Old Dominion University Research Foundation

DEPARTMENT OF MECHANICAL ENGINEERING AND MECHANICS
COLLEGE OF ENGINEERING AND TECHNOLOGY
OLD DOMINION UNIVERSITY
NORFOLK, VIRGINIA 23529

EXPERT SYSTEMS FOR AUTOMATED MAINTENANCE OF A MARS OXYGEN PRODUCTION SYSTEM

By

Robert L. Ash, Principal Investigator

Jen-Kuang Huang, Director

and Ming-Tsang Ho, Graduate Research Associate

Final Report
For the period ending November 30, 1989

Prepared for the
National Aeronautics and Space Administration
Langley Research Center
Hampton, Virginia 23665-5225

Under
NASA Grant NAG-9-308
David McKay, Technical Officer

October 1989

Old Dominion University Research Foundation is a not-for-profit corporation closely affiliated with Old Dominion University and serves as the University's fiscal and administrative agent for sponsored programs.

Any questions or comments concerning the material contained in this report should be addressed to:

Executive Director
Old Dominion University Research Foundation
P. O. Box 6369
Norfolk, Virginia 23508-0369

Telephone: (804) 683-4293
Fax Number: (804) 683-5290

DEPARTMENT OF MECHANICAL ENGINEERING AND MECHANICS
COLLEGE OF ENGINEERING AND TECHNOLOGY
OLD DOMINION UNIVERSITY
NORFOLK, VIRGINIA 23529

EXPERT SYSTEMS FOR AUTOMATED MAINTENANCE OF A MARS OXYGEN PRODUCTION SYSTEM

By

Robert L. Ash, Principal Investigator

Jen-Kuang Huang, Director

and Ming-Tsang Ho, Graduate Research Associate

Final Report
For the period ending November 30, 1989

Prepared for the
National Aeronautics and Space Administration
Langley Research Center
Hampton, Virginia 23665-5225

Under
NASA Grant NAG-9-308
David McKay, Technical Officer

Submitted by the
Old Dominion University Research Foundation
P.O. Box 6369
Norfolk, Virginia 23508-0369

October 1989

ABSTRACT

Expert Systems for Automated Maintenance
of a Mars Oxygen Production System

Ming-Tsang Ho
Old Dominion University, 1989
Director: Dr. Jen-Kuang Huang

A prototype expert system has been developed for maintaining autonomous operation of a Mars oxygen production system. Normal operation conditions and failure modes according to certain desired criteria are tested and identified. Several schemes for failure detection and isolation using forward chaining, backward chaining, knowledge-based and rule-based are devised to perform several housekeeping functions. These functions include self-health checkout, an emergency shut down program, fault detection and conventional control activities. An effort has been made to derive the dynamic model of the system using Bond-Graph technique in order to develop the model-based failure detection and isolation scheme by estimation method. Finally, computer simulations and experimental results have demonstrated the feasibility of the expert system and a preliminary reliability analysis for the oxygen production system is also provided.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Chapter 1

Introduction

Future space missions will return humans to the moon as well as place them on Mars [1]. Unlike the Apollo program, these missions will be used in a building block approach which will enable man to ultimately live on those planetary bodies for indefinite periods of time. However, the goal of unlimited stay time has a profound impact on mission design. That is, it is neither desirable nor practical to resupply extra terrestrial outposts with consumables from earth. Research on continuous life support systems is far ahead of other elements at this time, but the fully closed ecological system approach must be augmented. There are too many ways for a closed ecology to become disrupted and fail, requiring emergency resupply of basic items ranging from oxygen to water and food.

Unlike the scientific aspects of an extraterrestrial research station, the production of commodities from in situ resources will involve continuous or repetitive operations that will likely be boring to a human. Furthermore, the cost per hour for a human operator in space is staggering. On the other hand, production of oxygen for fuel or for life support may be vital for survival and these systems must be extremely reliable and robust. The research which is reported here is directed toward estabilishing the framework and methodology required to operate a mission critical system autonomously. The use of Martian atmosphere as a feedstock for the production of oxygen has been under investigation for some time [2-5]. Since Mars atmospheric composition is known accurately [6] (also shown in Table 1.1) and is not sensitive to landing site, it is an ideal feedstock for early missions using in situ resources. Furthermore, the yttria stabilized zirconia cell that can be used to remove oxygen from heated Mars atmosphere has been demonstrated [7]. However, in order for the technology to become accepted as a viable option for supporting early missions to Mars, it is necessary to demonstrate that these systems can be operated reliably for long periods of time with minimum human intervention.

1.1 Objective and Motivation

A research effort has been underway at Old Dominion University which is directed toward estabilishing the reliability of a Mars oxygen separation system. While Mars atmosphere must be extracted and oxygen must be liquified and stored, a variety of hardware elements can be used between atmospheric collection and storage [4, 5 and 8]. However, the oxygen extraction process using heated Mars atmosphere and a zirconia based electrochemical cell has been identified as the preferred oxygen separation element. Since pumps, compressors, refrigerators and power generating equipment are accepted elements in processing plants and since a multiplicity of options are available for each element, our research has focused only on developing a well controlled and reliable oxygen separation system which can be operated as an endurance test bed.

It is important to recognize that utilization of in situ resources requires as much research in expert system/autonomous control as in development of simple, efficient and reliable chemical processes. As new resources and products are identified, new processes can be studied. However, the operation and control methodology, which is similar to many life support applications can be developed now.

In addition to space systems, expert systems technology has been applied extensively to many fields, including chemical plants and power plants in the past ten years [9,10]. Expert systems can deliver human expertise to provide several functions like maintenance, operation, prediction, control and design, and enhance the performance and reliability of the processing unit or control system. However, in a recent survey, San Giovanni and Romans [9] showed that over 95% of the expert systems applied to the chemical plants are less than two years old and over 90% of the applications relied on shells, not languages. This indicates that expert systems technology is still in its infant stage and more experienced knowledge-based maintenance and more flexible tools are required. This survey also reported that most of the expert systems (54%) were used for system diagnosis and prescription. In addition to chemical plants, automated maintenance by expert systems in a variety of terrestrial applications are also been studied actively [11,12].

Concerning the maintenance for any dynamic system, one of the main problems is system failure detection and isolation (FDI). Various hardware mechanisms and software algorithms providing FDI capability have been proposed [10,13-19]. Determination of failure threshold which is a common problem to all FDI designs has been studied [15,20,21]. When choosing failure thresholds, two possible decision errors must be considered: false alarm and missed failures. The probabilities of these two errors are functions of failure thresholds. Markov theory [20] can be applied to determine time varying thresholds which cannot be solved by classical detection theory. Besides the threshold determination, another important topic is the FDI algorithm for linear stochastic systems [17,18]. The methods include the design of specific failure-sensitive filters, statistical tests on filter innovations and development of jump process formulations. Applications of these methods to chemical plants need further studies.

The current trend in space system design usually includes redundant elements in critical areas to enhance the FDI capability.

However, expert systems will be more complicated. In addition to hardware redundancy, analytical redundancy is an alternative approach which utilizes the analytical relationships between different elements in the system to detect and identify the component failure, so that system performance can be improved without increasing hardware elements. This technique will be used in our expert system.

The objective of this work is to develop a preliminary expert system to provide the automation of housekeeping functions such as self-health checkout, emergency shutdown, failure detection and isolation and active control action for the Mars oxygen production system.

1.2 Thesis Outline

In Chap. 2, the hardware set-up of the Mars Oxygen Production System shown in Fig. 1.1 and 1.2 is described in detail. The system is divided into four subsystems which are Intake System, Oven and Cell Unit, Vacuum System and Data Acquisition System. The function of each subsystem is presented and experimental results are provided.

In Chap. 3, four kinds of Expert System approaches for failure detection and isolation schemes using forward chaining ,pattern matching, application software and PROLOG language are introduced. Failure modes are determined based on the experimental data.

In Chap. 4, a process fault detection and isolation scheme based on modeling and estimation for the system is introduced. An effort has been made to work out the dynamic model of the system using the Bond-Graph technique. Comparison between the numerical simulations of the model and the experimental results is given in Chap. 5.

In Chap. 5, several experimental results are presented to demonstrate the feasibility of these approaches.

In Chap. 6, a preliminary study of the reliability analysis of the Mars Oxygen Production system is discussed. Finally, conclusions and some remarks about the future works are given in Chap. 7.

Fig. 1.1 The Prototype of The "Mars Machine"

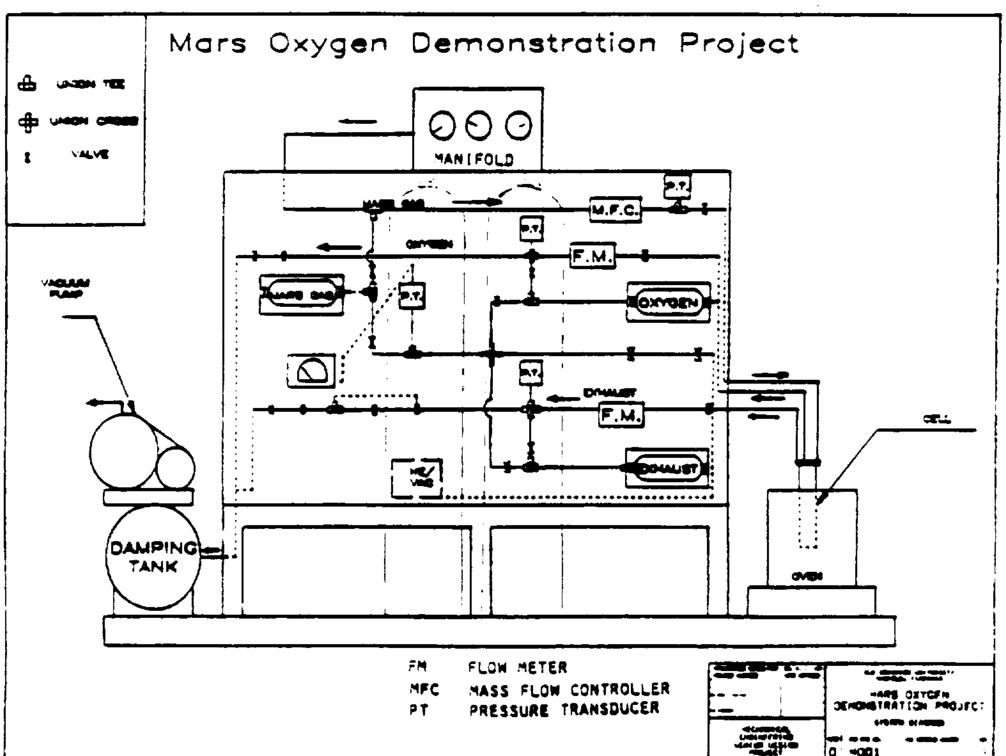


Fig. 1.2 Schematic Diagram of The "Mars Machine"

Table 1.1

Nominal and Simulated Martian Atmosphere Compositions

Component	Percent by Volume	
	Nominal* (Mars Gas)	Simulated
CO_2	95.32	95.32
N_2	2.7	2.7
A_r	1.6	1.78
O_2	0.13	0.13
CO	0.07	0.07
Trace†	Balance	-------

* Reference 2

† H_2O, 0.03%; N_e, 2.5 ppm; K_r, 0.3 ppm; X_e, 0.08 ppm; O_3, 0.03 ppm.

Chapter 2

The Mars Oxygen Production System

The Mars oxygen production system [24] is shown schematically in Fig. 2.1. A continuous supply of simulated Mars atmosphere (95.32% CO_2, 2.7% N_2, 1.78% Ar, 0.13% O_2 and 0.07% CO by volume), is provided to the cell via gas storage bottles and a manifold system. The mass flow rate is controlled by a mass flow controller and the cell is operated at a pressure of 50-120 mb which is a nominal operating pressure for the Mars system. The gas and cell are maintained at a temperature between 1000 K and 1200 K. Oxygen is extracted from the Mars gas at a pressure between 10 mb and 120 mb where the specific collection pressure will be determined as part of an optimization study. (It is desirable to collect oxygen at a pressure above Mars ambient so that leaks do not cause collected oxygen to become contaminated with Mars atmosphere.) Both the exhaust gas and oxygen, leaving the cell, are monitored through separate mass flow meters before being recombined in a ballast tank and expelled to the atmosphere through a vacuum pump. Additional system components include an oven for heating the cell and heat exchangers for protecting downstream flow measuring devices. Feedgas, exhaust gas and collected oxygen can be sampled to verify that the system is operated properly. The operating and collection pressures are regulated through a series of manual valves which will be tuned prior to endurance testing. Temperatures are measured using chromel/alumel and copper/constantan thermocouples and those data are supplied to an IBM PC/AT computer through a Keithley 500 data acquisition system. In addition, mass flow rates, cell voltage and pressure measurements are provided to the computer through the data acquisition system. The computer can control the feedgas mass flow rate as well as shut down the power supplies for the cell and for the oven. The hardware setup of each subsystem will be described in the following sections.

