau + \frac{\theta}{2} \]

where \( \tau \) and \( \theta \) originate from the model obtained during low noise conditions.

• Good default values for \(\alpha\) and \(\beta\) are provided in the program (2 and 5, respectively).

• The input magnitude for the PRBS signal should be the same as that used in step testing. A move to a more aggressive yet non-“plant-hostile” magnitude may be possible, but requires instructor consent.

• Having many cycles of data is preferable, since it decreases variance in the parameter estimates; however, it increases the overall extent of identification testing (which is plant-hostile). Select the number of cycles to be implemented such that the overall test duration ranges between 8 to 10 hours.

Please record all the information resulting from the program, as this information will be needed when working with the input generation point on the TDC3000.

From your group display, call the detailed point display for ID_INP##, which will generate the PRBS input on the TDC3000. Go to the CUSTOM page for the point and make sure that the following parameters are correctly specified:

**START:** This parameter turns the input signal generation on and off.

**FN_FLOW:** The tag name for the fuel gas regulatory control point for your partition (FIC419##) should be present here.

**INPUT:** This should read PRBS.

**NMB_STG:** Enter the number of shift registers obtained from inputdsc2001.m.

**SWT_TIME:** Enter the switching time (as a number of samples) as computed by inputdsc2001.m. The parameter SMP_TIME is the index to a counter which keeps track of the sampling instants at which the PRBS algorithm must be executed.

**FLW_RATE:** Please make sure that this parameter is set at the nominal value for the fuel gas flowrate (5.0 MSCF/HR)

**PERCNT:** Specify the signal magnitude as a percent (in fractional terms) of the nominal value listed in FLW_RATE. The variation in the input signal is thus:

\[ \text{FLW RATE} \pm \text{FLW RATE} \times \text{PERCNT} \]
LEN_PRBS: This is the total length of the PRBS sequence in samples, as determined by:

\[
\frac{\text{Duration of One Cycle} \times \text{No. of Cycles}}{\text{Sampling Time}}
\]

COUNTER: This parameter keeps track of how many sampling instants have been implemented in the identification test. The initial value for this parameter (at the start of testing) should be 1; PRBS testing will stop once COUNTER reaches LEN_PRBS.

Resume data collection (again with a clean data file, this one to be named `ident##c.xx`) and initiate execution of the PRBS input. The nature of the PRBS input makes graphical analysis of the data very difficult; nonetheless, it is informative to visually examine how the signal excites both high-frequency (i.e., short time) and low-frequency (i.e., long time) dynamics of the plant. The test will run for multiple cycles, which will take a number of hours (as specified through LEN_PRBS). Once your team has ascertained that the PRBS input generation and data collection points are working properly, you are free to go. The instructor and the TA will assume responsibility for stopping data collection after the experiment has been completed. This final dataset will be posted on your team’s file exchange area in the ChE 461 myASU web page.

If your team carries out this experiment successfully, the sequence of step and PRBS tests should resemble what is shown in Figures D.4 and G.5.

When analyzing the PRBS data set using `pIDfurn`, please keep in mind that detrending operations will be needed in order to make it useful for ARX estimation. Specifically, you will need to use the differencing data preprocessing option to remove nonstationarity caused by drifting disturbances; the differencing operation (to the first degree) is described by the equation

\[
x'_t = \nabla x_t = x_t - x_{t-1}
\]

The differencing operation in `pIDfurn` is applied to both the input \( u \) and output \( y \) time series. (A theoretical justification of the importance of differencing, and why must it be applied to both the input and output data is presented in ChE 494/598 Introduction to System Identification, i.e., “the course in the spring.”). Please be sure to keep at least one cycle of the PRBS sequence for crossvalidation purposes.

Lab Report

The lab report should present in an organized, coherent manner the multitude of models that were estimated from the various furnace identification tests. Be sure to report your time constant and deadtime values in seconds and the gain in engineering units, since this will be needed later on in the control design labs (Lab J). Clearly, some experimental conditions were more conducive than others to proper modeling, and some model parameters appear to be more reliable than others. Contrast the different approaches to model building
in terms of handling the low noise and high noise conditions. What are the pros and cons of PRBS testing relative to pulse/step testing? Comment as well on the pros and cons of graphical versus regression-based empirical approaches to modeling building. Generate a list of potential sources of error for each approach, and classify these in order of importance. Determine the conditions under which each method is most reliable, and discuss the practical considerations associated with obtaining these conditions.

As you complete your lab report, keep in mind the following:

- It is possible to generate huge numbers of plots and data from pIDfurn. Keep the number of plots included in the body of your report to a reasonable number, while prudently using the appendix to include additional substantiating information. Use tables to effectively communicate the results of your analysis.

- Please recognize in your analysis that the underlying plant model between fuel flow setpoint and gasoil temperature remains the same throughout the experiment; all that changes in each case is the magnitude and character of the noise and the input signal used for identification.

- Carefully examining the ARX step response from pIDfurn will give you insights regarding the “true” model structure of the furnace simulation, and how valid is a first-order with deadtime approximation for such a system. Comment about this in your lab report.

- The presentation made during class on system identification is clearly your best reference on this material. A copy of the PRBS generating program (which is executed by ID_INP##) has been posted on the course web page.
Figure D.4: Typical 8 hr trend showing low noise pulse, high noise pulse, and PRBS tests. The PRBS test shows four cycles of the input signal.

Figure D.5: An 8 hr trend rescaled to show the low noise pulse, high noise pulse, and initial PRBS tests in more detail.
Experiment E: Mixing Tank Sensor Calibration and Dynamic Modeling

Objective

The objective of this experiment is to determine, via first principles modeling and system identification techniques, the dynamics relating inlet flowrate changes in the tank to changes in the outlet salt concentration of a continuous stirred mixing process. Computer-based and graphical modeling techniques will be examined as part of this experiment. The success of the modeling task hinges upon accurate measurement of salt concentration in the tank (among other factors). This will be accomplished by calibrating a conductivity sensor in the tank exit stream, specifically through the choice of the PVEUHI and PVEULO parameters found in the first page of the analog input point CI100.

Background

Figure E.1 shows both the process and the instrumentation used in this experiment. All points in this system (with the exception of the data collection point) reside in the Process Manager. Control of tap water into the process is accomplished by the regulatory control point FIC100. Water flow as measured by an orifice meter is recorded by the analog input point FI100, which provides the Process Value (PV) to the regulatory control point. The controller output signal is sent to the analog output point FM100, which adjusts the valve position on the water line. A Proportional-Integral (PI) control algorithm in FIC100 is tuned for smooth response to setpoint changes specified by the operator. A similar arrangement is used to adjust the flow of a concentrated salt solution into the tank via the regulatory control point FIC101.

The level in the tank is measured with a differential pressure cell (d/p) with one leg connected to the bottom of the tank and the other leg open to the atmosphere. The regulatory control point LIC100 compares this level with a desired level and manipulates the flow through the drain line.

The salt concentrations leaving and entering the tank are measured with conductivity cells and read into the system via the Process Manager analog input points CI100 and CI102, respectively. The correlation between the signal from the conductivity meters and
the concentration in grams/liter is accomplished by judicious specification of the range limits PVEUHI and PVEULO in the detail displays of CI100 and CI102. CIC100 is a regulatory control point used in Experiment H (Mixing Tank Control) which adjusts the salt inlet flowrate setpoint (FIC101.SP) to keep exit stream salt concentration under control (CI100.PV). Please leave this point on MANUAL throughout the experiment. The PV value of CIC100 will always be the same as that of CI100.

Group display 1 (Figure E.2) shows a summary of the important points needed for this experiment. The data collection point (DACQTANK) can be used to collect measurements of all process measurements into an ASCII file (IDENTTK.XX) at 10 second intervals. This file can then be used for offline analysis of the system responses. Figure E.3, meanwhile, shows the first page of the detailed display for the analog input point CI100. Calibration will define the appropriate range limits for this point, and thus result in an accurate measurement.

**Procedure**

**Getting Started**

There are two main tasks that must be conducted simultaneously during the initial part of the experiment. One is to create a series of standard solutions that will build a correlation between concentration (in \( \text{g/ℓ} \)) versus conductivity (in millimhos). The other task is to run the reactor at various steady-state conditions to establish a relationship between % of scale in the analog input point CI100 and conductivity (which in turn defines concentration).

The conductivity measurements obtained from the online conductivity sensors are displayed as the PV values for CI100 and CI102 in the mixing tank group display. CI102, the inlet stream salt concentration, has been already calibrated and will be referred to as an example in the remaining portions of the manual. The main concern of your group, then, is to calibrate the exit stream sensor by relating a percent-of-scale measurement of conductivity in CI100 to a meaningful value in units of \( \text{g/ℓ} \).

We will describe first the procedure for collecting tank samples needed for sensor calibration. Bear in mind that the mixing tank has been in the past a fickle system to work with; ask your instructor or TA regarding any special considerations that may apply for this semester’s lab. This includes current limit ranges for the process variables.

1. First, becoming familiarized with the flow and level control loops that constitute the mixing tank system (as described in the Background section). Is the level measurement accurate? If not, adjust the level on the reference leg of the d/p cell until the value of LIC100.PV on the TDC3000 agrees with the actual level in the tank.

2. Set all loops to AUTO (with the exception of CIC100) using as setpoints for fresh water a flowrate of 1.5 gpm, a salt water flowrate of 0.2 gpm, and tank level of 35%.

3. Watch the trend of CI100.PV until its value reaches steady-state. At steady-state,
Figure E.1: Brine-water mixing tank schematic (top) and photograph (bottom).
### Figure E.2: Group Display for Mixing Tank

<table>
<thead>
<tr>
<th>CI100</th>
<th>MINING TANK CONCENTRATION HA MIXING TANK EXPERIMENT B</th>
<th>FIRST PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>100%</td>
<td>PUAUTO 3.63738 ALARM LIMITS</td>
<td>POINT DATA</td>
</tr>
<tr>
<td>75%</td>
<td>PUAUTO 3.63738</td>
<td></td>
</tr>
<tr>
<td>50%</td>
<td>PUAUTO 3.63738</td>
<td></td>
</tr>
<tr>
<td>25%</td>
<td>PUAUTO 3.63738</td>
<td></td>
</tr>
<tr>
<td>0%</td>
<td>PUAUTO 3.63738</td>
<td></td>
</tr>
</tbody>
</table>

### Figure E.3: First page, detailed display, CI100 point

<table>
<thead>
<tr>
<th>CI100</th>
<th>MINING TANK CONCENTRATION HA MIXING TANK EXPERIMENT B</th>
<th>FIRST PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>100%</td>
<td>PUAUTO 3.63738 ALARM LIMITS</td>
<td>POINT DATA</td>
</tr>
<tr>
<td>75%</td>
<td>PUAUTO 3.63738</td>
<td></td>
</tr>
<tr>
<td>50%</td>
<td>PUAUTO 3.63738</td>
<td></td>
</tr>
<tr>
<td>25%</td>
<td>PUAUTO 3.63738</td>
<td></td>
</tr>
<tr>
<td>0%</td>
<td>PUAUTO 3.63738</td>
<td></td>
</tr>
</tbody>
</table>
record the value for CI100.PVRAW (keep in mind that this signal is in terms of a 0-100% scale).

4. Obtain a sample solution from the mixing tank. Unplug the stirrer and use a beaker to scoop a sample from the tank. Take your sample to the bench-scale conductivity sensor and obtain a conductivity reading. Then pour the sample solution back into the mixing tank.

5. Repeat this procedure using three different sets of inlet salt flow and fresh water flow setpoints. Select these such that the CI100.PVRAW values adequately span the 0-100% range. Note: Use your first-principles modeling information (the Matlab/SIMULINK model, the linearized transfer function model, or both) to make judicious decisions on these setpoints. Opt for choices of these variables that will also minimize the time to steady-state. You still have half of your lab experiment remaining.

To complete the calibration process, a series of standard solutions are needed that relate salt concentration to conductivity. This is described next.