2.1 Intake System

The function of the Mars Gas Intake System is to supply the simulated Mars Gas to the zirconia cell from a pair K-sized bottles (about 2 cubic feet). The components of the Intake System include two K-sized bottles, pressure transducers, a manifold system and a mass flow meter/controller. The two bottles are pressurized with the Mars Gas and are connected to an automatic change-over manifold system. Pressure in each bottle is monitored using a pair of pressure transducers (Omega Engineering, Model PX300-1KGV). The manifold system (Union Carbide System SG 9150) is used to provide a continuous supply of feed gas for endurance testing. When operating, one bottle will be active while the other is on reserve. When the regulator of the manifold system senses a delivery pressure drop of 3 to 5 psi, it automatically initiates flow through the reserve bottle. The empty bottle may be replaced and a new bottle can be installed. The replenished bottle then becomes the new reserve bottle. The feed gas is throttled from a high bottle pressure of between 100 and 1000 psi to a pressure of approximately 40 psig before it is introduced into the system pipeline.

A mass flow controller (Omega Engineering, Model FMA-766-V) is used to control the flow of feed gas. A set point voltage for the flow controller between 0 and 5V can be issued from the analog output modules of Keithley 500 data acquisition system to maintain a corresponding flow rate set point of between 0 and 370 standard cubic centimeters per minute (sccm). The flow controller hook-up is shown schematically in Fig. 2.3. The feed-gas pressure is monitored using a pressure transducer (Omega Engineering, Model PX241- 15NG).

2.2 Oven and Cell Assembly

The function of the oven and cell assembly is to generate oxygen from the carbon dioxide rich Mars Gas. Both the oven and the zirconia cell were manufactured by Ceramatec Corporation, Salt Lake City, Utah and a schematic of the cell assembly is shown in Fig. 2.4. The cell assembly is heated to a temperature between 1000K and 1200K by the oven. The temperature is controlled and maintained at a constant temperature by using an ON/OFF controller (Omega Engineering, Model 6100). The controlled feed gas is introduced into the cell via the feed gas inlet pipe. By the solid electrolyte process [7] with approximately 1.0 V potential across the zirconia tubular membrane and at a temperature between 1000K and 1200K, oxygen, which has been dissociated from carbon dioxide can be pumped across the membrane. Molecular oxygen will be ionized on the porous cathode, moved through the zirconia tubular membrane, and recombined into oxygen molecules on the porous anode which is inside the zirconia tubular membrane. Separate oxygen and exhaust gas flows leave the cell via the oxygen outlet and exhaust outlet pipelines which are connected to a vacuum system. The pressures of feed-gas and oxygen are monitored using two pressure transducers (Datametrics Type 600) and the temperature in the cell is measured using K-type thermocouples.

The basic relation among the required dc voltage, the current generated, and some system parameters is as following [25] :

$$V_t = E(Fco_2, P1o_2, P2o_2, T_c, s) + E_o + (R_s + R_z)I \quad (2.2\text{-}1)$$

where V_t is the total applied voltage, volts

E is the pressure potential, volts

Fco_2 is the CO_2 flow rate, sccm

$P1o_2$ is the supply partial pressure of O_2, mb

$P2o_2$ is the product partial pressure of O_2, mb

T_c is the operating temperature, K

s is the flow resistance

E_o is the ionization overpotential, volts

R_s is the surface resistance, ohms

R_z is the membrane resistance, ohms

I is the current generated, Amp

The relationship for the pressure potential was found by experimental evaluation to be (Nernst equation) [7]:

$$E = \frac{R\,T_c}{z\,F} \ln \frac{P2o_2}{P1o_2} \qquad (2.2\text{-}2)$$

where R is the gas constant (8.31441 J/(gm-mole K))

z is the equivalent mole (equal to 4 for O_2)

F is the Faraday constant (96484.56 Amp.S/mole)

The Conversion Efficiency h can be expressed as following :

For feed-gas : F_1 (gm-mole/s) = F_1' (gm/hr)/(44.01.3600)

For oxygen : F_2 (gm-mole/s) = I_z (Amps)/(4.96484.56)

Conversion Efficiency : $h = 2\ F_2 / F_1$ (2.2-3)

2.3 Vacuum System

The function of the Vacuum System is to provide a pressure sink so that the Mars gas will flow downstream to a ballast tank and the system pressure can be maintained below one atmosphere. The components of the Vacuum System include the pipelines for oxygen and exhaust gas which are connected to a 45.3 liter ballast tank and a vacuum pump (Gast, Model 5BA-1). Both the oxygen and exhaust gases are combined in the ballast tank and pumped back to the atmosphere. The flow of oxygen and exhaust gas is monitored using mass flow meters (Omega Engineering, Model FMA-863-V for oxygen and Model FMA-866-V for exhaust gas). The pressures in the oxygen and exhaust gas lines and the pressure inside the ballast tank are all monitored using pressure transducers (Omega Engineering, Model 241-15NG) and can be regulated by adjusting two metering valves (Vo and Vw which are items 10 and 12 in Fig. 2.1).

2.4 Data Acquisition and Control Systems

All the outputs of the sensors including pressure transducers, flow meters and thermocouples can be acquired by the Data Acquisition, Control and Display System (DACAD). Figure 2.2 shows a schematic diagram of this system which is composed of a power supply box, a signal junction box, an alarm control box, a Keithley 500 data acquisition box and an IBM PC AT computer with a color monitor (and an Epson FX-85 printer). The power supply box is used to provide two kinds of dc voltages (+10v and +/-15v) for excitation of sensors. Table 2.1 lists the capacity of the power supply box and the power required for each sensor. If an expanded set of sensors is needed in the future, it will be necessary to make sure that the overall power consumption by all sensors does not exceed the maximum allowable power output of the power supply. All the sensor outputs are connected by shielded cables to the signal junction box where each signal is routed to a specified channel of Analog Input Module (AIM2) in the Keithley 500 box. Table 2.2 lists the color codes for different channels.

The Keithley 500 data acquisition unit has a 12-bit converter and is capable of analog or digital I/O. Seven optional modules with different I/O functions are configured and installed in the unit (see Table 2.3). Figure 2.5a and 2.5b are the circuit diagrams which show the connection pin codes among the sensors, the power supply connector, the signal connector, and the modules. The Keithley 500 unit is connected to the IBM PC AT which served as the data acquisition host as well as the control system manager. The Keithley 500 unit employs Soft 500 software which is compatible with BASIC to facilitate data acquisition and control programming.

The general formulas for determining the relationship between signal output of the flow meters and mass flow are as follows :

For F_1 and F_3 : $F_{gas} = V_{signal}\ 500\ Cf_{gas}/5.0$ (2.4-1)

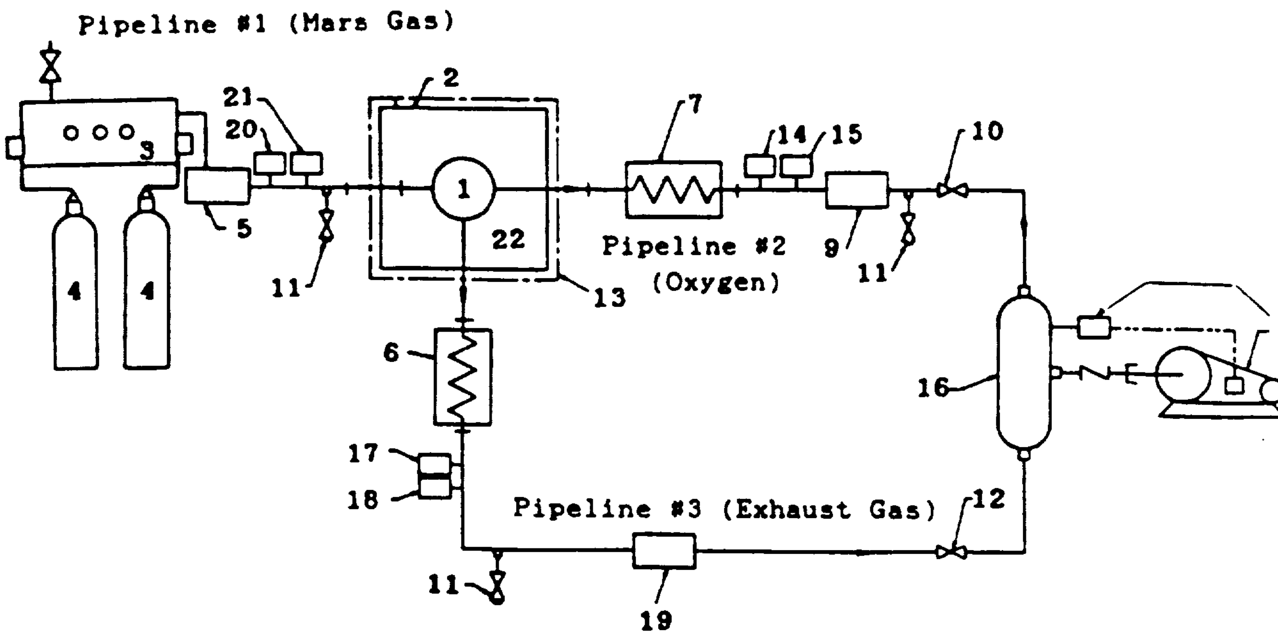
For F_2 : $F_{gas} = V_{signal}\ 10\ Cf_{gas}/5.0$ (2.4-2)

where F_{gas} is the flow of gas to be metered, sccm

V_{signal} is the signal output of flow meters, volts

Cf_{gas} is the conversion factor for the gas to be metered (0.746, 0.765 and 0.996 for CO_2, Mars gas and O_2 respectively)

Note : These values can be found or derived from the operator's manual.



LIST OF COMPONENTS			
ITEM NO.	QTY	DESCRIPTION	SENSOR SYMBOL
1	1	Zirconia Cell/Voltage&Current Sensors	V_z, I_z
2	1	Oven/Thermocouples	T_{c1}, T_{c2}, T_{c3}
3	1	"MARS GAS" Manifold	
4	2	"MARS GAS" Bottles/Pressure Transducer	P_p, P_r
5	1	Flow Control Valve W/ Meter	F_1
6	1	Exhaust Gas Heat Exchanger	
7	1	Oxygen Heat Exchanger	
8	1	Vacuum Pump/Pressure Transducer	P_t
9	1	Oxygen Flow meter	F_2
10	1	Oxygen Outlet Valve	
11	3	Sampling Valve	
12	1	Exhaust Gas Valve	
13	1	Oven Insulation	
14	1	Oxygen Pressure Transducer	P_z, P_2
15	1	Oxygen Thermocouple	T_2
16	1	Damping Tank	
17	1	Exhaust Gas Pressure Transducer	P_3
18	1	Exhaust Gas Thermocouple	T_3
19	1	Exhaust Gas Flow meter	F_3
20	1	"MARS GAS" Pressure Transducer	P_c, P_1
21	1	"MARS GAS" Thermocouple	T_1
22	1	Oven Heating System	V_o, I_o
23	1	Ambient Pressure Transducer(not shown)	P_a

Fig. 2.1 Schematic Diagram of Mars Oxygen Processor System

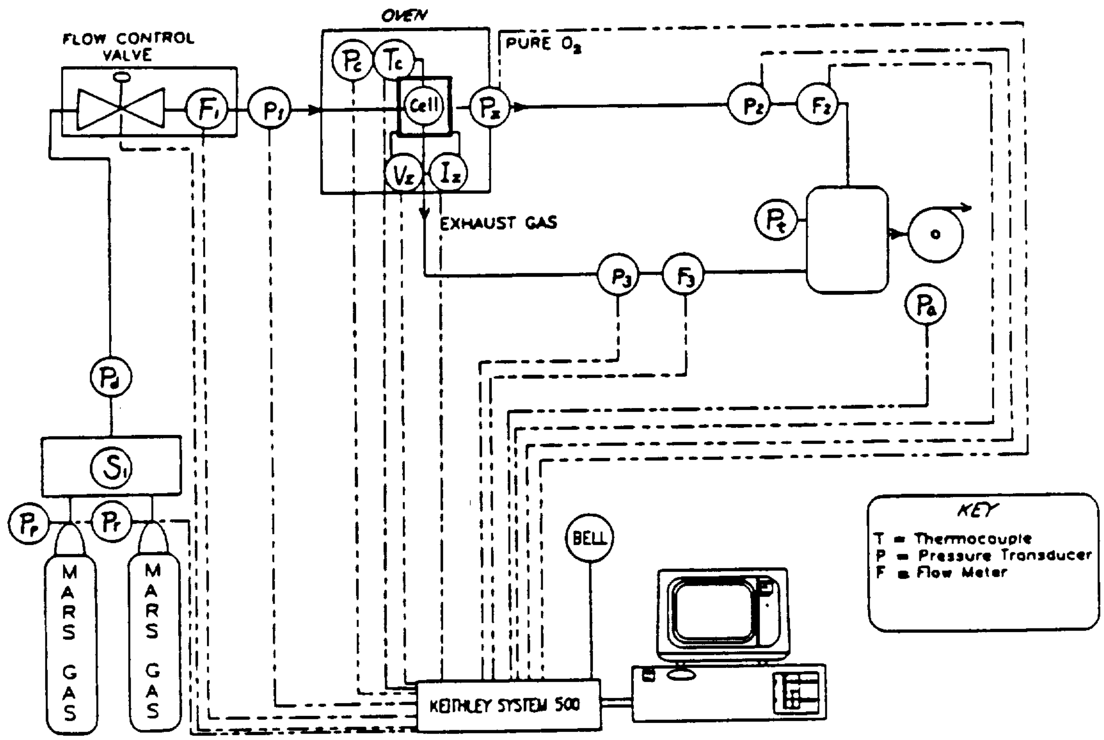


Fig. 2.2 Schematic of The Data Acquisition and Control System

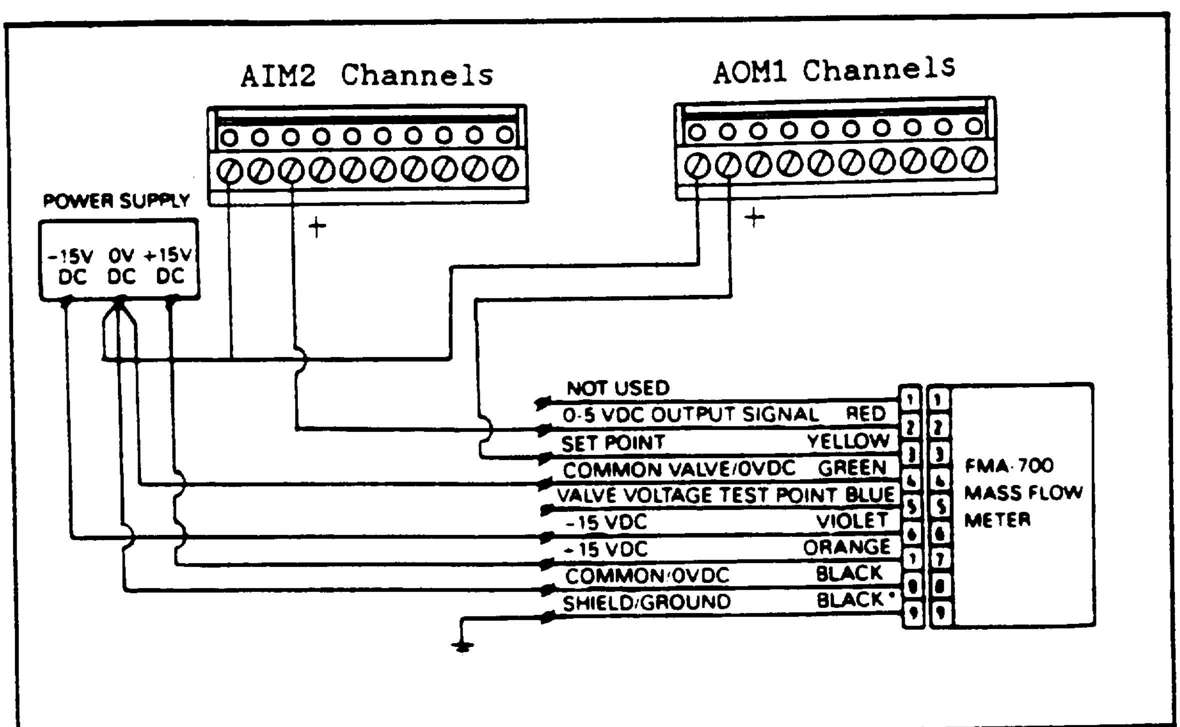


Fig. 2.3 Schematic of The Flow Controller Hook-up

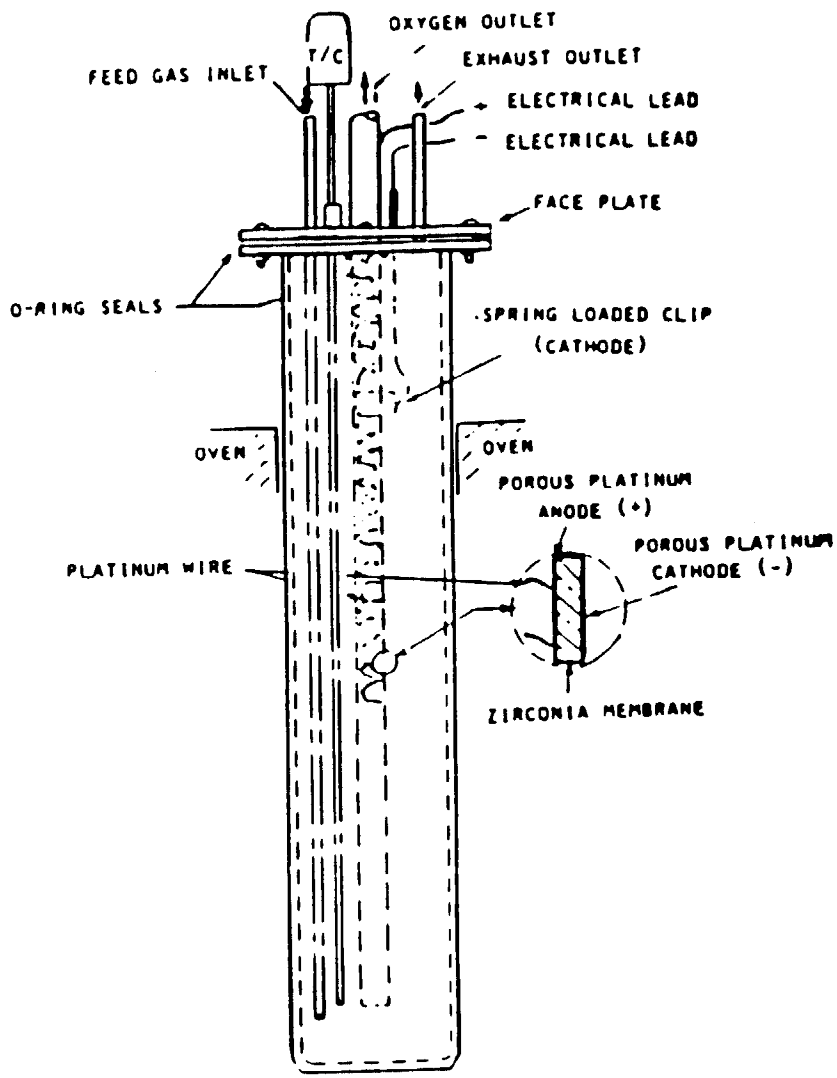


Fig. 2.4a Schematic of The Cell Assembly

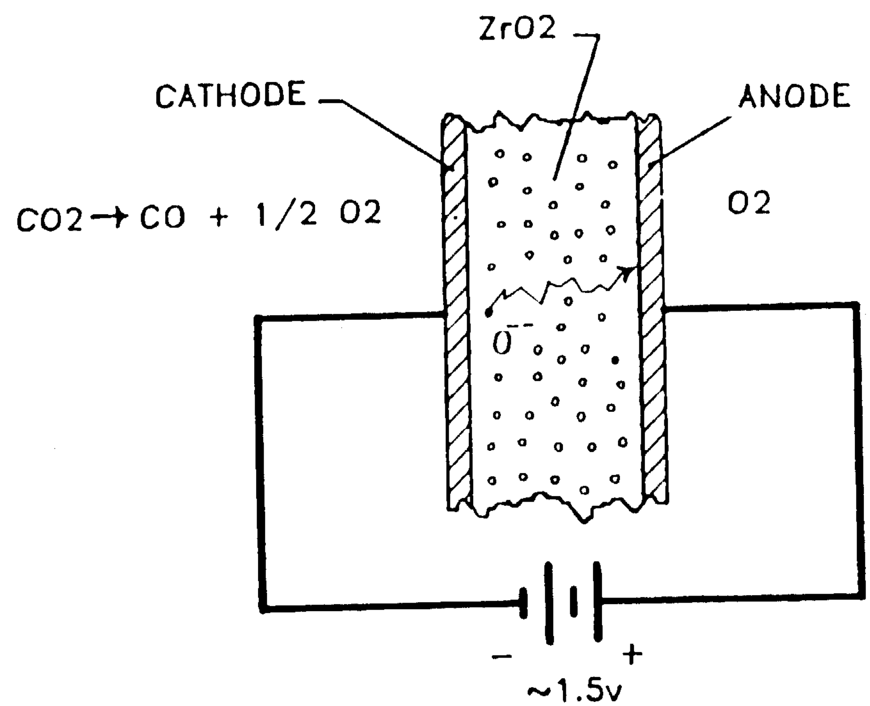


Fig. 2.4b ZrO_2 Cell Schematic

```
Power Side                                       Signal Side
  A   B        C              D                    E           F    G      H
+15V 1-----(R1)-----| Datametrics  |-----(R1)-----1   +    Slot4,CH4
-15V 2-----(B2)-----| Type 600     |-----(B2)-----3   GND
GND  37----(W4)-----|              |-----(S4)-----25  CASE
CASE 19----(S4)-----| Pc           |

+10V 3-----(R1)-----| Omega        |-----(R1)-----4   +    Slot4,CH5
GND  4-----(B2)-----| PX241-15NG   |-----(B2)-----5   GND
CASE 19----         | P1           |        -----25  CASE

+10V 5-----(R1)-----| Omega        |-----(R1)-----6   +    Slot4,CH6
GND  6-----(B2)-----| PX241-15NG   |-----(B2)-----7   GND
CASE 19----         | P3           |        -----25  CASE

+10V 7-----(R1)-----| Omega        |-----(R1)-----8   +    Slot4,CH0
GND  8-----(B2)-----| PX241-15NG   |-----(B2)-----9   GND
CASE 19----         | Pt           |        -----25  CASE

+10V 9-----(R1)-----| Omega        |-----(R1)-----      +    Slot6,CH0
GND  10----(B2)-----| PX300-1KGV   |-----(B2)-----      GND
CASE 19----         | Pp           |        -----      CASE

+10V 23----(R1)-----| Omega        |-----(R3)-----      +    Slot6,CH1
GND  24----(B2)-----| PX300-1KGV   |-----(B4)-----      GND
CASE 19----         | Pr           |        -----      CASE

+10V 11----(R1)-----| Omega        |-----(R1)-----10  +    Slot4,CH7
GND  12----(B2)-----| PX241-15NG   |-----(B2)-----11  GND
CASE 19----         | P2           |        -----25  CASE

+15V 20----(R1)-----| Datametrics  |-----(R1)-----12  +    Slot4,CH8
-15V 21----(B2)-----| Type 600     |-----(B2)-----13  GND
GND  22----(W4)-----|              |-----(S4)-----25  CASE
CASE 19----(S4)-----| Pz           |

+15V 13----(R1)(BLK1| Omega        |-----(R1)-----14  +    Slot4,CH1
-15V 14----(B2)(ORG2| FMA766-V     |        -----15  GND
GND  34----(G3)(W&G3|              |-----(Y3)-----16  S.P.Slot5,CH0
CASE 19----(S4)-----| F1           |-----(S4)-----25  CASE
```