**Calibration of Conductivity to Estimate Concentration**

This section describes the steps that must be performed to relate concentration in terms of millimhos to concentration in terms of $g/\ell$ in the mixing tank. For accomplishing this job, a series of standard salt solutions must be mixed and measured with a bench-scale conductivity sensor.

**Operating Guide for the Benchtop Conductivity Meter**

The operating guidelines for the conductivity meter located in COB B154 are as follows:

1. Connect the plastic cell to the conductivity meter base.
2. Rinse the plastic cell with distilled or deionized water for use (and eventually, storage).
3. Prepare sample solution in a beaker, and immerse the plastic cell so that the solution completely fills the electrode chamber with no air bubbles present.
4. Adjust the knob from the lowest range to highest range (20 $\mu$mho to 2000 millimho) until you find the proper range for measuring the concentration of sample.
5. When all measurements are completed, rinse the cell and store, per TA instructions.

**Calibration Curve between Conductivity and Concentration**

Figure E.4 shows the calibration curve for concentrated salt solutions, corresponding to the inlet salt stream in the mixing tank. You need to obtain a similar relationship corresponding to the more dilute concentrations exhibited by the exit stream of the mixing tank.

The calibration curve between the conductivity [millimho] and the salt concentration [$g/\ell$] can be generated by using the following procedure:
1. Measure 1, 2, 4, 6, 8, and 10 grams of salt on the scale, and pour each salt amount into four separate one ℓ volumetric round flasks.

2. Fill the 1 ℓ volumetric round flasks using deionized water. Deionized water is available in COB B154.

3. Agitate the flasks for perfect mixing and make sure that the liquid volume equals to 1 ℓ.

4. Measure the conductivity using the conductivity meter and record the conductivity and concentration.

5. Plot the concentration vs. conductivity using MATLAB. The **polyfit** command will compute a linear relationship to the measured data set. In the MATLAB prompt type the following commands (use your own values, of course),

   ```
   >> x = [20 37 71 100 120 140]; % Conductivity [millimho]
   >> y = [1 2 4 6 8 10]; % Salt concentration [g/ℓ]
   >> p = polyfit(x,y,1)
   ``

   The coefficients of `p` (p(1) and p(2)) correspond to the linear relationship:

   \[ y = p(1)x + p(2) \]

   To view how well this curve fits the data, enter the following MATLAB commands:

   ```
   >> ye = polyval(p,x);
   >> plot(x,y,‘o’,x,ye,’-’);
   >> gtext(‘y [g/l] = p(1) x [millimho] + p(2)’);
   >> xlab(‘Conductivity [millimho]’); ylab(‘Concentration [g/l]’);
   ``

   polyval is a MATLAB function which allows you to validate the polynomial fit.

   The calibration between the conductivity [millimho] and the salt concentration [g/ℓ] can now be used to generate data points that define the final calibration curve between salt concentration (in g/ℓ) and % of scale in the CI100 point. An example curve corresponding to the CI102 point is shown in Figure E.5.

   Using the four conductivity measurements from the tank samples and the calibration curve between conductivity and concentration, apply **polyfit** in MATLAB to obtain a final calibration curve for the system. From this curve, determine the concentration value for 0% of scale; this value is the low limit on the instrument range, or PVEULO. The value for 100% of scale, meanwhile, is the high range on the instrument, or PVEUHI. Go to the DETAIL display for both the CI100 and CI100 points, and enter these parameters (have the TA help you here). You are now finished with the calibration portion of the lab; you may proceed with the rest of the experiment.

**Empirical Modeling (System Identification) via Step Testing**

Empirical dynamic modeling in the mixing tank (i.e., system identification) consists of generating process reaction curves by performing step changes in the setpoint of the inlet salt
stream controller. You are asked to generate process reaction curves corresponding to changes in the inlet salt and inlet fresh water flowrates.

Following calibration, the tank should be at steady-state. Record all the PV values corresponding to these steady-state conditions. Have the TA initiate data collection with the DACQTANK point prior to step testing. Perform your first step change on the salt water setpoint (FIC101.SP). Trend and plot the resulting response in salt concentration (CI100.PV), as well as the input (FIC101.SP) and the actual salt flow (FIC101.PV). Make sure that the trends have the same scales and record the range limits for each plot (since all trends are listed in % of scale). As soon as steady-state is reached, record all PV values, then change the setpoint of the fresh water flow (FIC100.SP) while leaving the salt water flowrate unchanged. The fresh water flowrate has a much wider range than that of the inlet brine flow; make sure to enter a significant change in this variable. Trend the appropriate PV and SP values as before. Use graphical analysis to estimate the gain, time constant and delay for this system for each step change. Remember to keep your gain in units of (g/l)/gpm.

First-Principles Modeling

One of your EPA problems involved the use of total mass conservation and salt species accounting to derive the lumped parameter system equations describing the level and concentration dynamics of this system. A second EPA problem asked that you derive the transfer function relating the outlet concentration in the tank to changes in the flow rate of brine to the tank (for this lab you will also need the transfer function relating outlet concentration to fresh water flow, which was not requested in the EPA). These transfer functions explicitly relate gain and time constant to the process parameters (tank volume, salt water flowrate, fresh water flowrate, salt water concentration, etc.) Using the measurements you have recorded for the three different steady-state conditions experienced during step testing, obtain gain and time constant estimates for each input value, and compare these with the empirically-obtained values. The first-principles modeling task will require that each team measure the tank dimensions; a meter stick is provided for this purpose. There is also a hydrometer that can be be used to estimate the density of the inlet brine.

These equations were the basis for the Matlab/SIMULINK model represented by the files brine.m and brine99.mdl. In this experiment, modified forms of these files (brine2.m, brine99b.mdl) will be supplied by the instructor to allow a comparison of the measured data with the predictions made by both the nonlinear and linearized models. These files allow the user to read an ASCII data file generated by the DACQTANK point, and also take into account the action of level control (LIC100) on the system. The use of these programs and any modifications that need to be made by the team will be explained by the instructor during the lab prep session. The parameters of the linearized models depend on the operating conditions chosen; this should be kept in mind when discussing the validity of the linearized model predictions shown by the programs.
Lab Report

Your report should first describe the results of the CI100 calibration procedure, with supporting tables and plots.

You also need to report on the values found for gain, deadtime and time constant for the different experimental runs. Keep the gain in units of (g/l)/gallons per minute as noted previously. Compare the results predicted by the first-principles model versus the experimentally-obtained transfer functions given by system identification. Please note that the term “first-principles modeling results” refers to both the linearized transfer function models and the results from the nonlinear state-space model. How well do the models parameters/predictions agree with each other? How accurate are the linear model predictions versus the nonlinear ones? There are multiple sources of error in this system, as well as a number of assumptions made during modeling which contribute to mismatch between the calculated versus predicted values. Identify as many of these sources of error as you can imagine, and classify them in terms of significance. (Note: steady-state calculations by themselves can help you discern some of these sources of error).

Include in your discussion a candid evaluation of first-principles versus system identification-generated models (recall the statement: “all models are wrong, some models are useful”). State what you perceive are the pros and cons for each technique from the standpoint of an engineering scientist (i.e., a person interested in the most fundamental approach) versus a practicing engineer (i.e., an individual whose main interest is getting a good, fast solution).
Figure E.4: Calibration Curve (Conductivity vs. Concentration) for High Concentration Salt Solutions

Figure E.5: Calibration Curve (Inlet Stream Salt Concentration [g/ℓ] vs. PV [%]) for CI102
Experiment F: Heat Exchanger
Instrumentation

Objective
The object of this experiment is to “configure” or program the TDC 3000 system to read the temperature of a stream in the heat exchanger apparatus, display the temperature on the console and use the displayed temperature to evaluate the dynamics of the system.

Background

Process Manager
The Process Manager provides flexible and powerful process scanning and control capabilities. As seen in the figure the PM consists of the Process Manager Module (PMM) and the Input/Output (I/O) subsystem.

Process Manager Module
The PMM consists of

- Communication Processor and Modem: It provides high performance network communications such as network data access and peer-to-peer communications.
- I/O Link Interface Processor: It is the PMM interface to the I/O subsystem and provides high speed I/O access for communications and control functions.
- Control Processor: It executes regulatory, logic and sequence functions and includes a powerful user programming facility.

The PMM is partitioned into slots. A tagged slot is referred to as a data point in the TDC 3000 system. The eight types of data points configured into PMM slots have been discussed in experiment B.
Figure F.1: Outline of the Process Manager
I/O Subsystem

It performs input and output scanning and processing on all field I/O. The I/O processors work in conjunction with the Field Termination Assemblies. The following 8 I/O processors are available for the PM:

- Analog Input - High Level (16 pts)
- Analog Input - Low Level (8 pts)
- Analog Input - Low Level Multiplexor (32 pts)
- Smart Transmitter Interface (16 pts)
- Analog Output (8 pts)
- Pulse Input (8 pts)
- Digital Input (32 pts)
- Digital Output (16 pts)

*Analog Input* The main function of the analog input processors is signal conversion and conditioning.

- PV Source
- PV Clamping
- EU Conversion
- PV value status
- PV filter (single lag)
- Software calibration

The low level multiplexor processor provides an economical way to bring in a large number of data acquisition signals. The number of points they can handle varies on the type of analog input.

*Smart Transmitter Interface* The smart transmitter interface processor is the process manager’s interface to Honeywell’s advanced series of smart transmitters. It supports the functions for PV processing, EU conversion, and alarming supported by the other analog input processors. It also provides bad PV and bad database protection for added security.

*Analog Output* Analog output processor provides the following functions:

- Readback check of actual output current.
- Output characterization.
- Output default action on failure.
• Modes and associated functions to support
  – Manual loader station
  – DDC control
• Software calibration.

It also provides separate D/A converters and power regulator per channel for maximum output security.

*Pulse Input* Precise control using high accuracy pulsed-type sensing devices is possible with the pulse input processor. The result is improved product quality and reduced material waste.

*Digital Input* It provides the following functions
- Event counting (accumulation)
- Push button and status type inputs
- Time deadband on alarms for status inputs
- Input direct/reverse
- PV source selection
- State alarming for status inputs

*Digital Output* It provides the following functions
- Output types
  – Latched
  – Pulsed
  – Pulse - Width modulated
- Output default action on failure
- Output readback checking

**Heat Exchanger**

Figure G.2 shows a schematic of the heat exchanger along with the locations of the valves and thermocouples. Tap water is run directly into the tube (cold) side. Steam is sparged into a second water line and the resulting hot water goes into the shell (hot) side. The thermocouples are connected to amplifiers which convert the low mV signals into the 4-20 mA range compatible with the TDC 3000.

The control strategy to be implemented is shown in Figure G.3. There is a control loop on the inlet cold water so that this flow rate can be monitored and regulated from the console. The overall objective is to control the outlet tube side temperature by manipulating the flow of steam which adjusts the temperature of the shell inlet water. E.g., if the temperature of the outlet tube side stream is low, the amount of steam can be increased which in turn increases the inlet shell side temperature.
Procedure

Process Point Building

In order for the TDC 3000 to use the temperature in a display or calculation, the 4-20 mA signal must be converted from an analog or electrical signal into a digital number in the computer using a circuit in the Process Manager (PM). Building or configuring a point means using software to indicate where the wires are connected to the PM, defining Tag Name to identify the process variable being built and establishing calibrations so the temperature or other process variable is displayed in engineering units.

The procedure below will allow you to build a point, TI501, corresponding to the temperature of the shell side inlet stream and add it to the points already configured.

Build PM Process Point

1. From the Engineering Personality Main Menu, select NIM, this displays the NIM Build Type select menu.

2. Select Process Point Building from the NIM Build Type select menu, the NIM process point build type menu is displayed.

3. Select the type of process point you are working with from the NIM process point build type menu (in this case it will be an Analog Input point), this will display the PED for the selected point.