Fig. 2.5a Circuit Diagram of The Data Acquisition System

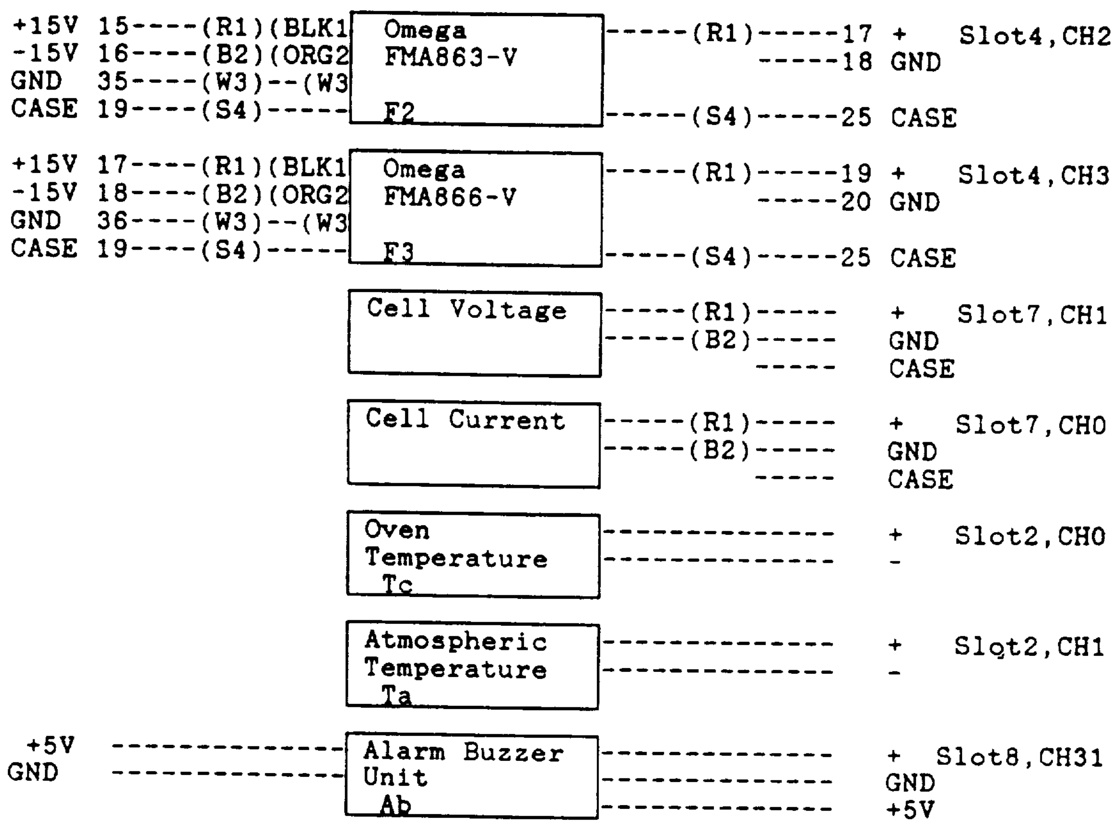


Fig. 2.5b Circuit Diagram of The Data Acquisition System

Table 2.1 Sensor Power Requirements

Power supply PST-10				Power supply PST-B15			
Sensor Model		Power Required V	mA	Sensor Model		Power Required V	mA
P1	Omega PX241-15NG	+10	20	Pc	Datametrics Type 600	+15	30
P2	Omega PX241-15NG	+10	20	Pz	Datametrics Type 600	+15	30
P3	Omega PX241-15NG	+10	20	F1	Omega FMA766-V	+15	250
Pt	Omega PX241-15NG	+10	20	F2	Omega FMA863-V	+15	25
Pp	Omega PX300-1KGV	+10	10	F3	Omega FMA866-V	+15	25
Pr	Omega PX300-1KGV	+10	10				
Total			100				360

Note: The maximum allowable current is 240mA for PST-10 and is 500mA for PST-B15.
The buzzer has one independent power supply PST-5.

Table 2.2 Color Codes for Channels in The Signal Junction Box

PORT A (channel 0-7)		PORT B (channel 8 - 15)	
Channel #	Color Code	Channel #	Color Code
Channel 0	Blue	Channel 8	Blue
Channel 1	Violet	Channel 9	Violet
Channel 2	Green	Channel 10	Green
Channel 3	White	Channel 11	White
Channel 4	Yellow	Channel 12	Yellow
Channel 5	Black	Channel 13	Black
Channel 6	Red	Channel 14	Red
Channel 7	Orange	Channel 15	Orange
Ground	Brown	Ground	Brown

Table 2.3 Configuration of The I/O Modules in the Keithley 500 Box

HARDWARE SETUP

AMM1: 8 channel analog input, 12 bit A/D converter. Slot 1 only.

SLOT	CARD	SWITCH CONFIGURATION	MODULES			
1	AMM1	Range: -10. to 10.V	ADM1	AIM6	AOM3	PCM2
2	AIM7		ADM2	AIM7	AOM4	PIM1
3	NONE		AIM1	AIM8	DIM1	PIM2
4	AIM2		AIM2	AIM9	DIO1	PROT
5	AOM1	R:0)10.B 1)10.B 2)10.B 3)10.B 4)10.B	AIM3	AMM1	DOM1	STP1
6	AIM5	Gain:0)x100 1)x100 2)x100 3)x100	AIM4	AOM1	GPIB	STP2
7	AIM4	Gain:0)x1 1)x1 2)x1 3)x1	AIM5	AOM2	PCM1	NONE
8	DIO1	Port : A) IN B) IN C)OUT D)OUT				
9	NONE					
10	NONE					

F2-FILE F3-MODULE F4-SWITCHES F5-CHANNELS
F9-LIST F10-EXIT TO DOS
Thu May 18 19:41 Path: C:\SOFT500\

Chapter 3

System Failure Detection and Isolation

In order to provide autonomous maintenance for oxygen production for a long duration, a knowledge-based expert system for system failure detection and isolation has been developed. In this chapter, the system operating conditions and parameters are first defined from experimental data. Then, failure modes for a specific set of operating conditions can be defined and rule-based diagnostic and decision making algorithms are developed to handle the necessary housekeeping functions.

3.1 System Operating Conditions and Parameters

The definition of the system operating conditions is based on the experience in construction and operation of the Mars oxygen production system. However, different operating condition sets can be determined according to several kinds of operating criteria such as optimal conversion rate of CO_2 to O_2 or maximum oxygen production which can be obtained from a specified constant feed-gas flow rate. For a set of operating condition definitions, system parameter data including pressure, flow rate, temperature, voltage and current are "collected" from sensors and stored in array in a computer through the data acquisition system. Some of the system parameters can be adjusted each time the system is initialized and hence are called "Independent System Parameters". Though they are not "independent" in the common sense because of a certain degree of correlation existing among them, they can be adjusted to a desired value near the set points. Independent System Parameters include cell temperature (Tc), voltage across the sealed tubular zirconia membrane (Vz), set point of feed-gas mass flow rate (F1), feed-gas delivery pressre (Pd), pressure in the cell (Pc) and pressure of oxygen in the zirconia tubular membrane (Pz). They can be represented by a initial input vector I defined as following :

$$\bar{I} = \begin{bmatrix} i1 \\ i2 \\ i3 \\ i4 \\ i5 \\ i6 \end{bmatrix} = \begin{bmatrix} Tc \\ Vz \\ F1 \\ Pd \\ Pc \\ Pz \end{bmatrix} \qquad (3.1\text{-}1)$$

Note that all pressures shown here are absolute pressures. Pressure readings from the pressure transducers which measure the gage pressure are converted to absolute pressure by adding the atmosphere pressure.

Once the Independent System Parameters have been set, the other system parameters can then be determined by the system characteristic. These "Dependent System Parameters" are pressure in the feed-gas pipeline (P1), pressure in the oxygen pipeline (P2), pressure in the waste-gas pipeline (P3), pressure in the ballast tank (Pt), waste-gas flow rate (F3), current across the sealed tubular zirconia membrane (Iz) and oxygen flow rate (F2). The system initializing procedures are described as followings :

1. Turn on the computer (IBM PC/AT), the data acquisition system (Keithley Series 500), and the power supply box for sensors and wait at least 30 min. for the mass flow meters to be warmed up.

2. Turn on the oven unit and set the temperature controller to the desired temperature set point (Tc). Wait until the oven temperature is stable.
3. Open the valve of the primary feed-gas bottle and adjust delivery pressure (Pd) to the desired value.
4. Turn on the vacuum pump and plug in the power cable for the solenoid valve.
5. Turn on the dc power supply for the zirconia membrane and adjust the voltage across the membrane (Vz) to the desired value.
6. Run the program "MODFDI" and press the F1 key to choose the function on the Real-Time Update Screen. Key in the desired feed-gas flow rate set point (F1) to answer the prompt. Updated real-time system information, including all the sensor outputs will then show up on the screen.
7. Set the cell pressure (Pc) to the desired value by adjusting the metering valve at the waste-gas pipeline (Vw).
8. Set the oxygen pressure (Pz) to the desired value by adjusting the metering valve at the oxygen pipeline (Vo).
9. Monitor the current (Iz) until it is stablized.
10. If the system is in its normal operating condition, terminate the Real-Time Update Screen and start to run the Expert System program for F.D.I. (Failure Detection and Isolation).

Determination of the best initial input vector (I) has been a part of the optimization study. Several tests were performed to determine the suitable initial system parameters and the results are shown in Chapter 5. One of the most important system parameters is the oven temperature (Tc), which is controlled by an oven heating subsystem. In order to make the resistance of the zirconia cell lower than 15 ohm, Tc was determined to be between 1000K and 1200K according to the Temperature-Resistance curve (Fig. 3.1). However, the higher the oven temperature, the lower the allowed maximum voltage which can be applied across the zirconia cell (Vzc) without damaging the cell. The value of the critical voltage has to be determined experimentally.

If the applied voltage (Vz) exceeds the critical potential (Vzc), the oxygen atoms will be pulled out of the zirconia matrix and the cell will deteriorate. The critical potential for a certain temperature can be determined experimentally. For example, set Tc=1200K, Vz=0.8V and keep all other system parameters constant. Observe the current (Iz) for at least 6 hours. If the current remains within a certain constant range without dropping significantly, increase Vz a little and run it for another 6 hours. If the current drops significantly during the 6 hour interval, decrease Vz a little and run it again. By such a repeated process, the critical potential can be found. If the above procedures are performed for several different Tc set points, one would be able to find the best operating condition which yields the maximum current output.

The optimal mass flow set point of feed-gas (F1) can also be determined by experimental testing. Tests which are performed for feed-gas flow rate between 50 and 150 sccm show that if the cell pressures are kept constant, higher feed-gas flow rates can produce higher current. However, the conversion efficiency is higher for lower feed-gas flow rates. Therefore, the mass flow set point for the feed-gas is chosen to be 50 sccm in order to obtain higher conversion efficiency.

The feed-gas delivery pressure is set to be 40 psig according to the specifications of the flow controller. There is a trade-off between increasing and decreasing the pressure ratio of Pc to Pz (Here we define the pressure ratio to be Pc / Pz). Increasing the pressure ratio will decrease the pressure potential (Eq. 2.2-2) so that the current across the zirconia membrane and the oxygen production will be increased. However, if there is leakage somewhere at the membrane, higher pressure ratio will cause the oxygen pipeline to be contaminated by feed-gas faster. Several experimental results are presented in Chap. 5 and some of those are chosen to be the operating condition definitions. The corresponding nominal sensor values and allowable tolerances for each set of operating condition definitions are given in Chap. 5.

3.2 Failure Modes

For each set of operating condition definitions, failure modes for subsystems, sensors and leakage are identified and listed in Tables 3.1, 3.2 and 3.3. The corresponding isolation levels and required control actions issued from expert systems are also provided.

Table 3.1 General System Failure Modes

No.	Failure Mode	Isolation Level	Automatic Action
G-01	Tc Too High	Oven Subsystem	Shutdown Oven
G-02	Tc Too Low	Oven Subsystem	Alarm Signal
G-03	Vz Too High	Power #1 Subsystem	Shutdown Power
G-04	Vz Too Low	Power #1 Subsystem	Alarm Signal
G-05	Vz Unstable	Power #1 Subsystem	Alarm Signal
G-06	Vz Bad Connection or Clip Oxidized	Cell Subsystem	Shutdown System
G-07	Z-Cell Degraded	Cell Subsystem	Shutdown System
G-08	Vz Short Circuit	Cell Subsystem	Shutdown System
G-09	Manifold Switch F	Manifold Subsystem	Alarm Signal
G-10	Pr Too Low	Manifold Subsystem	Alarm Signal
G-11	Pd Too High	Manifold Subsystem	Shutdown System
G-12	Pd Too Low	Manifold Subsystem	Alarm Signal
G-13	F1 Too High	DAC Subsystem	Shutdown F1
G-14	F1 Too Low	DAC Subsystem	Alarm Signal
G-15	Vw Set-point High	Pipe Network	Alarm Signal

G-16 Vw Set-point Low	Pipe Network	Alarm Signal
G-17 Vo Set-point High	Pipe Network	Alarm Signal
G-18 Vo Set-point Low	Pipe Network	Alarm Signal
G-19 Pump Degraded	Vacuum Subsystem	Shutdown System
G-20 PST-10 Failure	Power Subsystem	Shutdown System
G-21 PST-15 Failure	Power Subsystem	Shutdown System
G-22 FMs Ground Error	Power Subsystem	Alarm Signal

Table 3.2 Sensor Failure Modes

No. Failure Mode	Isolation Level	Automatic Action
S-01 T/C 0 Failure	DAC Subsystem	Alarm Signal
S-02 VMz Failure	DAC Subsystem	Alarm Signal
S-03 FM1 Failure	DAC Subsystem	Alarm Signal
S-04 FM2 Failure	DAC Subsystem	Alarm Signal
S-05 FM3 Failure	DAC Subsystem	Alarm Signal
S-06 IMz Failure	DAC Subsystem	Alarm Signal
S-07 SW1 Failure	DAC Subsystem	Alarm Signal
S-08 PTp Failure	DAC Subsystem	Alarm Signal
S-09 PTr Failure	DAC Subsystem	Alarm Signal
S-10 PTd Failure	DAC Subsystem	Alarm Signal
S-11 PTc Failure	DAC Subsystem	Alarm Signal
S-12 PTz Failure	DAC Subsystem	Alarm Signal
S-13 PT1 Failure	DAC Subsystem	Alarm Signal
S-14 PT2 Failure	DAC Subsystem	Alarm Signal
S-15 PT3 Failure	DAC Subsystem	Alarm Signal
S-16 PTt Failure	DAC Subsystem	Alarm Signal

Table 3.3 Leakage Failure Modes

No. Leakage Location	Isolation Level	Automatic Action
L-01 Pipeline #0-1	Pipe Network	Alarm Signal
L-02 Pipeline #1-1	Pipe Network	Alarm Signal

L-03 Pipeline #1-2	Pipe Network	Alarm Signal
L-04 Pipeline #1-3	Pipe Network	Alarm Signal
L-05 Cell Cap	Pipe Network	Alarm Signal
L-06 Zirconia Tube	Pipe Network	Shutdown System
L-07 Pipeline #2-1	Pipe Network	Shutdown System
L-08 Pipeline #2-2	Pipe Network	Shutdown System
L-09 Pipeline #2-3	Pipe Network	Shutdown System
L-10 Pipeline #2-4	Pipe Network	Shutdown System
L-11 Pipeline #2-5	Pipe Network	Shutdown System
L-12 Pipeline #3-1	Pipe Network	Alarm Signal
L-13 Pipeline #3-2	Pipe Network	Alarm Signal
L-14 Pipeline #3-3	Pipe Network	Alarm Signal
L-15 Pipeline #3-4	Pipe Network	Alarm Signal
L-16 Pipeline #3-5	Pipe Network	Alarm Signal

Where: failure modes with "G" mean "General System Failure"
failure modes with "S" mean "Sensor Failure"
failure modes with "L" mean "Leakage"

The control actions are determined by examining the following criterions :

(1) If a failure mode will damage any other subsystem or component when the failure is not able to be treated immediately after it happens, the control action for the failure mode is set to be "Shutdown System".

(2) If a failure mode will cause the oxygen pipeline to be contaminated, the control action for the failure mode is set to be "Shutdown System".

(3) The control action for any other failure mode which is not so critical as criterion (1) and (2) is set to be "Alarm Signal".

The schematic diagram for the leakage location representation numbers is shown in Fig. 3.2.

Among those failure modes, two kinds of failure modes are critical and deserve more attention, i.e. failure of the zirconia tube and large fluid line leak (see Fig. 2.1). The cell may be damaged by either using very high cell voltage from the power supply subsystem or by being heated to higher temperatures than the set-point. The maximum allowable voltage, Vmax, can be derived from Eq. 3.1-2 or 3.1-3. If the oxygen cell is damaged, no oxygen will be produced and the expert system will evoke shutdown procedures immediately. However, if multiple oxygen cells are used in the future, the expert system should be able to isolate the damaged cell and reconfigure the overall cell network. The expert system may even follow some testing procedures to make sure the damaged cell has really failed. The second critical mode is a major leakage in a flow line. Because the pressure inside the system is far below atmospheric pressure, a significant leak will draw external air into the system and the line pressure will increase rapidly toward the atmospheric level. A large leakage in "pipeline #1" or "pipeline #3" (see Figure 3.2) will degrade the oxygen production seriously and make the pressure difference between inside and outside of the zirconia tube large so that the tube might be broken. So, the whole system should be shut down immediately. But, if the leakage is small, it is considered to be noncritical and only "Alarm Signal" will be issued. A leakage in the zirconia tube or "pipeline #2", even a small leakage, is considered to be critical because it might cause the oxygen in pipeline #2 to be contaminated.

3.3 Expert System Methodology

There are two basic types of inference engines. One is called "forward chaining" and the other is called "backward chaining" [26]. Forward chaining starts from the beginning, collects information based on the rules in the knowledge base and reachs conclusions based on this information. While, backward chaining takes all the possible conclusions and then tries to verify each one by working backward to known facts. In this research, forward chaining first goes through some block diagrams which represent the failure detection and isolation algorithm and then acquires sensor data, when the algorithm calls for it, until certain failure modes are reached. Backward chaining gathers all the sensor data, creates a sensor pattern file, and then compares the pattern file with each pattern file of failure modes until a pattern match is found. Whether a system should use forward chaining or backward chaining depends on the number of facts, which are the sensor values, and the number of conclusions which are the failure modes. Forward chaining works better when there are only a limited number of facts and many conclusions could be reached from these facts. Backward chaining works better when there are only a few possible conclusions with many facts. In this research, the number of failure modes is comparable to the number of sensor values, so both methods are used. However, only backward chaining is implemented for further testing.

3.3.1 Expert System Using Forward Chaining

Corresponding to each failure mode listed in Tables 3.1, 3.2 and 3.3, a set of all known or postulated symptoms have been considered. A knowledge-based expert system is then developed to provide the following functions [27] :

(1) Data Acquisition and Self-health Checkout

On a regular, timely basis, the expert system will perform the following tasks: (a) Collect and store system parametric data (pressure, temperature, voltage, flow-rate) in an array in computer memory at desired time intervals. (Important parametric data like cell voltage and oven temperature may have higher sampling rates.) (b) Evaluate system performance after each sampling cycle by using the failure detection algorithm described later. (c) Update system displays which report system performance. (d) Periodically transfer representative data (i.e., statistical average, standard deviation, etc.) to printer or floppy-disk for each parameter monitored. The computer memory will only save data acquired during the latest sampling time interval in order to reduce memory requirements. (e) If any failure mode is detected, the alarm signal will be activated and the corresponding failure messages will be shown on the screen to tell the operator whether the system has been shutdown due to critical failure or certain adjustments are required. Figure 3.3 shows the suggested data sampling, storing and failure detection algorithm circle diagram.

(2) Emergency Shutdown

For emergency conditions, the expert system will evoke the shutdown procedures which may include : (a) Activation of the local alarm, notifying the immediate area of a system failure. (b) Disengagement of power to the system, turning the feed-gas flow controller (item 5 in Fig 2.1) off. (c) Transfer all data in the current memory to disk for subsequent failure analysis. (d) Place local telephone calls to notify project personnel of a system failure and (e) Switch the system from its autonomous mode to an interactive mode. After system shutdown, the expert system may ask some questions about further diagnosis or repairing procedures, and can work interactively with a terrestrial control center.

(3) Failure Detection and Isolation (F.D.I.)

A knowledge-based algorithm for system Failure Detection and Isolation (F.D.I.) has been developed and is described by using the block diagrams shown in Fig. 3.4. From the acquired sensor data, the F.D.I. algorithm can identify each existing failure mode listed in Tables 3.1, 3.2 and 3.3 under the following assumptions:

(a) Main power never fails or it has to be monitored by another higher level F.D.I. algorithm.

(b) Computer and data acquisition system (Series 500 box) never fails or they have to be monitored by another higher level F.D.I. algorithm.

(c) Only one failure mode may occur at a time.

For those failure modes not considered or those which combine at least two known failure modes, the F.D.I. algorithm can only display the existing symptoms, turn on the alarm and continue to monitor the system. Any new failure mode found in the future during operation can be added to the knowledge-base. If hardware redundancies (like oven thermocouples) or analytical redundancies are available in the future, the F.D.I. algorithm will be able to detect and distinguish between sensor failure and subsystem failure.

Because a cell failure is critical, the cell voltage, cell current and oven temperature are checked more frequently. All the other parametric data are checked after every 200 sampling cycles of the cell voltage and current. If the cell voltage exceeds Vzc, cell current exceeds 500 mA (maximum measurable current for current measurement design) or oven temperature exceeds Tch, a cell failure is acknowledged and system shutdown procedures are evoked.

If hardware redundancies exist, the F.D.I. algorithm can be developed as follows by assuming only one sensor fails at a time. Suppose we put three thermocouples at different places inside the oven with the following measurements :

$$[Tcmi] = [ki]\ Tc + [vi] \tag{3.3-1}$$

where

$$[Tcmi] = \begin{bmatrix} Tcm1 \\ Tcm2 \\ Tcm3 \end{bmatrix} ; \quad [ki] = \begin{bmatrix} k1 \\ k2 \\ k3 \end{bmatrix} ; \quad [vi] = \begin{bmatrix} v1 \\ v2 \\ v3 \end{bmatrix}$$

where Tcmi (i=1,2,3) is the measurement value of each thermocouples, Tc is the averaged temperature of the oven, vi (i=1,2,3) is the measurement noise and ki represents the factor related to different spacial placements of the thermocouples. Eq. 3.3-1 means that the measurement values of the three thermocouples can be expressed as a linear transformation of the averaged temperature of the oven and measurement noise. If three calibrated thermocouples are located at the same point, then k1=k2=k3=1, otherwise ki can be determined by testing under a stable condition. From Equation 3.3-1, we can derive three parity residuals :

$$[Par] = [Vij]\ [Tcmj] \qquad (3.3\text{-}2)$$

where

$$[Par] = \begin{bmatrix} Par12 \\ Par23 \\ Par13 \end{bmatrix} ; \quad [Vij] = \begin{bmatrix} 1/k1 & -1/k2 & 0 \\ 0 & 1/k2 & -1/k3 \\ 1/k1 & 0 & -1/k3 \end{bmatrix}$$

These three parity residuals are then compared with three corresponding thresholds Th12, Th23 and Th13 to identify any sensor failure. For example, Par12 >Th12, Par23 <Th23 and Par13 >Th13 indicate that thermocouple #1 has failed and Tcm1 should not be used. All thresholds are chosen to be 10% of the change of Tc and can be adjusted according to the desired accuracy for each thermocouples. Similar algorithm can be applied to other sensors if the hardware redundancies are available.