4. Fill in the PED configuration data as follows:

```
IDF NET>FW>HTEX1_DB, ENTITY TI501( )

NAME ............ "TI501"
NODETYP ......... PM
PNTFORM .......... FULL
PTDESC .......... "STEAM INLET TEMPERATURE"
EUDESC .......... "DEG F"
KEYWORD .......... "STM IN"
UNIT ............ H6
NTWKNUM .......... 03
NODENUM .......... 17
MODNUM .......... 9
SLOTNUM .......... 1
PNTMODTY ....... HLAI
SENSRTYP ........ 1_5_V
PVCHAR .......... LINEAR
INPTDIR .......... DIRECT
```
Use the [Page Forward] and [Page Back] keys to go from page to page in the PED. Upon completing the data entry for each page, always press [ENTER] key before moving to the next page in the PED. After entering all values into the PED use the load function (F12) key.

**Thermocouple Calibration**

Before turning any valves, make sure that all control loops shown in HTEX5 are in the manual mode and the outputs are set to zero. Have the T.A. turn on the air supply to the instrumentation and the main water supply valve. Valve 5 on the tube side supply line should be closed and valves 2 and 3 open. Put controller FIC551 into the automatic mode and enter a set point of 1.0 gpm and check that that a flow rate of 1.0 gpm is attained. Close valve 9 and open valves 6 and 7, and adjust the flow rate of water to the shell side by outputing 30% manually on FM553, such that it opens valve 8 and corresponds to 10% on the rotameter.

Have your T.A. open the main steam valve. Close valve 18 and open valves 15 and 16 on the steam line. You will use controller TIC552 in the manual mode to adjust the steam flow rate through valve 17.

In order to calibrate the thermocouple, you will need to use a standard, in this case, a digital thermocouple meter. The T.A. will instruct you on the operation of the meter. Over a limited temperature range, the output from the T.C. can be assumed to be proportional to the temperature. Therefore you will only need two data points to establish the calibration. The expected range of temperatures during projected experiments is from 105 to 235 F. Adjust the steam flow rate so that the reading of the raw input (PVRAW) for point TI501 is between 0 and 10 %. When the reading is stable, read the temperature on the digital
meter. Increase the steam flow until you get a PVRAW between 90 and 100 % and record the corresponding temperature. Using the two data points, determine the linear equation that relates temperature to % reading. Determine the temperatures corresponding to 0% and 100% and enter these values into PVEULO and PVEUHI respectively. They should be close to 105 and 233 F. If your results differ by more than 10 degrees, consult your T.A. This completes the calibration procedure and the proper value of the temperature is displayed in F as TI501.

**Dynamic Testing**

Before the control loop can be closed, a model of the process must be obtained. With TIC552 in manual mode, output a value of 25% to the steam valve. Trend both TI503 and TIC552OP. (TIC552OP is trended so that time zero can be identified on the response curves.) When the system is at steady-state, have the T.A. print a copy of the screen. Then introduce a step change to the steam valve by changing the output of TIC552 from 25 to 50%. Wait for the system to come to steady-state and then print a copy of the display. Repeat the process by changing the output from 50% back to 25%. Be sure to print a copy of your final screen.

Assuming the system can be described with a first-order plus dead time model, determine the gain, time constant and dead time for the process using both sets of data. Determine the gain in terms of both °F/%OUTPUT and %PV/%OUTPUT. For your final model, use the averages of the parameters from the two runs.

**Lab Report**

No special instructions here; follow the format described in the syllabus.
Figure F.2: Heat Exchanger Hardware Schematic
Figure F.3: Heat Exchanger Control Strategy
Experiment G: Heat Exchanger Control

Objective

The purpose of this experiment is to 1) obtain a dynamic model relating steam valve position to outlet tube water side temperature in the SCOB B190 heat exchanger and 2) use this model to generate Proportional-Integral (PI) controller settings for the regulatory control point TIC552. You will be asked to compare the performance obtained from various tuning rules on the closed-loop responses obtained from the exchanger for both load and set point changes.

Background

Heat Exchanger

Figure G.2 shows a physical layout of the heat exchanger along with the locations of the valves and thermocouples. Tap water is run directly into the tube (cold) side. Steam is sparged into a second water line and the resulting hot water goes into the shell (hot) side. The thermocouples are connected to amplifiers which convert the low mV signals into the 4-20 mA range compatible with the TDC3000.

The group display for this experiment is shown in Figure G.1; it is composed of the following points:

TI501. This is an analog input point whose PV represents the temperature measurement of the steam/water mixture entering the shell side of the exchanger.

TI502. An analog input point that measures the temperature of the steam/water mixture exiting the shell side of the exchanger.

TI503. An analog input point that measures the temperature of the water exiting the tube side of the exchanger. The PV of this point is the controlled variable for the closed-loop that you will be tuning in this experiment.

TI504. An analog input point that measures the temperature of the water entering the tube side of the exchanger.
**FIC551.** A Process Manager (PM) regulatory control point that controls the water flowrate in the tube side of the exchanger. Changes to the FIC551 flowrate are used in the latter part of the experiment to generate disturbance changes in the exchanger. *This loop should remain in AUTO mode throughout the course of the experiment.*

**TIC552.** Another PM regulatory control point which controls the outlet tube water temperature (TI503.PV) by adjusting steam flow to the shell side (TIC552.OP). *This is the loop that you will be tuning and evaluating in this experiment.*

**DACQHX.** A custom point on the Application Module which writes various process measurements to an ASCII text file. The program saves these values to a file named `heatdat.xx`. This information can be loaded into a Matlab workspace in the same manner as the `ident##a.xx ... ident##c.xx` files in Lab D. The columns are organized as follows:

- **Column 1.** This contains a time stamp for each data record (in 10 second increments).
- **Column 2:** TIC552.OP.
- **Column 3:** TIC552.PV.
- **Column 4:** TIC552.SP.
- **Column 5:** TI501.PV.
- **Column 6:** TI502.PV.
- **Column 7:** TI503.PV. This is the same measurement as TIC552.PV.
- **Column 8:** TI504.PV.
- **Column 9:** FIC551.PV.

**FM553.** An analog output point that specifies the valve position for the water inflow valve on the shell side of the exchanger. The output of this point should remain at 30% open at all times during the execution of this experiment.

The feedback control strategy that is implemented by TIC552 is shown in Figures G.3 and G.4. FIC551 regulates the flow of cold water to the inlet tube side of the exchanger. The overall control objective is to control the outlet tube side temperature (TI503) by manipulating the flow of steam which determines the temperature of the shell inlet water, which is measured by TI501. The inlet tube water temperature and outlet shell side temperature are monitored by TI504 and TI502, respectively.

This experiment requires that your team first generate a process reaction curve via step testing to obtain a transfer function relating steam valve position to outlet water temperature for the heat exchanger. In EPA No. 8, tuning rules such as the Cohen-Coon, ITAE-optimal, open-loop Ziegler-Nichols, and Improved IMC-PI were used to obtain values for the controller gain and integral time constant. In this experiment, your team will be asked to compare the performance of the improved IMC-PI and ITAE-optimal rules for both setpoint tracking and disturbance rejection in the heat exchanger. For details on these methods you may refer to the following portions of your textbooks:
Improved IMC-PI ITAE-optimal

Section 12.3 SEM, Table 15.6, OgRay
Table 12.3 SEM, Table 15.4, OgRay

Detailed information regarding the improved IMC-PI tuning rules is provided in the *Internal Model Control: A Comprehensive Approach* report developed by Dr. Rivera.

![Figure G.1: Group Display for Heat Exchanger Control, showing the results of an open-loop step change in steam valve position from 50 to 25% open.](image)

**Process Manager PID Controller**

This controller equation implemented in TIC552 is the “interactive” PID controller which is one of the standard control algorithms available in the Process Manager of the TDC3000

\[
1 + \frac{T1}{s} \quad 1 + \frac{T2}{s} \\
CVP = K \times \left( \frac{PVP - SPP}{PVP} \right) \\
\frac{T1}{s} \quad 1 + a \frac{T2}{s}
\]

The controller parameters K, T1, and T2 are found in the CONTROL ALGORITHMS page of the detail display for the TIC552 point. The derivative mode can be shut off by setting T2=0. The parameter “a” is fixed by Honeywell to 0.1, and is necessary in order for derivative action to be physically realizable. CVP stands for the “calculated value” of the controller (in percent of scale). This parameter gets further processed by the system to become the controller output parameter (.OP). PVP and SPP are the PV and SP values of
the temperature in percent of scale. Note that \( K \) is expressed in terms of percent/percent, which means that the \( K_c \) obtained from a tuning rule must be scaled by the term

\[
K = \frac{P_{VEUHI} - P_{VEULO}}{C_{VEUHI} - C_{VEULO}} \times K
\]

Please keep in mind as well that \( T_1 \) and \( T_2 \) must be expressed in units of minutes.

**Procedure**

**Action Plan**

Get together as a team and compare your answers to EPA No. 8, Problem 2. From the results of this problem it should be clear that the Improved IMC-PI and ITAE-optimal tuning rules provide much better performance than the Cohen-Coon or open-loop Ziegler-Nichols tuning. Be sure to include a copy of your solutions with your action plan, and demonstrate a working Matlab/SIMULINK simulation of the problem to the instructor.

At this stage of the course, you have been exposed to using the MATLAB function `hextune.m`, as well as the `pidfurn` modeling package. These represent resources which will allow you to effectively and efficiently complete the requirements of this experiment. Your action plan should include how (and where) you plan to incorporate these tools into the execution of your laboratory; your written statements can be further elaborated in your meeting with the Instructor.

**Lab Session**

**Open-Loop Dynamic Testing and Controller Tuning**

The T.A. will turn on the air supply to the instrumentation and the main water supply valve. Valve 5 on the tube side supply line should be closed and valves 2 and 3 opened; similarly, valves 9 and 18 should be closed and valves 6, 7, 15, and 16 should be open. As a result, flows through valves 4, 8, and 15 are directly adjusted by points FIC551, FM553, and TIC552 on the TDC3000. Initially, controller FIC551 will be in AUTO mode with a set point of 1.0 gpm. The water flow to the shell side is set to a constant value by manually specifying a 30% output on FM553, this should correspond to 10% on the rotameter in the heat exchanger panel. Initially, the TIC552 regulatory control point should in the MANUAL mode with the OP value (corresponding to steam valve position) set to 50%. Please be sure that the system has come to steady state before beginning any work.

Prior to system identification, please initiate the data collection routine (via point DACQHX; all data is recorded at 10 second intervals and organized as noted previously). To obtain a process reaction curve for this system, trend both the PV value of TI503 and the OP value of TIC552. (TIC552.OP must be trended so that time zero can be identified on the response curves.) Introduce a step change to the steam valve by changing the output of TIC552 from
Figure G.2: Heat Exchanger Hardware Schematic

Figure G.3: Universal Station - Heat Exchanger TDC3000 Schematic (HTEX5).
50 to 25%; watch and rescale the trend corresponding to this process reaction curve. Be sure to print and/or save a copy of your trended group display (as shown in Figure G.1) and stop the data collection once the system has reached steady-state. While inflection point analysis can be used to obtain model parameters from the process reaction curve, it is best to use \texttt{pIDfurn} to determine the model parameters via regression analysis. Please keep in mind the following when using \texttt{pIDfurn} on this data set:

1. Load the input/output data into the Matlab workspace via the commands

   \begin{verbatim}
   >> load heatdat.xx
   >> u = heatdat(:,2);
   >> y = heatdat(:,3);
   >> T = 10;   % This is the sampling period (in seconds)
   \end{verbatim}

   Start \texttt{pIDfurn} and load/preprocess the input-output vectors as you were required to do in the Lab D data analysis.

2. Since your team will not have the time to generate a crossvalidation data set, use the \texttt{Estimation = Validation} option in the \textbf{Validation Data} section of the first GUI window.
3. Use one of the two information criteria provided in the program (either the Akaike Information Criterion (AIC) or Rissanen’s Minimal Description Length (MDL)) to determine a suitable ARX order for your model. These measures will trade-off the improvement in the goodness-of-fit caused by overparametrization in the ARX estimation problem with the increase in the total number of model parameters.

4. Make sure to examine the simulated model output on the data before proceeding to the Output Error estimation section of the program.

Do not take too long in doing this modeling/identification task, since there are still four closed-loop tests remaining to be conducted in this experiment!