Besides hardware redundancies, analytical redundancies methods can be used for F.D.I. of flow measurements. For example, if there is no leakage in the whole system (see Fig. 2.1) and the cell performs in a nominal condition, then

$$F1 = F2 + F3 \qquad \text{(in gm/hr)}$$
$$= f2.F1 + f3.F1 = (\ f2 + f3\ )\ F1, \quad \text{with } f2+f3=f1=1$$

and

$$[Fmi] = [fi]\ F1 + [vi] \tag{3.3-3}$$

where

$$[Fmi] = \begin{bmatrix} Fm1 \\ Fm2 \\ Fm3 \end{bmatrix} ; \quad [fi] = \begin{bmatrix} f1 \\ f2 \\ f3 \end{bmatrix} ; \quad [vi] = \begin{bmatrix} v1 \\ v2 \\ v3 \end{bmatrix}$$

which are similar to Eq. 3.3-1. So we can identify the failure of one of three flow meters without having hardware redundancy. However, this method can be applied only by assuming that deviation of F1 from its nominal set-point is small and there is no leakage and no other subsystem failure. Because the system is nonlinear, [fi] cannot be kept constant if F1 deviates far from its nominal set-point. The same problem will occur if leakages or subsystem failures exist in the system. So this method cannot be used to identify the failure of sensors. Instead, we use other system parameters to identify the failure of sensors. If all the other sensor readings are normal except a particular one which exceeds the range of a pre-assigned threshold, then it follows that the particular sensor has failed. For most cases, this F.D.I. algorithm assumes that only one failure mode exists at a time.

(4) Computer Program Realization

Based on the algorithm shown in Fig. 3.4, a computer program named "MODFDI1.SIM" has been developed using BASIC language to provide numerical simulations. BASIC programming was chosen in order to interface with the hardware configuration (Keithley 500 data acquisition system) with high speed data processing in the future. The nominal values and allowable tolerances of the system parameters which are assigned in data files in the program are provided from experimental data in Chap. 5. Another computer program called "MODTRD3.BAS" provides the demonstration of real-time F.D.I of three redundant thermocouples in an oven.

3.3.2 Expert System Using Backward Chaining (or Pattern Matching)

With the same assumptions as those of the previous section (section (3) of 3.3.1), another methodology of F.D.I. using backward chaining has been developed and described as follows:

(1) Symbolic Manipulation

Suppose there are n sensors installed in a system and their output values acquired by the data acquisition system at the instant kT are:

Si(kT) i=1,2,...,n ; T=sampling period

or simply Si(k)

Si can be a digital value or can be transfered into a knowledge representative word such as "HIGH","LOW" or "OK",..etc. Failure modes can be tested and the corresponding sensor output values for each failure mode are recorded. Assume that each failure mode can be identified uniquely by a set of current sensor output values Si(k),i=1,2,...,n

i.e. FM(k) = f(S1, S2, ... , Sn) (3.3-4)

where FM(k) = failure mode identified at the instant kT

Value of FM(k) can be any of the failure mode codes listed in Tables 3.1, 3.2 and 3.3. FM(k) = N means that the system is in normal operating condition. FM(k) = ? represents the case in which the current version of the knowledge-base cannot identify the failure. It might be a new failure mode which has not been recognized or a failure mode with two or more known failure modes happening at the same time. In summary, the failure mode at a discrete time can be identified by examining the pattern of all the sensor output.

(2) Knowledge and Rule Base

The threshold of each sensor output for different failure modes may not be the same. Here we use GnnSSH and GnnSSL to represent the upper limit and the lower limit of sensor output SS for failure mode Gnn. For example, G01TCL means the lower limit of the oven temperature Tc for a general failure mode G01; and L01P3H means the upper limit of the waste-gas pressure P3 for leakage failure mode L01. The relation between failure modes and sensor outputs for the MOD machine have been tested and listed in the following table which is considered to be the rule base for the F.D.I. algorithm.

Table 3.4 Rule Base for F.D.I. Algorithm

Rule 0: If all the sensor outputs are normal except a particular one which is out of the normal range,
Then the particular sensor failed.

Rule 1: If Tc>G01TCH and Iz>G01IZH and Vz<G01VZL and F2>G01F2H,
Then failure mode G01 is justified.

Rule 2: If Tc<G02TCL and Iz<G02IZL and Vz>G02VZH and F2<G02F2L,
Then failure mode G02 is justified.

Rule 3: If Vz>G03VZH and Iz>G03IZH and F2>G03F2H and Pz>G03PZH,
Then failure mode G03 is justified.

Rule 4: If Vz<G04VZL and Iz<G04IZL and F2<G04F2L and Pz<G04PZL,
Then failure mode G04 is justified.

Rule 5: not to be used for this approach

Rule 6: If Iz<G06IZL and Vz>G06VZH and Pz<G06PZL and Tc is normal,
Then failure mode G06 is justified.

Rule 7: If Iz<G07IZL and Vz>G07VZH and Pz<G07PZL and Tc is normal,
Then failure mode G07 is justified.

Rule 8: If Iz>G08IZH and Vz<G08VZL and F2<G08F2L and Tc is normal,
Then failure mode G08 is justified.

Rule 9: If Pd<G09PDL and SW1=P AND Pp<G09PPL AND Pr>G09PRL,
Then failure mode G09 is justified.

Rule 10: If Pr<G10PRL,
Then failure mode G10 is justified.

Rule 11: If Pd>G11PDH and Pp<G11PPH,
Then failure mode G11 is justified.

Rule 12: If Pd<G12PDL and Pp>G12PPL,
Then failure mode G12 is justified.

Rule 13: If F1>G13F1H and F3>G13F3H and P1>G13P1H and P3>G13P3H,
Then failure mode G13 is justified.

Rule 14: If F1<G14F1L and F3<G14F3L and P1<G14P1L and P3<G14P3L,
Then failure mode G14 is justified.

Rule 15: If Pc>G15PCH and P1>G15P1H and F1 and F3 are normal,
Then failure mode G15 is justified.

Rule 16: If Pc<G16PCL and P1<G16P1L and F1 and F3 are normal,
Then failure mode G16 is justified.

Rule 17: If Pz>G17PZH and P2>G17P2H and Pc is normal,
Then failure mode G17 is justified.

Rule 18: If Pz<G18PZL and P2<G18P2L and Pc is normal,
Then failure mode G18 is justified.

Rule 19: If Pc>G19PCH and Pz>G19PZH and F1 and F3 are normal,
Then failure mode G19 is justified.

Rule 20: If all sensor readings are normal except P1,P2,P3,Pt,Pp,and Pr,
Then failure mode G20 is justified.

Rule 21: If F1,F2,F3,Pc and Pz are almost zero,
Then failure mode G21 is justified.

Rule 22: If F1,F2 and F3 are negative,
Then failure mode G22 is justified.

Rule 23: If P1>L01P1H and P3>L01P3H and F3>L01F3H and F1 is normal,
Then Leakage failure L01 is justified.

(3) Testing Procedures

In order to determine the thresholds for each failure mode, several testing categories are suggested. The normal operating conditions are chosen to be :

T = 1200 K, Vz = 1.0 V, F1 = 50 sccm, F2 = 0.32 sccm, F3 = 50 sccm,
Iz = 150 mA, S1 = P, Pp = Pr > 100 psi, Pd =40 psi,
Pc = 125 mb, Pz = 50 mb, P1 =190 mb, P2 =0 mb, P3 =100 mb, Pt = 0 mb

---

Before starting the test, it is necessary for the system to be operated under normal conditions. The failure test can be divided into several steps described as follows:

1. <u>Statistical tests of sensor readings</u>
2. <u>Tests of each individual general failure mode</u>
3. <u>Tests of leakage failure</u>

The detailed testing procedures and results are given in Chapter 5.

(4) Computer Program Realization

Based on the knowledge base and the testing results, a computer program named "MODFDI2.BAS" has been developed using BASIC language and is listed in Appendix A. The flow chart for MODFDI2.BAS is shown in Fig. 3.5. Figure 3.6 is the main menu as well as the four basic functions of MODFDI2.BAS. To use this program, one has to follow the basic steps described in Section 3.1 to initialize the system. (Refer to Fig. 3.7 The Suggested System Initialization Parameters.) Once the system has been adjusted within normal operating condition (Step 10), press the F2 key and double check the system parameters (Fig. 3.8). If every individual parameter is correct, the F.D.I algorithm can be triggered. Figure 3.9 shows how the Real-Time Update Screen looks.

The data acquired during the latest operating period can be saved in program memory. Function F3 is to load the data from program memory to the floppy-disk so that we can analyze the data using Lotus1-2-3.

3.4 Expert System Using Application Software

It is easier to develop expert systems by using expert system shells. Expert system shells are tools which allow us to create expert systems without system programming. The criteria for selecting a competent software package for different jobs are presented in many expert system documents [9]. Generally speaking, the problem characteristics have to be considered first. The problem characteristics suggest certain solution features. The solution features together with the desired system features form the basis for choosing a particular shell. For the Mars oxygen production system, we want to diagnose the system failure modes from 16 sensor readings. There are at least 23 possible failure modes. Furthermore, we want the shell to be able to interface with the data acquisition system so that the diagnostic process can be in real time. Because an IBM/PC/AT computer is used, the memory capacity is also a crucial factor.

(1) 1st-CLASS and the Diagnosis Circle

Based on the above criteria, an application software called 1st-Class was chosen. There is a series of six screens in developing a knowledge base with 1st-Class (Fig. 3.10). The Files screen allows us to create, save, get and print or export knowledge-base (KB) files. The Definitions screen is the first step in creating a KB by specifying the following items:

(a) results--the outcomes, conclusions or goals
(b) factors in a KB--elements involved in determining the result
(c) values for each factor

The Examples screen lets us enter the examples that the inference engine will use to solve the problems. The Methods screen can be used to select a method for building a rule for the KB. Finally, we can either check the rules being built or run the ADVISOR program. The expert system shell 1st-Class can handle up to 32 factors, 32 values, 32 results and 255 examples which are enough for our case. Problems arose when we tried to interface 1st-Class with the SOFT500 data acquisition system. To execute the "ADVISOR.EXE" file that was built automatically when completing editing KB in 1st-Class needs 200K memory space. The minimum memory required for SOFT500 is 512K and the basic program to facilitate data acquisition needs about 20K. Therefore, we need at least 732K totally, but only 640K was available for the current configuration of computer memory. This problem has been solved by simply developing a data acquisition program of our own by using two basic commands "POKE" and "PEEK" so that we can acquire data without using SOFT500. This approach frees a lot of memory space and thus enables the interface between the data acquisition system and the 1st-Class expert system shell.

Figure 3.11 shows how this approach works.

1. After the basic program finishes each data acquisition period, it can build an answer file for 1st-Class.
2. At appropriate time, the basic program will run the 1st-CLASS ADVISOR program by executing the basic command "SHELL".
3. The answer file which is built in the first step will automatically answer ADVISOR's questions.
4. The ADVISOR can build a report file using the 1st-CLASS report generator.
5. The basic program can read the report, show the results on the screen and take the required action.
6. This completes one diagnosis circle and the program continues processing.

(2) Computer Program Realization

The basic program that acquires data, sends the answer file to and receives the report file from the ADVISOR program is called "MODFDI3.1SC". The knowledge base file that the ADVISOR program uses is called "MOD1SC3.KBM". Because the 1st-CLASS is not very good at handling digital manipulation, we cannot use different failure thresholds for different failure modes. We have to simplify the rule base and allow only three states (High, Low or Normal) for each sensor reading. The Match Method is chosen to compare the pattern of the answer file with that of the knowledge base in a desired order. Parts of program MODFDI3.1SC and file MOD1SC3.KBM are listed in the Appendix B and C.

3.5 Expert System Using AI Language PROLOG

Another way to develop an expert system is to use the dominant artificial intelligence language PROLOG. PROLOG is a declarative language or "what-type" language. This means that, given the necessary facts and rules, it can use deductive reasoning to solve programming problems. By contrast, BASIC, FORTRAN and other traditional computer languages are procedural languages or "How-type" languages: the programmer must provide step-by-step procedures telling the computer how to solve problems. The PROLOG programmer needs only to supply a description of the problem (or the goal) and the rules for solving it, and the PROLOG system will determine how to go for a solution.

A computer program called "MODFDI4.PRO" has been developed using TurboPROLOG to provide numerical simulations for the same purpose. Program MODFDI4.PRO is also listed in the Appendix D. A screen of TurboPROLOG is shown in Fig. 3.12.

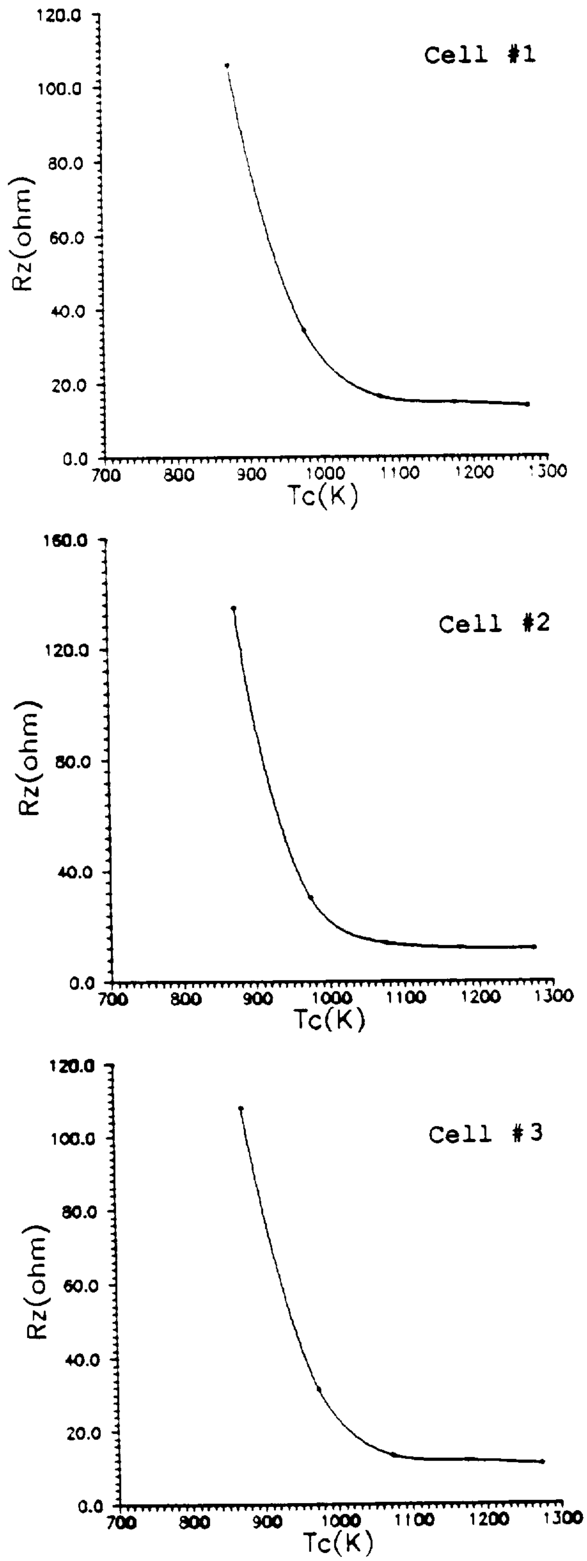


Fig. 3.1 Cell Resistance vs. Temperature Curve

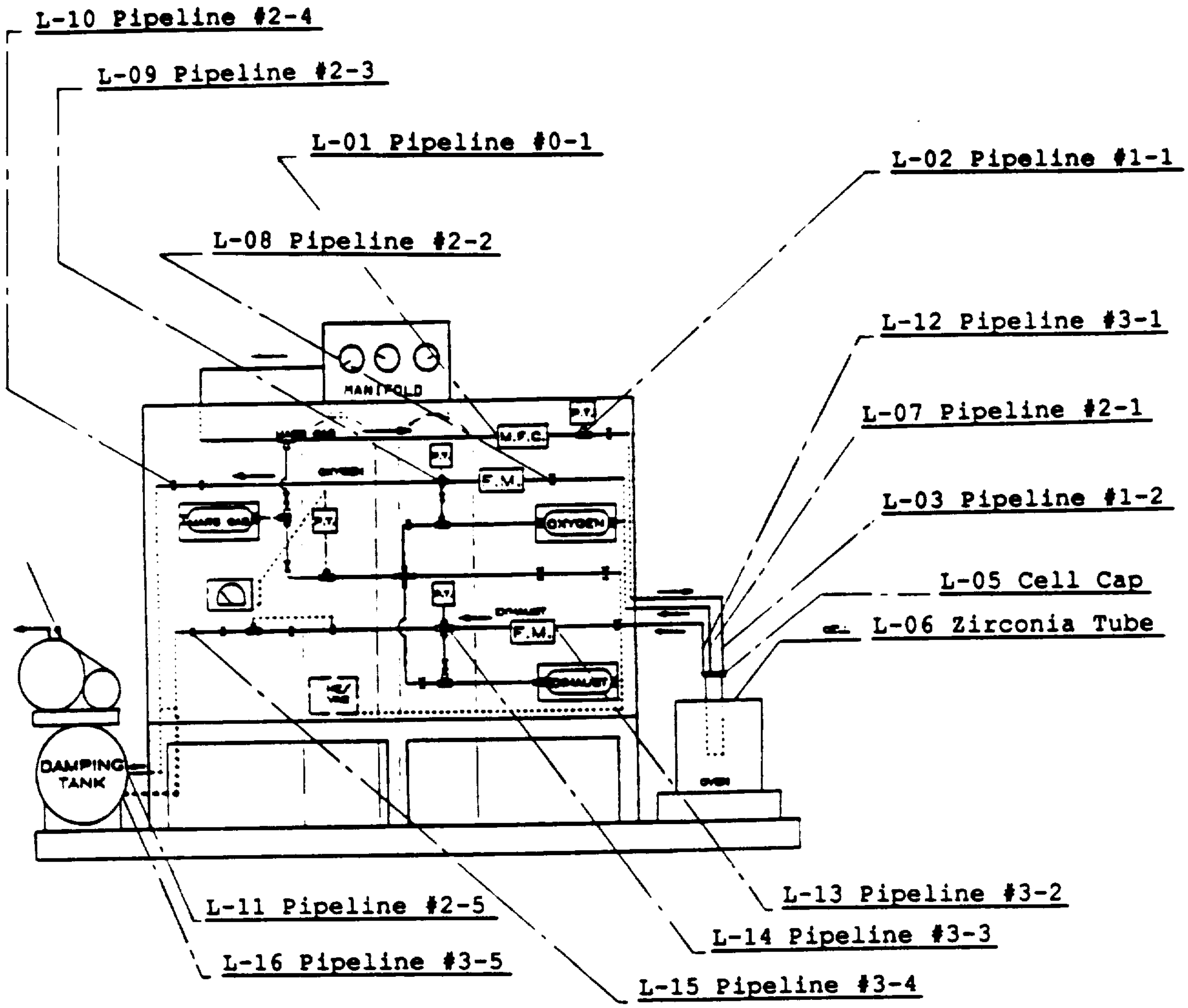


Fig. 3.2 Identification Numbers for Leakage Locations

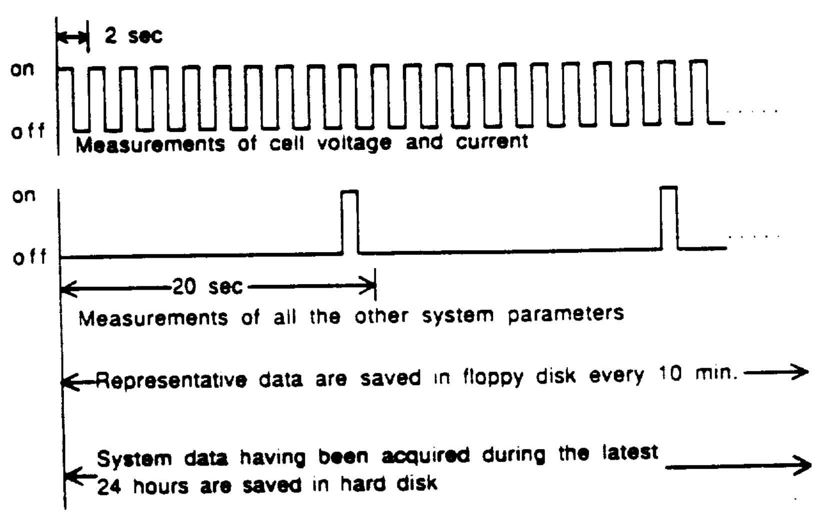
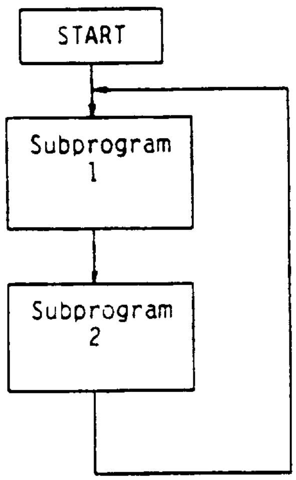
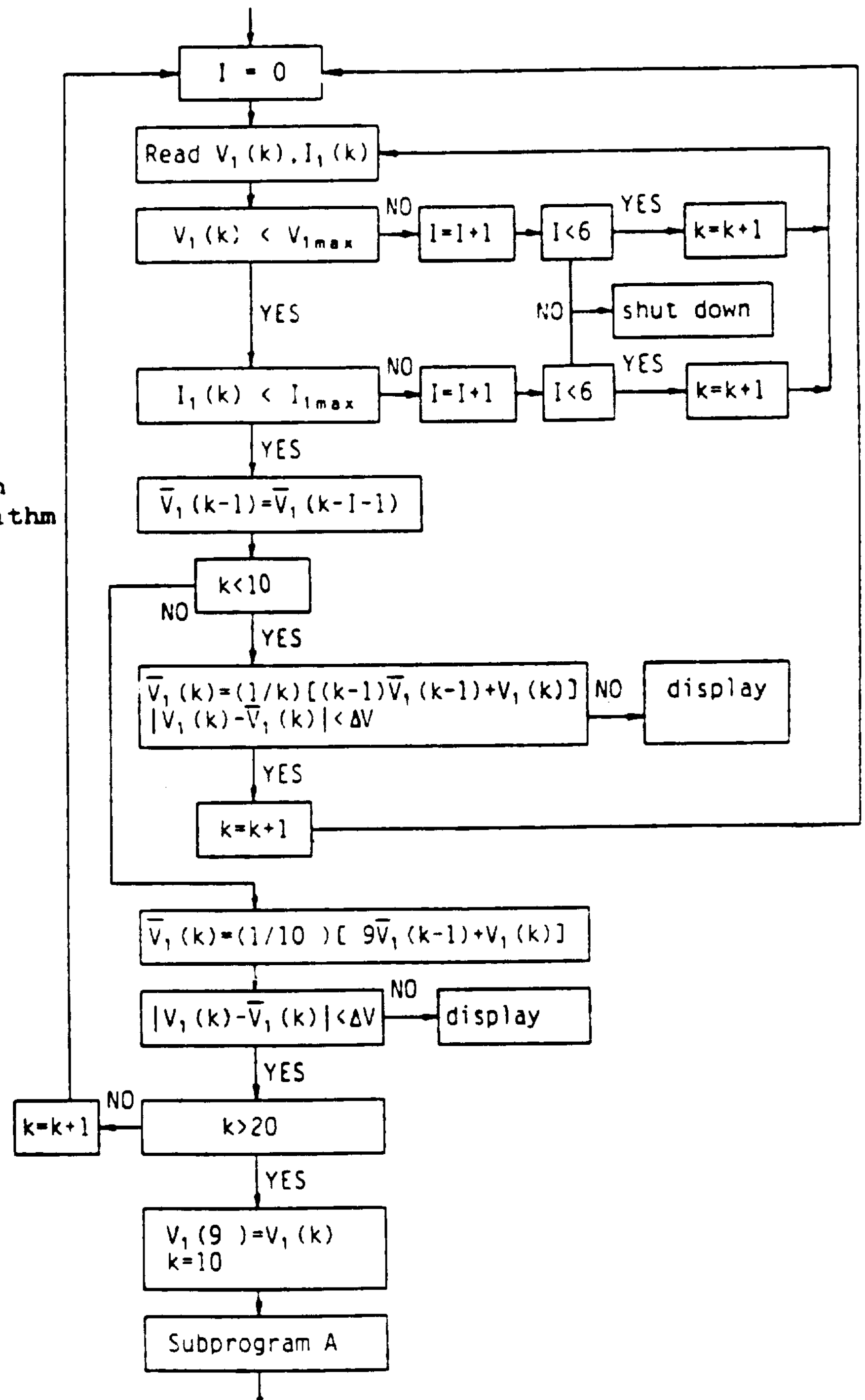