With your model parameters in hand use `hextune.m` to compute controller settings for the IMC and ITAE tuning rules. Be sure to express the delay and time constant values in minutes, not seconds! Your team will implement both ITAE setpoint and disturbance controller parameters, as well as two different filter time constants for the IMC tuning rule to address the setpoint tracking and disturbance rejection modes, respectively. Use the guidelines presented in EPA No. 8 to determine the values for these filter parameters. Do not forget to use the proper instrument ranges (PVEUHI and LO, CVEUHI and LO) to properly scale the gain into percent of scale. The values for these parameters can be found in the FIRST page of the detail display of TIC552. Recognize the value of Matlab with SIMULINK as a useful tool for predicting the results of different tuning parameters prior to closed-loop testing; please simulate each anticipated closed-loop experiment and consult with the TA or instructor prior to implementing these values on the exchanger. Note: You will need the Instructor’s or TA’s “all clear” before proceeding with the closed-loop testing. In the SIMULINK file, use as a disturbance model a first-order transfer function with steady-state gain set to $-1$ and time constant equal to that of the identified model.

Closed-Loop Dynamic Testing

Implement the controller settings obtained in the previous section by entering the values for $K$ and $T_1$ in the CONTROL ALGORITHM page of the detail display for TIC552. $K$ and $T_1$ represent the controller gain and the integral time respectively; as noted in Background section, make sure that your controller gain is properly scaled and that your integral time constant is in minutes. Resume data collection with DACQHX (make sure the initial process reaction curve `heatdat.xx` file has been renamed and deleted) and begin testing using the IMC setpoint tracking controller tuning settings. Make sure to trend (at the very least) the process measurements TI503.PV, the set point TIC552.SP, and the manipulated variable position, TIC552.OP. Since you have multiple Universal Stations at your disposal, consider trending all the process variables in the group display to detect any unforeseen upsets that may affect controller operation. Put the controller in AUTO and observe the initial response of the system. If the controller parameters you have entered are correct, the closed-loop system should remain stable. Introduce a change in set point of $+10 \, ^\circ\text{F}$ and observe the response. What is the closed-loop time constant and settling time? Is the system underdamped, overdamped, or critically damped? Consider the measures discussed in lecture for desirable
closed-loop performance. If the closed-loop response is not satisfactory, how could you change the controller parameters to improve performance? Having reached the setpoint (and some reasonable semblance of steady-state), change the PI controller settings to those for ITAE-optimal setpoint tracking. Introduce a set point change of $-10 \, ^\circ F$ (back to the original set point) and trend the resulting response. Once at setpoint (and reasonably at steady-state), enter the IMC controller parameters with filter value set for disturbance rejection. Introduce a load change into the system by changing the tube side water flow rate setpoint (FIC551.SP) to 1.5 gpm. Record the response. How well was the controller able to maintain the set point in spite of this disturbance? Repeat the load change (from 1.5 gpm to 1.0 gpm) but use the ITAE-optimal load tuning rule settings instead.

Figure G.5: A 2 hr trend summarizing a series of representative closed-loop control results on the heat exchanger. Top curves show the setpoint and controlled variable responses (TIC552.SP and TIC552.PV, respectively); center curve trends the manipulated variable response (TIC552.OP), while the bottom curve shows the disturbance variable (FIC551.PV).

Lab Report

Describe how the dynamic model for your system was obtained, and how controller tuning parameters were arrived at on the basis of this model. Discuss the performance of these settings in terms of the response to set point changes, answering the questions posed previously in this chapter. Display the closed loop responses and comment on their characteristics, e.g.,
was there any offset? overshoot? How do the closed-loop speeds of response compare in the different cases? Some well thought-out tables summarizing the results of the various closed-loop tests may be very useful in your discussion. How did the real-life responses compare to the simulated responses obtained via MATLAB/SIMULINK? What extent of movement was observed in the manipulated variable (note: keep in mind in your discussion the effect that saturation (i.e., high/low) limits on the steam valve have on the closed-loop responses obtained experimentally). Similarly, display and comment on the system responses to the load disturbance changes. Did the response have similar behavior to that obtained with a set point change? Why is it possible to get by with more aggressive controller tuning in the load disturbance case versus setpoint tracking? Based on your experience, describe the pros and cons of the Improved PI versus the ITAE tuning rules. Comment as well on the benefits (and drawbacks) of using information criteria (such as AIC and MDL) versus the use of a crossvalidation test in the open-loop modeling task. What general conclusions can you make regarding the “tuning rule” approach to control system design?

Note: The data collection file generated during closed-loop testing can serve a very useful role in helping your team fully analyze the closed-loop responses obtained from the exchanger. Overshoots, settling times, Integral Square Errors, etc. are much more easily discernible from the data collection file than from the group display trends obtained from the GUS and Universal Stations.
Experiment H: Mixing Tank Conductivity Control

Objective

In this experiment the conductivity of the product stream from the mixing tank is to be regulated by automatic control. The objective of this experiment is to apply PI control to the mixing tank process and obtain a well-tuned response for both setpoint tracking and disturbance rejection.

Procedure

Make sure before commencing the experiment that there is enough solution in the salt water tank to last for the duration of the lab. The salt solution is made by dissolving approximately 10 lbs of rock salt in pails of water and adding the clear (no crystals) solution to the feed tank. Use the TDC 3000 to bring the system to steady-state, using the experience gained from Lab E. The TA will suggest desirable set points for the fresh water flow, salt water flow, and level.

The salt water flow (controlled using $FIC_{101}$) will serve as the manipulated variable for this control system. To determine the transfer function relating conductance of the water in the tank to the salt water flow rate, make a step change in the set point on $FIC_{101}$ and record the response of the conductance ($CIC_{100}$). You will want to make the setpoint change large enough to insure a good signal-to-noise ratio for the resulting output response. Assume a first-order with delay model and graphically determine gain, time constant and dead time for the process. The gain should be expressed in both micromho/gpm and as percent output/percent input where % input represents the % of the range on the water flow rate set point scale corresponding to the salt water setpoint change and % output would correspond to the percent change seen on the conductivity scale. This is necessary since the TDC 3000 uses a controller gain in terms of percentages of input and output signals and not engineering units (see the Lab I write-up, for example, for more details).

Having obtained a model, please use MATLAB/SIMULINK to compare the tunings generated for this system from two different tuning rules. Note that the control system is cascaded, since the output of the concentration controller adjusts the setpoint on the salt water flow controller. Compare the improved IMC-PI tuning rule (Rivera et al., 1986 paper) with either one of the following tuning rules: Cohen-Coon (Table 12.2 SEM), ITAE...
(Table 12.3 SEM) or open-loop Ziegler-Nichols (Table 13.3 SEM). Whatever choice you make, please simulate your settings against the estimated model using MATLAB/SIMULINK or CONSYD. Choose adjustable parameters in the IMC rule such that a gentle overdamped response is obtained (with speed-of-response slightly faster than open-loop so you won’t have to spend all day in the lab). The other tuning rules do not have adjustable parameters; be sure, however, that you will obtain a stable response.

Having validated your controller tuning via simulation, enter the IMC-PI parameters on the TDC 3000 system (the TA will assist in this step). Examine the performance of the control system by applying a 100 micro-mho step change to the conductivity set point. Adjust the tuning constants on-line until the desired controller performance is obtained (this is accomplished by introducing a step change in set point after every adjustment). Once satisfactory performance is obtained for set point tracking, examine the controller performance subject to a step change in disturbance. Let the fresh water flowrate (FIC100) be the disturbance variable and introduce a step setpoint change in this variable. Record the response of the conductivity. Repeat the testing procedure (setpoint tracking and disturbance rejection) using the controller generated from the second tuning rule.

**Lab Report**

Be sure to characterize the controller performance in terms of the rise time, settling time, overshoot, and decay ratio. Report these values for the response of the conductivity when the system is subject to setpoint and disturbance step changes, and compare. Also discuss the pros and cons of each technique used for the analysis.
Experiment I: A/B Mixing Reactor Control

Objective

This experiment requires that you design and tune the temperature control system for the A/B Mixing Reactor featured in Lab B. You will draw upon your earlier experience with this system in Lab B and the results of various EPA problems related to this laboratory. The lab is comprised of two parts: identification of the temperature dynamics, followed by controller tuning.

Background

Your job here is to generate and test controller tuning parameters for the temperature control point $TIC^{2190\#}$. In EPA No. 10, Problem 3 you were asked to examine control laws for an integrating system which resulted in tuning rules for a Proportional-Derivative and Proportional-Integral-Derivative controller. These controllers can be implemented using the “interactive” form of the PID controller which is one of the standard control algorithms available in the TDC 3000

$$\frac{1}{T1*S} + \frac{1}{T2*S}$$

$CVP = K \cdot \left( \frac{1}{T1*S} \right) \left( \frac{1 + a*T2*S}{1 + T1*S} \right) \left( \frac{PVP - SPP}{PVP - SPP} \right)$

The controller parameters $K$, $T1$, and $T2$ are found in the CONTROL ALGORITHMS page of the detail display for the $TIC^{2190\#}$ point. The integral mode can be shut off by setting $T1=0$. The parameter “$a$” is fixed by Honeywell to an unspecified value (probably somewhere between 0.05 and 0.1), and is necessary in order for derivative action to be physically realizable. $CVP$ stands for the “calculated value” of the controller (in percent of scale). This parameter gets further processed by the system to become the controller output parameter (.OP). $PVP$ and $SPP$ are the PV and SP values of the temperature in percent of scale. Note that $K$ is expressed in terms of percent/percent, which means that the $K_c$ obtained from your tuning rule must be scaled by the term

$$\frac{PVEUHI - PVEULO}{CVEUHI - CVEULO}$$

$K = \left( \frac{PVEUHI - PVEULO}{CVEUHI - CVEULO} \right) \cdot K_c$
The heat exchanger control point (TIC552) also uses the interactive form of the PID controller, although only PI tunings are evaluated in this experiment.

**Procedure**

**Before Lab Time**

Although no action plan is required for this experiment, pre-lab preparation is essential. Please review the solutions to EPA No. 10, Problem 3 and EPA No. 13, Problem 1 and have the identification procedure and both sets of tuning rules written in your lab notebook prior to the start of the lab period.

**System Identification**

To generate a process reaction curve that will help you model the temperature dynamics of this system, perform the following sequence of operations on this reactor:

1. Starting with a reactor that is clean, empty, and ready, fill the reactor with 100 gallons of Solution A and 50 gallons of Solution B.
2. Stir the contents for 60 seconds.
3. Set the flow controller $FIC^{2190}$ to AUTO
4. Increase the SP to 10 klb/hr.
5. Wait for the contents temperature to rise 5 deg C.
6. Shut the steam flow.
7. Drain the contents.

*Since this is a simulation and to save time, the first two steps can be ignored.*

Print your group trend display corresponding to this sequence of operations and use it to fit an integrating system with delay, just as you did for EPA No. 13, Problem 1. *Please do not forget to account for a step change of magnitude $A$ in your model gain estimate.* Use these identified model parameters for the subsequent control portion of the experiment.

**Controller Tuning**

To begin testing the temperature controller, it must first be *commissioned*; begin by placing $FIC^{2190}$ in CASCADE mode; follow that by setting $TIC^{2190}$ in AUTO (make sure the controller parameters have been entered first). Carry out the following steps:
1. Start with the PD controller tuning using setting of $\lambda = 1.5$ min. Do not forget to calculate your controller gain in % of scale; make sure $\tau_I$ and $\tau_D$ are in minutes. Test your settings by making a 10 degree setpoint change in the reactor contents. Make sure that you trend the temperature PV, temperature setpoint, the manipulated variable (steam flow setpoint) and the steam flow PV. What constraints are active on your system? Do you get overshoot and/or offset?

2. Use the PID settings, this time with $\lambda = 1.0$ min. Again, observe which constraints are active, and check for overshoot and/or offset.

3. Return to the PD controller settings, and try enforcing rate-of-change limits by putting a value on the parameter OPROCLM, also found on the CONTROL ALGORITHMS page. This parameter stands for Output Rate-of-Change Limit and specifies the maximum rate, in percent per minute (%/min), at which the output of the controller is allowed to change. What is the effect of this constraint on your ability to meet your controller performance requirements?

**Lab Report**

The presence of hi/lo and rate-of-change constraints on the steam flow affect achievable control performance in this system. Discuss the scope and magnitude of these effects in your report. One way of diagnosing the problems created by constraints is by simulating the closed-loop behavior of this plant. To obtain full credit for this lab, generate MATLAB with SIMULINK simulations that show the performance of the control system under constrained and “constraint-free” conditions (note: to enforce the constraints in SIMULINK, use the saturation block that you have used in previous exercises. You need not simulate the rate-of-change (OPROCLM) constraint effect).

This lab is based on a very simplified model of the reactor dynamics. Please comment in your report on how different aspects of batch processing (different reaction kinetics, different reactants, varying amounts of A and B, etc.) would affect the identification and controller tuning procedure presented here.
Experiment J: Furnace Custom Control Strategy

Objectives

The principal objectives in this experiment are to tune and compare two kinds of feedback controllers on the gasoil furnace simulation. You will build upon the system identification results obtained for the furnace partition assigned to your team in Lab D. The controllers to be evaluated will be implemented via the following Application Module Regulatory Control Points:

PID1##. This point will use a digital implementation of the ideal PID controller law, which is available as a standard algorithm in the TDC3000.

PIDWF1##. A Control Language (CL) program replaces the standard PID control algorithm. This code implements a digital version of an ideal PID controller augmented with filter, and includes a built-in tuning rule based on the Internal Model Control (IMC) Design Procedure.

As shown in Figures J.1 and J.2, a switch point (SWITCH##) will allow your team to select between the two controller points PID1## and PIDWF1##. The ## in the tagnames refers to your team partition number, as provided to you in Lab D. Your team will test and compare each controller according to the following criteria:

1. Controlled and manipulated variable responses to step setpoint changes

2. Controlled and manipulated variable responses to disturbances (deterministic load changes in the feed flowrate, and stochastic drift)).

3. Controller sensitivity to measurement noise.

Furthermore, you will be asked to assess the robustness of your controller (i.e., the ability to handle plant/model mismatch) by generating experimental results using different models, and by comparing your responses against the simulated responses predicted using MATLAB with SIMULINK.
Figure J.1: Universal Station-based furnace control schematic, showing switch and regulatory control points.

Figure J.2: GUS-based furnace control schematic, with change zone display.
Background

The Application Module is an LCN-resident node in the Honeywell TotalPlant System / TDC 3000 which permits the implementation of more complex control calculations and strategies than are possible when using only process-connected devices, such as the Process Manager (PM). These custom algorithms and control strategies are implemented using Honeywell’s Control Language (CL) inserted at user-specified locations in the AM point. The AM supports seven different types of points: regulatory control, numeric, counter, flag, timer, custom and switch points. In this experiment we focus on the regulatory control and switch points.

Figure J.3 describes the various steps that are carried out by a regulatory control point. These are divided into two categories: steps that receive and process a plant measurement (PV PROCESSING) and steps that calculate and send out a controller output signal (CONTROL PROCESSING). The CL “insertion points” represent locations in the sequence of processing steps where control language blocks can be used to augment or replace an existing algorithm. In this lab, you will use a control language block inserted at the Control Algorithm Calculation (CTL_ALG) to replace the set of default TDC algorithms with a PID with filter algorithm that is tuned internally using an Internal Model Control tuning rule. This enhanced regulatory control point, PIDWF1##, will incorporate the results of various Exam Preparation Assignments. PID1##, on the other hand, uses no control language insertion, but rather implements a digital version of the “ideal” PID algorithm in classical form (EQNA in Honeywell terminology)

\[ u(s) = K_c (1 + \frac{1}{\tau_I s} + \tau_D s) e(s) \]  

(J.1)

Important implementation details: Remember that for PID1## (the standard PID controller point) the controller gain must be expressed in terms of percent/percent, which as noted in the Lab G manual means that \( K_c \) obtained from your tuning rule must be scaled by the term

\[ \frac{PVEUHI - PVEULO}{K} \]

\[ \frac{CVEUHI - CVEULO}{c} \]

Furthermore, T1 and T2, the integral (\( \tau_I \)) and derivative (\( \tau_D \)) time constants, respectively, must be in units of minutes.

The custom control strategy point PIDWF1##, on the other hand has a built-in tuning rule so your only worry in properly tuning this controller is entering the correct plant model and filter parameters. The point has been configured so that the model and filter parameters are entered through the CUSTOM page on the detail display of PIDWF##. Here the plant gain \( K_p \) (KP) must be in engineering units, while the time constant (TAUP) and delay (THETAP) must be in units of seconds, just as was calculated for Lab D analysis. The IMC adjustable parameter \( \lambda \) (expressed as TAUCL in the CUSTOM page) must also be in units of seconds.
PV PROCESSING

PV Input Processing
PV Input Processing
PV Filtering and Range Checking
PV Source Selection
PV Alarm Processing
PV Calculation

CONTROL PROCESSING

Initial Control Processing
Target Value, or Advisory Deviation Alarm Processing
Deviation Alarm Processing
Control Algorithm Calculation
Control Output Processing
Alarm Distribution Processing

PRE_GI
PRE_PVPR
PRE_PVAG
PV_ALG
PST_PVAG
PST_PVFL
PRE_PVA
PST_PVPR
PRE_CTPR
PRE_SP
PRE_CTAG
PST_CTAG
PST_CTPR
PST_GQ
BACKGRND

Figure J.3: AM Regulatory Control Point Processing Steps

PRE_PVPR, PST_PVAG, PRE_PVFL, PRE_PVA, PST_PVPR, PV_ALG, PST_PVAG, backgrnd, are insertion point names.

PST_CTAG, etc., are insertion point names.
Figure J.4: Graphical User Interface for *furntune* program

Figure J.5: Figure window for *furntune*, comparing ideal PI, PID, bumpless PID, and PID with filter closed-loop responses
Procedure

Action Plan

Lab J is the “capstone” experiment for ChE 461 and is assigned the most points of any laboratory work product. As a result, I expect to see your best efforts applied to this experiment. Please use your time in lab wisely and distribute your tasks efficiently among your team members. Pay close attention to the lab prep session on this subject, which will provide you with a good overview of what needs to be accomplished in the lab. Your action plan should include:

1. A statement of the models obtained from Lab D which will be used to determine controller parameters in this experiment. One of these models should be the one arising from graphical analysis of the raw data; the other should be a model arising from regression analysis via pIDfurn (either from the low noise step testing or PRBS data - your choice).

2. Controller parameters (obtained via tuning rules) and closed-loop simulations obtained from these models for the PID and PID with filter structures to be tested in the experiment.

A graphically-based MATLAB program entitled furntune has been developed to assist you in carrying out the results of this laboratory. Pictures of the main GUI and figure windows for furntune can be seen in Figures J.4 and J.5, respectively. Remember that you will need to evaluate two different models, so two different sets of simulations (involving two different sets of tuning constants) will be involved in your analysis. Because the elapsed time of a closed-loop test on the TDC3000 can be substantial, the effective use of MATLAB and SIMULINK will help expedite completion of the lab tasks (as opposed to tedious trial-and-error). As a guideline for the initial value of $\lambda$ use the tuning relationship

$$\lambda = \alpha(\tau + 0.5\theta)$$  (J.2)

with $\alpha = 0.85$. Observing the simulated responses, these settings may be too slow (or too fast). The control performance requirements you should aim for setpoint tracking are:

*Controlled Variable Response.* No oscillations (other than those arising from noise) with a settling time between 0.8 and 1.25 that of the open-loop plant.

*Manipulated Variable Response.* Mild oscillations with at most 20% overshoot. The rate-of-change on the manipulated variable should not exceed 35% of scale per minute.

Lab Session

Please carry out the following tasks:

1. First, familiarize yourself with the various points, group displays, and schematics that you will be working with during the experiment. When you arrive in the lab, you will notice that your partition’s assigned group display has been modified with the addition
Figure J.6: Group Display for Furnace Control at the beginning of the laboratory session

Figure J.7: Group Display for Furnace Control, illustrating proper settings for the FIC41## and SWITCH## points.
of the switch (SWITCH##), standard PID (PID1##), and custom PID with filter (PIDWF##) points. Initially all three points are inactive and in MANUAL mode, as seen in Figure J.6. Please go to the first page of the detail display for each point and make each point active through the proper selection of the PTEXECST parameter. Then make sure to set the SWITCH## and FIC41## points to CASCADE mode. After these changes your group display should look as shown in Figure J.7.

2. Begin the tasks that will allow your team to test each control point using 50 degree setpoint changes. The presence of two models for the plant (one from the graphical analysis, the other from regression analysis) means that you will have two sets of closed-loop responses for each controller.

Start your test using PID1## (pick Selection 1 in SWITCH##) using the tunings based on the graphical model parameters. Note that the switch point will ensure that only one controller is cascaded to the fuel gas control system. Be sure to begin data collection (via point DACQ##) before introducing any setpoint changes. Set the controller to AUTO mode and observe the action of the control system under automatic control. Any instability in the tuning parameters is usually reflected in rather wide swings of the fuel flow setpoint before any setpoint changes are made. Convinced (at least initially) the the tunings are stable on the system, make a setpoint change in the controller from 300 to 350 degrees; observe and trend the responses (as shown in Figure J.8). Having reached steady-state, set the PID1## controller to MANUAL, switch to PIDWF1##, (making use of the tuning parameters generated from the graphical model), set this controller to AUTO, and take the system back to 300 degrees. A typical response is shown in Figure J.9. Repeat the testing procedure using controller parameters obtained from your selected PIDfurn regression model. (Note: having both controllers in MANUAL when you switch will always insure that the error terms are initially at zero; please note as well that if the switching is done using the embedded display on the GUS station these steps are carried out automatically).

3. Having generated the four setpoint changes described in the previous item (two for each model corresponding to the PID and PID with filter controllers), make a determination as to which model resulted in the “best” closed-loop responses for the control system. Use the deterministic performance criteria discussed in lecture and the similarities (or dissimilarities) between the responses observed on the TDC3000 system before those obtained “offline” from furntune. Having made a choice of this “best” model, consistently apply tunings based on this model to the remaining closed-loop tests in the experiment.

4. EQNB “bumpless” control action. From simulations and actual implementation you will have noticed that the ideal PID controller response (PID1##) using EQNA displays large initial move sizes for step setpoint changes. To address this problem, reconstitute your PID1## point and replace EQNA with EQNB of the ideal PID controller, which is the so-called “bumpless” form of the PID controller

\[ u(s) = K_c\left(1 + \frac{1}{\tau_i s}\right)e(s) - K_c\tau_D s y(s) \]  

(J.3)
Figure J.8: Group Display for Furnace Control, illustrating proper PID controller operation

Figure J.9: Group Display for Furnace Control, illustrating proper PID with filter controller operation
To make the equation change in the PID1## point, it will first have to be made inactive on the detail display. The instructor or TA will assist you in using the Builder Commands in the Engineering Main Menu to access the Parameter Entry Display (PED) for the point and locate the “Algorithm Equation Type” parameter in which the equation type is defined. As you navigate through the PED, you will see many of the important configuration parameters that define a regulatory control point, as was described in the lab prep session.

Enter the change to EQNB in the PID point, and then go back to the DETAIL display and reactivate the point. Switch to PID1## and evaluate this controller for a +50 degree setpoint change as before. Make sure that the tuning parameters correspond to those of the “best” model as described in the previous step. Compare the responses with those obtained from the ideal (i.e. non-bumpless, EQNA) PID controller.

5. Robustness to plant/model mismatch. For this task restrict yourself only to evaluating the robustness of the PIDWF1## point. Introduce 25% changes in the gain, time constant, and delay of your controller model, as defined in the CUSTOM data segment (the sign of the changes is at your discretion). Leave \( \lambda \) (TAUCL) unchanged. Switch to the PIDWF## point and test the control system response to a -50 deg setpoint change. How does the performance of the control system compare relative to the previously observed response? Once the control system reaches steady-state (assuming the tuning is internally stable), make sure to restore the controller model back to its original values.

6. Rejection of Deterministic Feed Flow Changes. In this portion of the lab, you are asked to evaluate the effectiveness of each controller in rejecting deterministic load disturbances arising from ±10 MBBL/day changes in feedflowrate. In EPA No. 9 you were asked to use furntune to determine \( \lambda \) settings for each controller that achieve quarter damping. In this portion of the experiment you can either apply these settings or choose alternate settings that your team feels are appropriate. Note: the control system should bring back the outlet temperature to setpoint as quickly as possible without creating large move sizes; some oscillation may be acceptable.

Having made this decision, enter the desired tuning parameters in your regulatory control points, and switch first to the PID1## point. You will need to ask the instructor or TA to introduce a +10 MBBL/day increase in the feed flowrate. Once the control system reaches steady-state, switch to the PID with filter controller point and ask the TA to introduce a -10 MBBL/day change in feedflow. Compare the closed-loop results obtained from both control systems.

7. Control of Drift and “Variance Transfer”. This portion of the experiment needs to be initiated before the end of the lab period and will run for 12 hours afterwards. Place both controllers in MANUAL and stop data collection. Transfer the data file to your storage space. Please note that this data file will have all the responses corresponding to the deterministic setpoint tracking and disturbance rejection cases described previously. Ask the TA or instructor to reset the simulation so the process experiences the same kind of drifting disturbance that was present during PRBS testing in Lab D. Resume
data collection (this time with a clean file) and then proceed to select the “Timed Switch Mode” in your assigned GUS display. Pick “4 hours” as the evaluation interval. Activating this button will result in the following sequence of events: both controllers will be in MANUAL for four hours (allowing the data collection file to capture the “open-loop” drift), followed by PID1## being active for four hours, switching then to PIDWF## for another four hours. The TA will stop data collection after the 12 hours have elapsed and will post the data collection file in your team’s file exchange area in the course web page, as was the case in Lab D. Use MATLAB, Excel, or any computer-based tool to analyze the variance for $y$ and $u$, before and after control action, as was demonstrated in EPA No. 9. *Many thanks to ChE 598 student Chris Schene for coding the timed switch mode on the GUS furnace control schematic.*

Your group may wish to carry out additional tasks for the sake of extra credit and “ChE 461 brownie points.” These include:

*Rename your assigned Group Display with a “catchy” phrase.* Here you will need help from the TA or instructor to reconstitute your assigned group display from the Engineering Main Menu. Do not enter any inappropriate statements (vulgarity, hate speech, etc.), please. However, the more you can make Dr. Rivera laugh with your selection, the more brownie points your team will earn...

*“Bumpless” control action using EQN C.* Reconstitute the PID controller response (PID1##) and replace EQNB with EQNC of the Honeywell PID controller, which applies “bumpless” action to both the proportional and derivative modes

$$u(s) = K_c \frac{1}{\tau_I s} e(s) - K_c (1 + \tau_D s) y(s) \quad (J.4)$$

Try this controller for both a 50 degree setpoint change as before. Make sure that the PID parameters you use correspond to those from the “best” model, as determined from the results of Lab J1. Compare this result with that obtained from the EQNA and EQNB PID controller responses.

*Try out pIDtune.* *pIDtune* is a commercial PID tuning package developed by EngineSoft, a leading control engineering and software development establishment located in Tempe, Arizona. *pIDtune* differs from the *pIDfurn / furntune* in that it fully integrates identification data with PID controller tuning. *pIDtune* uses the frequency response from the ARX model to generate a “control-relevant” model that conforms to the IMC tuning. *pIDtune* also includes built-in simulation capabilities which allow the user to ascertain the effectiveness of the tuning parameters prior to controller implementation. Use *pIDtune* with your experimental data from Lab D (either low noise or PRBS data), and compare the tuning parameters (and resulting TDC3000 closed-loop responses) to those obtained from your own models. Evaluate *pIDtune* for the PID1## point first; consider both the setpoint tracking and load rejection cases. If time permits, you may try the *pIDtune* settings for the PID with filter controller.
Lab Report

Combined with Lab D, Lab J represents the culmination of all that has been taught in this course, so I expect you to show your best effort. Follow closely the format suggested in the Lab Syllabus. I will place particularly strong weight on Part 4 of the lab report format (statement of conclusions that can be drawn from the results). Just putting together a stack of plots with little or no explanation will not be satisfactory. Make meaningful comparisons between controllers (example: ideal PID versus PID with filter) and responses (setpoint tracking, disturbance rejection (to deterministic load changes and stochastic drift), noise sensitivity, robustness), and draw conclusions regarding the benefits or drawbacks of each controller in terms of performance and ease-of-use. Use tables and other graphical information to effectively make your point.

You should be able to answer the following questions in your report: which controller(s) are superior, under what circumstances, and why? Which Lab D model yields tuning parameters displaying superior closed-loop performance? How would practical engineering considerations (e.g., ease-of-use by operators and engineers, ease-of-implementation) affect your opinion on which control system is best?

The due date for all Lab J reports (regardless of when your team performs the experiment) is 5:00 p.m. on Friday, November 30. Additional guidelines on discussion and documentation relevant to this lab may be given during lecture by the instructor.

Data Collection Program

An enhanced data collection program will be available to you during this experiment. The ASCII file generated from the program (ident##d.xx for low noise/“deterministic” conditions, ident##e.xx for stochastic/drifting conditions) can be loaded and read in Matlab in exactly the same manner as was described in Lab D. It goes without saying that these data files can provide you with much more detailed information about your team’s closed-loop responses than those that can be observed from printouts or screen dumps from a native window display. The data collection file for this lab is structured as follows:

*Column 1.* This contains a time stamp for each data record (in 10 second increments).

*Column 2:* FIC419##.PV. This is the actual value of the fuel flow fed by the control system to the furnace. It represents the output of the inner loop in the cascade control system.

*Column 3:* FIC419##.SP. This is the desired or reference value of fuel flow fed by the control system to the furnace. This is the manipulated variable for the outer loop of the cascade control system being considered in this experiment.

*Column 4:* TI418##.PV. This is the measured value of the outlet temperature from the gasoil furnace, and represents the controlled variable for the outer loop of the cascade control system.
**Column 5:** FI418##.PV. The feed flowrate value represents a measured disturbance in the control system.

**Column 6:** PID1##.SP. The setpoint for the standard TDC3000 PID control point, as described at the beginning of this lab chapter.

**Column 7:** PIDWF1##.SP. The setpoint for the custom TDC3000 PID with filter point, as described at the beginning of this lab chapter.

## in the tag names refers to the partition number for your team per Lab D.

**CL Program**

A Control Language package is composed of two parts. One is the custom data segment, which contains names of point-specific parameters which are called by the control language block:

FILE NET>S00>PADEFB.CL 05 Aug 92 11:02 SS

PACKAGE

```
---=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-
CUSTOM

PARAMETER KP "Gain - Process"
PARAMETER TAUP "Lag Time constant - Process; Units = Seconds"
PARAMETER THETAP "Dead time - Process; Units = Seconds"
PARAMETER STP "Sample time - Process; Units = Seconds"
PARAMETER RESET_C:LOGICAL "set ON to restart(initialize) controller"
PARAMETER TRACK_PV:LOGICAL "set ON to have PV tracking"
PARAMETER TAUCL "Adjustable parameter:Closed loop time constant"
PARAMETER KC "Tuning parameter:Controller gain"
PARAMETER TI "Tuning parameter:Integral time"
PARAMETER TD "Tuning parameter:Derivative time"
PARAMETER TAUF "Tuning parameter:"
PARAMETER KERR "Controller Coefficient: e(k)"
PARAMETER KERR1 "Controller Coefficient: e(k-1)"
PARAMETER KERR2 "Controller Coefficient: e(k-2)"
PARAMETER KMOV1 "Controller Coefficient:dm(k-1)"
PARAMETER ER "Current error:e(k)"
PARAMETER ER1 "Previous error 1:e(k-1)"
PARAMETER ER2 "Previous error 2:e(k-2)"
PARAMETER MOVE "Move :m(k)-m(k-1)"
PARAMETER MOVE1 "Move :m(k-1)-m(k-2)"

END CUSTOM
```
BLOCK PADEPRG (GENERIC $REG_CTL; AT CTL_ALG)
LOCAL ALPHA

-- the SP tracks the PV when
-- the point is in MANUAL MODE
IF (TRACK_PV = ON AND MODE=MAN) THEN SET SP=PV

IF RESET_C = ON THEN(
&-- initialize controller coefficients, errors and moves
& SET ER,ER1,ER2,MOVE,MOVE1,KERR,KERR1,KERR2,KMOV1 = 0.0;
& SET RESET_C = OFF)
--
SET ER = SP - PV
-- Calculate controller tuning parameters
SET KC = ****************************
SET TI = * Homework Problem *
SET TD = *
SET TAUF = ****************************
-- calculate controller coefficients
SET ALPHA = ****************************
SET KERR = *
SET KERR1 = * Homework Problem *
SET KERR2 = *
SET KMOV1 = ****************************
-- calculate current move
SET MOVE = KERR*ER + KERR1*ER1 + KERR2*ER2 + KMOV1*MOVE1
-- store previous errors and moves
SET ER2 = ER1
SET ER1 = ER
SET MOVE1 = MOVE
-- calculate current output
SET CV = OPEU + MOVE

END PADEPRG
END PACKAGE
Experiment K: pH Reactor Control

Objective

All the experiments you have conducted in the laboratory thus far have dealt mainly with linear control problems. This experiment will introduce you to a control problem displaying significant nonlinearity and chemical reaction. A laboratory scale pH reactor is used to demonstrate this phenomena. The apparatus considered (Figure K.1) is a CSTR with a strong acid-strong base mixture which reaches equilibrium almost instantaneously. Sodium hydroxide and Hydrochloric acid solutions with concentration of 0.001 M are used as feed to the reactor. The feed is slightly acidic in nature, representing the real life situation of most of the waste water treatment plants. In the control part, pH of the effluent stream is the measured variable and the base flow rate is the manipulated variable for the system.

Background

The pH Problem

pH control has been found to be of great use in waste water treatment and biotechnology processing. pH control presents difficulties due to large variations in process dynamics, for e.g., in the case of a waste water stream from a steel mill the magnitude of fluctuations in flow rate and pH are in the range 2000-10,000 gpm and 2-14 respectively [2]. The main difficulty arises from a static nonlinearity between pH and concentration, which depends on the substances in the solution and their concentrations. This nonlinearity is most severe around the neutralization point (pH=7) zone, where very large changes in the process gain occurs, causing problems for a linear controller.

The nonlinear gain feature of this problem can be visualized by examining a typical titration curve for strong acid-base mixture, as shown in Figure K.2. Effluent pH is the measured variable and the caustic flow rate is the measured quantity; the slope of the titration curve at any pH set point reflects the steady-state process gain. It is seen from the titration curve that the gain changes drastically with changes in pH.

A Control Problem Solution

Some basic tenets of proper pH control most commonly followed in industry [2] are:

- Good mixing.
Experiment K: pH Control

23 May 93 12:12:30  3

Figure K.1: pH Reactor Schematic

- Minimize dead time in the entire loop.
- Enough residence time should be allowed for neutralization to take place.
- Characterize the controller with the titration curve.

The first three items are taken into account via intelligent process design and are independent of the action of the control system. The last item, meanwhile, requires a modification to the control strategy, as will be discussed further.

In this experiment we focus on a strong-acid/strong base mixture and apply a solution to the problem suggested in the text by Åström [1]. A more detailed solution to this problem can be found in the recent papers by Wright and Cravaris [5] and Wright et al. [6]. The basis for the controller problem solution is that nonlinearity can be overcome by converting the pH measurement into an estimated concentration of excess base in solution \( x \). The control problem thus consists of using this estimated concentration as the controlled variable in lieu of pH, which transforms the control problem from a nonlinear problem into an equivalent linear one. Consider addition of hydrochloric acid and sodium hydroxide to water in concentrations \( x_A \) and \( x_B \) respectively. The reaction can be written as:

\[
x_A \text{HCl} + x_B \text{NaOH} + H_2O = [H^+] + [Cl^-] + [Na^+] + [OH^-]
\]

with the corresponding charge balance

\[
[H^+] + [Na^+] = [Cl^-] + [OH^-]
\]
Figure K.2: Titration Curve - Strong acid-base mixture
Because HCL and NAOH represent a strong acid-base pair, we can assume total dissociation, leading to the material balance

\[ [H^+] + x_B = x_A + [OH^-] \]

from which we obtain an estimate of the excess concentration of base in the tank from the pH measurement

\[ x = x_B - x_A = [OH^-] - [H^+] = \frac{K_w}{[H^+]} - [H^+] = 10^{pH-14} - 10^{-pH} \]

\[ K_w = 10^{-14} \] is the dissociation constant for water. The conversion relation

\[ x = x_B - x_A = 10^{pH-14} - 10^{-pH} \quad (A) \]

is the basis for our “linearizing” control strategy (to be distinguished from a “linearized” solution - think about this).

If we considered a strong base-weak acid pair, the corresponding excess base concentration is

\[ x_B = 10^{pH-14} - 10^{pH} + \frac{x_A}{1 + 10^{pH_A-pH}} \]

Note that the excess base concentration estimate in this case cannot be estimated exclusively from a pH measurement. Mixtures of weak acids and bases lack a simple linearizing transformation as Eqn. (A). In these cases, concentration of the components must be measured on-line or estimated from a titration curve ([5],[6]).

The modified control structure for the pH problem is shown in Figure K.3. The transfer function between \( u \) and \( x \) is linear as the measured pH and the reference pH are transformed into equivalent concentrations using equation A.

![Figure K.3: Modified Control Structure](image)

**Description of the apparatus and implementation in the TDC3000**

The apparatus used here is a continuous stirred tank reactor as shown in Figure K.1. The acid and base inlets are positioned so as to facilitate uniformity along with the variable speed
agitator. The reactor has two outlets, one at the bottom and the other at the top. The top outlet is always left open so as to prevent any overflows. The sensor consists of an electrode and displays the pH by means of a meter.

The acid and base solutions are stored in overhead tanks and their flow to the reactor is controlled by two variable speed pumps. These are positive displacement pumps whose flow rate is directly proportional to pump speed. These pumps are interfaced to the TDC and can be operated from the universal station. The acid pump has been set up to be operated in MANUAL mode and the base pump takes its input from the controller point.

Only one sensor for the purpose of measuring the pH is used. The sensor is a pH electrode which is connected to the meter. The output from the pH sensor is a voltage signal in the range -5 to +5 volts corresponding to pH in the range 2 to 12. This voltage signal is fed as input to a transmitter calibrated to output current in the range 4-20 mA, this is done to attain compatibility with the TDC3000.

In this experiment both Process Manager and Application Module points are used. The Process Manager incorporates the pH indicator point and the pump analog output points. Regulatory points in the Application Module permit the use of the linearizing transformation of pH into excess base concentration. The operating groups for this experiment are shown in Figures K.4 and K.5. A brief description of the points in the operating groups follows:

**PHI101** A High Level Analog Input point in the process manager, is used for the purpose of measuring the pH. This uses the pH electrode as the sensor and takes 4-20mA signal as the input. This 4-20 mA signal is converted to 1-5 V in the process manager.

**PHMA & PHMB** Analog Output points in the process manager, the signal from the points run the pumps. PHMA and PHMB represent the acid and base pumps respectively. The outputs from these points are 4-20 mA signals which are converted to 1-5 V for compatibility with the pumps. These pumps can either run in MANUAL mode or can take the output of a controller.

**AFI100 & BFI100** Numeric points which indicate the flow through the pump for the corresponding output percent. A custom point connected to an AM-CL program is used for this purpose. The AM-CL code implements the relationship between the flow and the percent output. A copy of this code is included at the end of the write-up.

**PHIC101** A regulatory point in the Application Module. Input to this point is the pH measured by the point PHI101, which converts pH into excess base concentration. An AM-CL block connected to this point is used for this purpose which also implements the PID control law. The output from this point goes to the base pump. The AM-CL code is at the end of the write-up.

**PHIC102** This is a regulatory point in the Application Module that implements a standard PID algorithm using the default TDC algorithm. This point uses as the control error the difference in pH values without accounting for the nonlinearity in the system. The sole purpose of this point is comparison with the other controller point.

**CI201** A numeric point in the network interface module, it converts the pH into concentration using the algorithm NET>MKS>CI.CL connected to custom point PHCI.
### Figure K.4: Operating Group for the pH reactor

<table>
<thead>
<tr>
<th>GROUP 002</th>
<th>PH CONTROL</th>
</tr>
</thead>
<tbody>
<tr>
<td>100%</td>
<td></td>
</tr>
<tr>
<td>75%</td>
<td></td>
</tr>
<tr>
<td>50%</td>
<td></td>
</tr>
<tr>
<td>25%</td>
<td></td>
</tr>
<tr>
<td>0%</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>PHIC101L</th>
<th>PHIC102SP</th>
<th>PHIC102P</th>
<th>PHIC102P</th>
<th>PHIC102S</th>
<th>PHIC102P</th>
<th>PHIC102SP</th>
<th>PHIC102P</th>
<th>PHIC102P</th>
<th>PHIC102P</th>
</tr>
</thead>
<tbody>
<tr>
<td>PH</td>
<td>PH</td>
<td>PH</td>
<td>PH</td>
<td>PH</td>
<td>PH</td>
<td>PH</td>
<td>PH</td>
<td>PH</td>
<td>PH</td>
</tr>
<tr>
<td>PHICONTL</td>
<td>PHSP</td>
<td>PHSP</td>
<td>PHSP</td>
<td>PHSP</td>
<td>PHSP</td>
<td>PHSP</td>
<td>PHSP</td>
<td>PHSP</td>
<td>PHSP</td>
</tr>
<tr>
<td>CONTROL</td>
<td>FLOWATE</td>
<td>FLOWATE</td>
<td>FLOWATE</td>
<td>FLOWATE</td>
<td>FLOWATE</td>
<td>FLOWATE</td>
<td>FLOWATE</td>
<td>FLOWATE</td>
<td>FLOWATE</td>
</tr>
<tr>
<td>10.1/min</td>
<td>10/1/min</td>
<td>10/1/min</td>
<td>10/1/min</td>
<td>10/1/min</td>
<td>10/1/min</td>
<td>10/1/min</td>
<td>10/1/min</td>
<td>10/1/min</td>
<td>10/1/min</td>
</tr>
<tr>
<td>1001/1</td>
<td>1001/1</td>
<td>1001/1</td>
<td>1001/1</td>
<td>1001/1</td>
<td>1001/1</td>
<td>1001/1</td>
<td>1001/1</td>
<td>1001/1</td>
<td>1001/1</td>
</tr>
<tr>
<td>50</td>
<td>7.0</td>
<td>7.0</td>
<td>7.0</td>
<td>7.0</td>
<td>7.0</td>
<td>7.0</td>
<td>7.0</td>
<td>7.0</td>
<td>7.0</td>
</tr>
<tr>
<td>50</td>
<td>6.99</td>
<td>7.0</td>
<td>6.99</td>
<td>7.22</td>
<td>20.0</td>
<td>-10.0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>25</td>
<td>75.0%</td>
<td>-6.9L</td>
<td>-6.9L</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

INIT
- MAN
- CAS

### Figure K.5: Operating Group for the pH reactor

<table>
<thead>
<tr>
<th>GROUP 003</th>
<th>PH CONTROL 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>100%</td>
<td></td>
</tr>
<tr>
<td>75%</td>
<td></td>
</tr>
<tr>
<td>50%</td>
<td></td>
</tr>
<tr>
<td>25%</td>
<td></td>
</tr>
<tr>
<td>0%</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>PHIC101L</th>
<th>PHIC102SP</th>
<th>PHIC102P</th>
<th>PHIC102P</th>
<th>PHIC102P</th>
</tr>
</thead>
<tbody>
<tr>
<td>PH</td>
<td>PH</td>
<td>PH</td>
<td>PH</td>
<td>PH</td>
</tr>
<tr>
<td>PHICONTL</td>
<td>PHSP</td>
<td>PHSP</td>
<td>PHSP</td>
<td>PHSP</td>
</tr>
<tr>
<td>CONTROL</td>
<td>FLOWATE</td>
<td>FLOWATE</td>
<td>FLOWATE</td>
<td>FLOWATE</td>
</tr>
<tr>
<td>10.1/min</td>
<td>10/1/min</td>
<td>10/1/min</td>
<td>10/1/min</td>
<td>10/1/min</td>
</tr>
<tr>
<td>1001/1</td>
<td>1001/1</td>
<td>1001/1</td>
<td>1001/1</td>
<td>1001/1</td>
</tr>
<tr>
<td>50</td>
<td>-6.8</td>
<td>-6.8</td>
<td></td>
<td></td>
</tr>
<tr>
<td>25</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0%</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

MAN
- CAS
Procedure

This experiment is to be carried out in one session. The sequence of steps to follow is:

1. Calibration of the acid and base pumps
2. Preparation of the acid and base solution
3. Modelling the system dynamics
4. Controller design and implementation

Calibration of the acid and base pumps

Fill the acid and base tanks with water and access Group 3 in the Universal Station. Put the system points PHMA and PHMB in MANUAL. Turn pump A on set the output of PHMA to negative five percent. Using the graduating cylinder and a stop watch, compute the flow rate for that particular output percentage. Repeat this process for 0, 10, 25, 50, 75, and 100 % output values. Use MATLAB to plot the output percentages against flowrate - the function POLYFIT will help you find a polynomial expression relating these two variables. Repeat the procedure for pump B using PHMB. These equations must be incorporated into an AM-Control Language program called PUMP. This is done by entering the Command Processor and typing

ED NET>MKS>PUMP.CL

Having made the changes, compile the CL code with the command

CL NET>MKS>PUMP.CL -UL -NX

Inactivate the custom point PUMP from the detail display. Then the following steps are executed in the Command Processor

UNLK NET>MKS>PUMP PUMP

LK NET>MKS>PUMP PUMP

Re-enter the detail display and activate the custom point PUMP.

Preparing the acid and base solution

Prepare a caustic solution of 0.001 M from NaOH pellets. Begin by adding 0.4 gm of NaOH pellets to one liter of distilled water. Put 100ml of this solution in a one liter volumetric flask, and fill to the line with distilled water for a final solution of 0.001 M NaOH. Similarly, prepare a 0.001 M HCl solution from a 12.1 N HCl solution. First add 0.83 ml concentrated HCl to one liter water. Then place 100ml of this 0.01 M HCl solution in a 1-liter volumetric flask and fill to the line as before to obtain the final solution.
Modelling the system dynamics

Use step testing to obtain the process model. Set all controllers and pumps in MANUAL. Initially set the outputs of pump A and B to 70 percent and 75 percent respectively. Once the system attains steady state (pH of approximately 4.5), change the output of PHMB to 90 percent. Trend the point CI201 and watch the corresponding change in concentration. Graphically obtain the model parameters of a first-order transfer function for each trend, the trend based on \(x\) (the excess base concentration), and the trend based on pH.

Controller Design and Comparison

Using the process models obtained in the previous step, use the IMC design procedure to determine the controller parameters. Once the values for \(K_c\) and \(T_i\) are known, enter these in the custom page of the controller point PHIC101. Test the parameters for a step setpoint change to the neuralization point.

Similarly, generate controller parameters for PHIC102 based on the pH-measurement based model. Since a standard TDC algorithm is used here, be sure to scale the controller gain appropriately, as discussed in previous labs. Enter these parameters in the control algorithm page and test for setpoint changes.

If time permits, consider performing a load disturbance test on each controller by diluting the concentration on the acid feed tank (the teaching assistant will provide further instructions here).

Lab Report

Make sure in your report that you assess the pros and cons of the “linearizing” strategy for pH control. What benefits are observed in the controlled variable and manipulated variable response and which method displays most sensitivity to disturbances? Comment also on what differences would result if one compared this experiment with the more realistic problem of wastewater treatment, where the nature (and concentration) of the acid/base pairs may vary unpredictable. How would an “adaptive” controller (see SEM page 427) be useful in such a situation.
& -4.9162E-05*(PHMB.OP)**3
END ABFLOW

FILE NET>MKS>CI.CL 18 May 93 13:27 SS

BLOCK CI (POINT PHCI; AT GENERAL)
-- THIS BLOCK CONVERTS PH INTO CONCENTRATION
EXTERNAL PHI101, CI201

SET CI201.PV = 10**(PHI101.PV-8)-10**(6-PHI101.PV)
END CI

This CL is used to accomplish the PV and Control Processing for this experiment.

FILE NET>MKS>PH_CL.CL 18 Apr 93 14:18 SS

PACKAGE
------------------------------------------
CUSTOM PARAMETER K "CONTROLLER GAIN"
PARAMETER T1 "INTEGRAL TIME CONSTANT"
EU "SECONDS"
PARAMETER T2 "DERIVATIVE TIME CONSTANT"
EU "SECONDS"
PARAMETER STP "SAMPLE TIME"
EU "SECONDS"
VALUE 2
PARAMETER RESET_C:LOGICAL "SET ON TO RESTART CONTROLLER"
PARAMETER TRACK_PV:LOGICAL "SET ON TO HAVE PV TRACKING"
PARAMETER KERR "CONTROLLER COEFFICIENT: e(k)"
PARAMETER KERR1 "CONTROLLER COEFFICIENT: e(k-1)"
PARAMETER KERR2 "CONTROLLER COEFFICIENT: e(k-2)"
PARAMETER ER "CURRENT ERROR: e(k)"
PARAMETER ER1 "PREVIOUS ERROR 1:e(k-1)"
PARAMETER ER2 "PREVIOUS ERROR 2:e(k-2)"
PARAMETER MOVE "MOVE: m(k)-m(k-1)"
PARAMETER MOVE1 "MOVE: m(k-1)-m(k-2)"
END CUSTOM
------------------------------------------
BLOCK PHCNTRL (POINT PHIC101; AT CTL_ALG)

LOCAL PVCLC, SPCLC
--SP TRACKS PV WHEN POINT IS IN MANUAL
IF (TRACK_PV = ON AND MODE=MAN) THEN SET SP=PV
IF RESET_C = ON THEN (SET ER, ER1, ER2, MOVE, KERR, KERR1, KERR2 = 0;
& SET RESET_C = OFF)

SET PVCLC = (10**(PV-14) - 10**(-PV))
SET SPCLC = (10**(SP-14) - 10**(-SP))
SET ER = SPCLC - PVCLC

--CALCULATE CONTROL COEFFICIENTS

SET KERR = K*(1 + STP/T1 + T2/STP)
SET KERR1 = -K*(1 + 2*T2/STP)
SET KERR2 = K*T2/(STP)

--CALCULATE CURRENT MOVE
SET MOVE = KERR*ER + KERR1*ER1 + KERR2*ER2

--STORE PREVIOUS ERRORS
SET ER2 = ER1
SET ER1 = ER

--CALCULATE CURRENT OUTPUT
SET CV = OPEU + MOVE

END PHCNTRL
END PACKAGE
Bibliography


Graphic Display Guidelines

Overview

Accessing the Picture Editor:

- Selecting Picture Editor target located on Engineering Personality Main Menu display.
- Select Command Processor from Main Menu display. Type PE on command line. This loads the Picture Editor overlay.

Exiting the Picture Editor:

| CTRL and [HELP], or typing in END, returns you to the Engineering Main Menu. |
| Press the [ESC] key. You are now in the Command Processor. When done, press [CTRL] and [HELP] to return to Picture Editor. |

1. To modify an existing graphic, type READ XXXXX where XXXXX is the schematic name. If the schematic exists on a volume other than where you have set path, then you must type in READ and the full pathname like NET> VOL> XXXXX.

2. To save the graphic, type WRITE to save under the currently displayed pathname or type WRITE NET> VOL> XXXXX to write to a different file.

3. To use all available screen space - type SET ROLL 0 2 - this rolls the screen two spaces down and gives you access to the screen located under the top command line. To get to the bottom, type SET ROLL 0 0.

Elements of the Display

Shapes(Lines, Solids)
- Generally portray the process.
• Considered background information.

• Any two dimensional shape can be created.

• Two Types:
  – Hollow (outlined).
  – Solid (fully colored).

Commands: ADD LINE
           ADD SOLID

Behavior

• Given to all shapes, lines, and text on display.

• Four aspects:
  – Color
  – Intensity
  – Blink
  – Normal/Reverse video background.

• Two types:
  – Fixed (remains the same under all conditions).
  – Conditional (changes according to user-specified conditions).

• Command: ADD BEHAVIOR

Values

• Represents measured quantities or conditions in the process.

• Use “live” numbers or status conditions (OFF, TIMEOUT, etc.).

• Usually accomplished by labels.

• Command: ADD VALUE
Variants

• Used to vary items for display on the screen.
• Cause text or picture to vary with process conditions.
• Allow three possible choices:
  – One of several subpictures (e.g., hollow pump vs. solid pump).
  – One of several text strings.
  – One of several subpictures, or one of several text strings, or blank screen.
• Command: ADD VARIANT

Subpictures

• May consist of any single or multiple display elements.
• Drawn once and used repeatedly in same or multiple displays (e.g., valves or pumps).
• Subpictures must have an Origin.
• Copies can be enlarged, reduced, and reversed horizontally or vertically, but not rotated.
• Each copy stored as a different subpicture.
• Its use saves time for both designer and data-entry person.
• Subpicture library available.
• Set path to the Media and Volume of the subpicture when subpictures are to be accessed.
• May be parameterized to create a general subpicture that does not reference specific tagnames (e.g., “&A” for tagname).
• Behavior may be:
  – Fixed (subpicture always added to display with same behavior).
  – Inherited (assumes current behavior of the display it is added to).
• Command: ADD SUB XXXXXX (where XXXXXX is the subpicture name).

Note: The subpicture CIRCLE and the subpicture QUARTER are part of the standard TDC3000 software.
Change Zones

- One of the available library subpictures.
- Allows operator to call up a control-panel-type display.
- Used for analog, digital, or process-module points.
- Standard change zones - 80 columns wide, three lines high.
- Normally called up on last three lines of display.
- Change zones called up by:
  - Configurable button.
  - Touch or cursor.
- Touch or cursor area placed in display, but not in change zone area.
- Command: ADD SUB CHG_ZONE

Note: The subpicture CHG_ZONE resides on volume &DSY.

Targets

- Designate areas on screen that can be activated by touch or cursor.
- Target selection causes some user specified action to occur.
- Common uses of a target:
  - Call up displays and provide screen management.
  - Simulate keyboard buttons.
  - Allow operator to enter values and/or store values in local display data base or system database.
  - retrieve data from specified data base.
    * Check to see if input values fall within specified limits.
    * Call up change zone.
- Three types of target shapes.
  - Solids: Appear as filled rectangles
  - Boxes: Appear as hollow rectangles
  - Invisible: Can be placed over an item on the screen without obscuring it
- Command: ADD TARGET
COLOR AND BEHAVIOR SELECTION

Half Intensity
- Produces a more acceptable display by lowering status light output.
- Best suited for background information.

Reverse Video Background
- Provides contrasting background for text.
- Draws attention
- Works well in a table of data for quickly drawing attention to desired item.

Blink
- Calls attention.
- Aids in searches.
- Should be used with only small objects in one part of the display at a time.
- Consider using black underline which changes to a bright color and blinks on alarm.

Color
- Color is useful to:
  - Separate different groupings shown on the same display.
  - Call attention to an item.
  - Indicate status.
- Use already familiar color conventions with black backgrounds.
- Consider ambient light and color when determining the appropriate intensity.
- Pick a nominal color for graphic and text.
- Use brighter colors only for highlight, emphasis, or attention getting.
- Use calm, low-emotion color for background.
- Overuse of color, blink, intensity, and reverse background degrades display.
- When two items of different colors overlap, only the higher-priority color shows.
- Recommended color scheme:
### PriorityColor Common Use

<table>
<thead>
<tr>
<th>Priority</th>
<th>Color</th>
<th>Common Use</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>White</td>
<td>Intermediate condition between on or off, open or closed, etc.</td>
</tr>
<tr>
<td>2</td>
<td>Cyan</td>
<td>Essential text, values, hardware failure indication, messages.</td>
</tr>
<tr>
<td>3</td>
<td>Magenta</td>
<td>Danger, immediate attention required (power industry).</td>
</tr>
<tr>
<td>4</td>
<td>Blue</td>
<td>Labels, nonessential data.</td>
</tr>
<tr>
<td>5</td>
<td>Yellow</td>
<td>Caution, attention required, abnormal condition, analog alarm, output values.</td>
</tr>
<tr>
<td>6</td>
<td>Green</td>
<td>Status: OFF, de-energized, closed, normal (power industry); ON, energized (others).</td>
</tr>
<tr>
<td>7</td>
<td>Red</td>
<td>Status ON, energized, open, bypassed, digital alarm (power industry); Danger (others).</td>
</tr>
<tr>
<td>8</td>
<td>Black</td>
<td>Background</td>
</tr>
</tbody>
</table>

### Terms Used

#### Expression

- Evaluates some value used with:
  - Conditional behaviors
  - Values
  - Variants
  - Bar charts

- Usually one of the two types:
  - Single entity such as a point reference (e.g., A100.PV)
  - Combination of entities and literal values (e.g., A100.PV + 3.0)

#### Variable type and format used

- Must be specified if not yet known to the system.
- Sensor inputs and calculations derived from them are real values.
- Only the two major format types are explained here:
  - Real (default format = R-ZZZZ9.99).
  - Self-defined enumerations (default format=TEXTL1:10)

- Format key:
Real:

R Real format - fixed decimal point.

F Real format - decimal point floats to the right if magnitude of number requires more digit positions than is specified in format.

+ Either “+” or “−” will be printed; sign floats if digit is zero and more than one “+” is indicated.

− A “−” will be printed if the number is negative; sign floats if digit is zero and more than one “−” is indicated.

L If digit is zero, its position nulled by left-justification.

Z If digit is zero, its position is blanked.

9 Represents a digit position.

. Position of a decimal point.

Text:

Text Text format: L-left justify, R-right justify, C-center justify.

: Separator between field specifiers. Starting character position: Width of output field (e.g., “2:8” = string characters 2–9 will be printed).

Parameter, Prompt, and Response

• Used for subpictures with values and conditional behaviors.

• Allows subpictures to contain greater complexity and versatility.

• Each time a parametrized subpicture is added to display:
  
  – PROMPT question appears on the screen.
  
  – User must give RESPONSE to prompt.
  
  – Specified data is substituted for the PARAMETER.

• PARAMETER denoted by “&” followed by identifier (e.g., &A.PRESS).

• Must specify variable type and format when using PARAMETERS.
Action

- Specified what happens when target is activated.
- Consists of one or more actors that specify desired action.
- Action format is typically actor name (parameters).
- Many types of actors - most common ones are:

<table>
<thead>
<tr>
<th>Actors</th>
<th>“Action”</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Group display</td>
<td>GROUP(group number, point)</td>
<td>GROUP(1,6)</td>
</tr>
<tr>
<td>Change zones</td>
<td>CHG_ZONE(point name)</td>
<td>CHG_ZONE(A100)</td>
</tr>
<tr>
<td>Detail display</td>
<td>DETAIL(point name)</td>
<td>DETAIL(HG0501)</td>
</tr>
<tr>
<td>Overview display</td>
<td>OVERVIEW</td>
<td>OVERVIEW</td>
</tr>
<tr>
<td>Schematic</td>
<td>SCHEM(”schematic name”)</td>
<td>SCHEM(”LOGO”)</td>
</tr>
<tr>
<td>Trend overview display</td>
<td>TRENDOVW</td>
<td>TRENDOVW</td>
</tr>
<tr>
<td>Cross screen</td>
<td>CROSSCRN(screen number)</td>
<td>CROSSCRN(02)</td>
</tr>
</tbody>
</table>