Fig. 3.3 Suggested Data Sampling and Expert System Circles

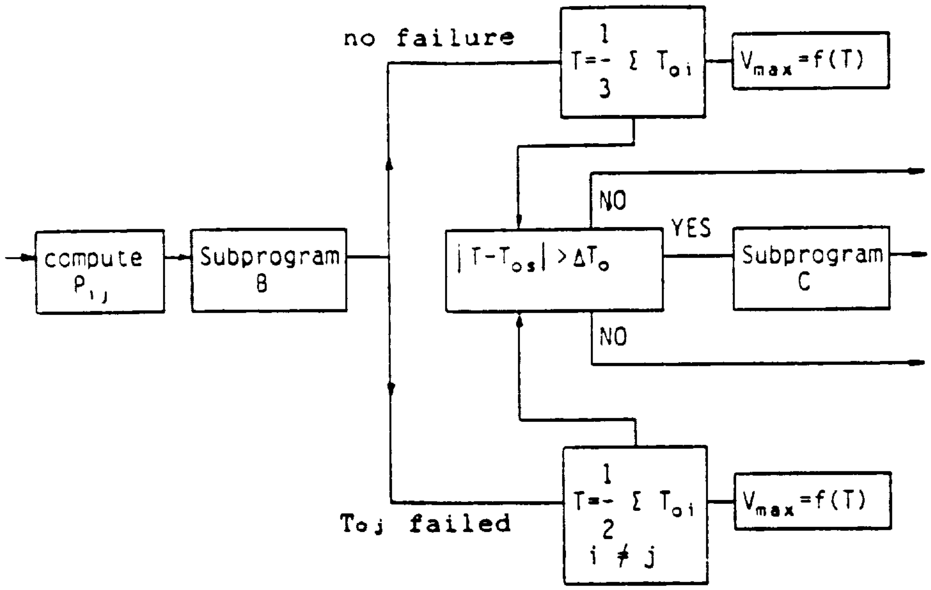


(a) Block Diagrams of Failure Detection and Isolation Algorithm

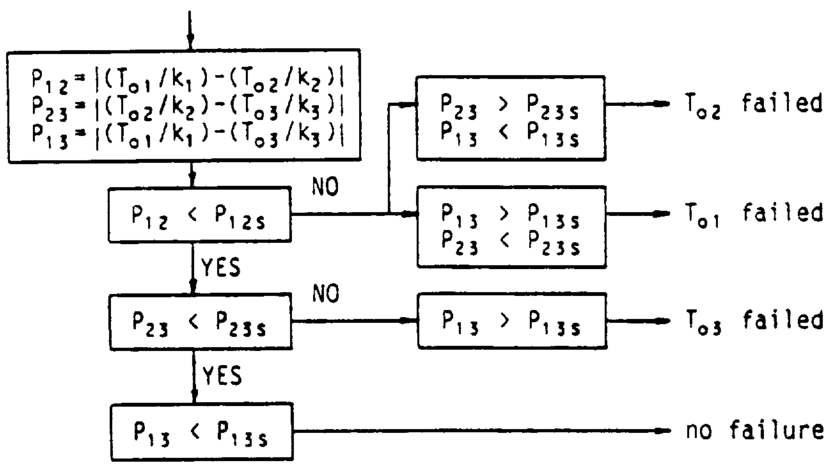


(b) Subprogram 1

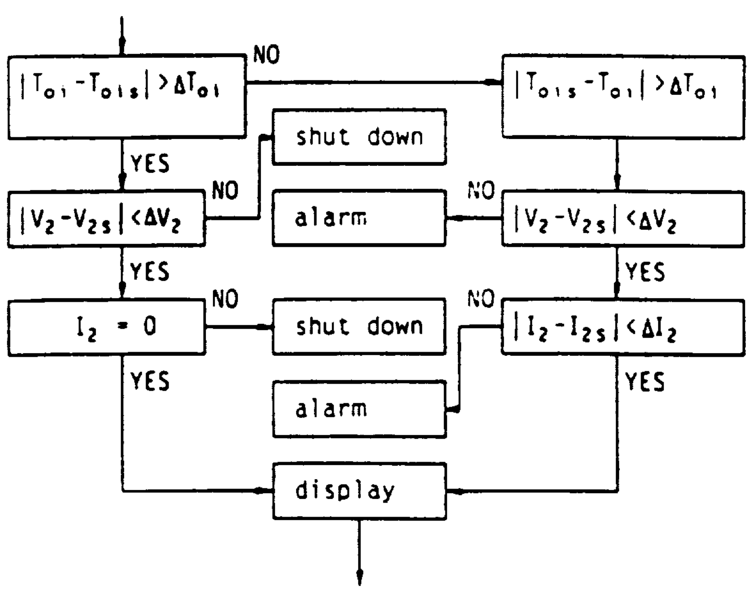
Fig. 3.4(a to i) Block Diagrams of F.D.I. Algorithm



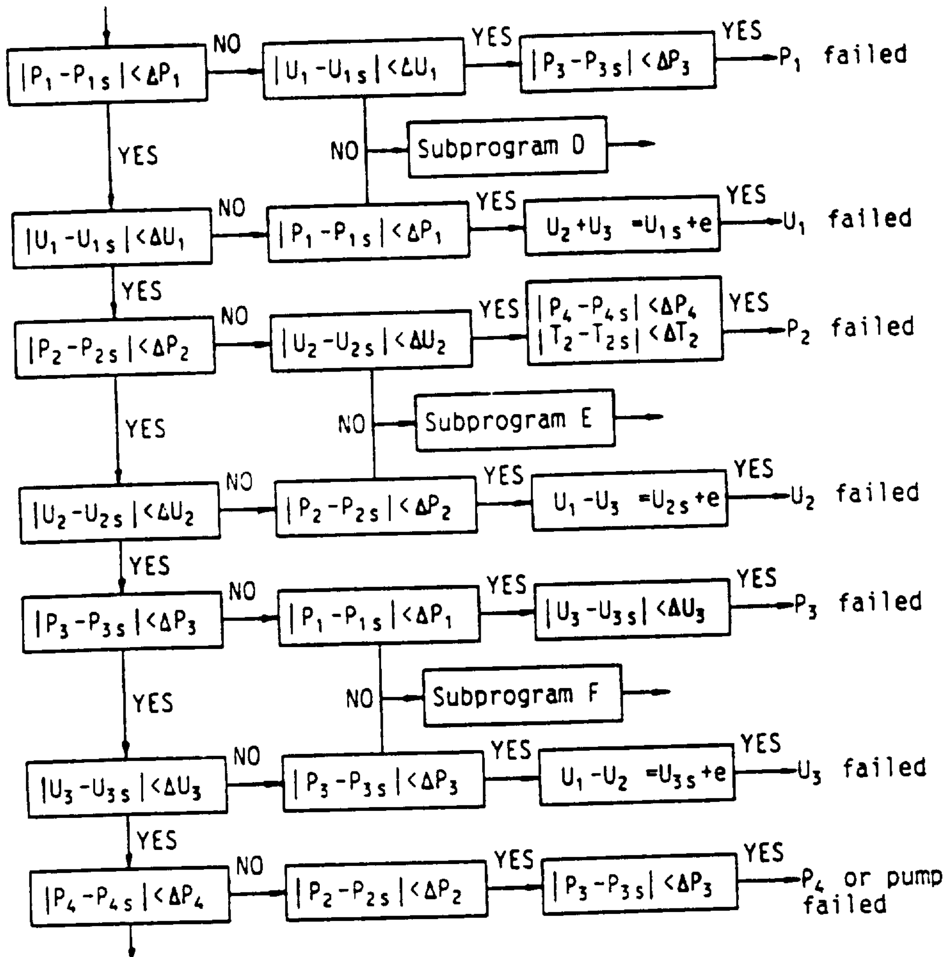
(c) Subprogram A



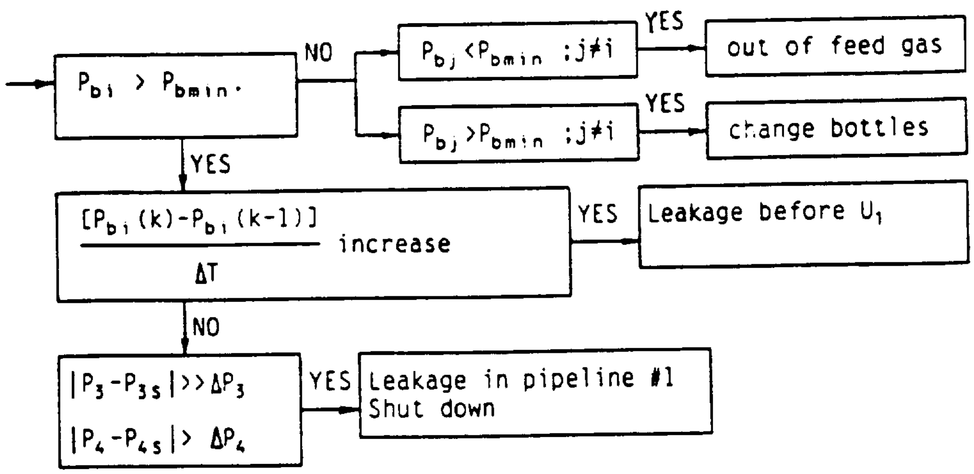
(d) Subprogram B



(e) Subprogram C

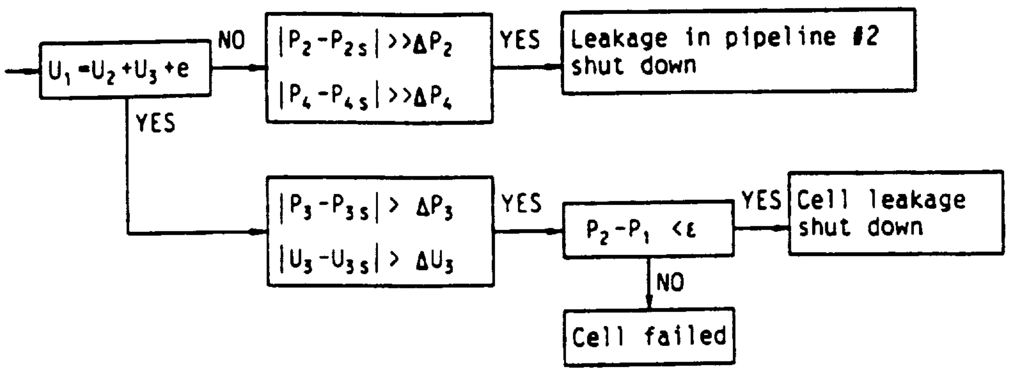


(f) Subprogram 2

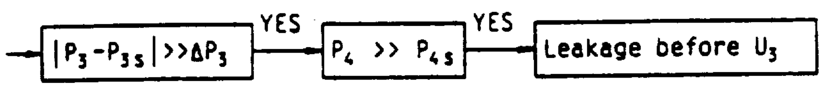


(Assume that bottle #i is currently being used)

(g) Subprogram D



(h) Subprogram E



(i) Subprogram F

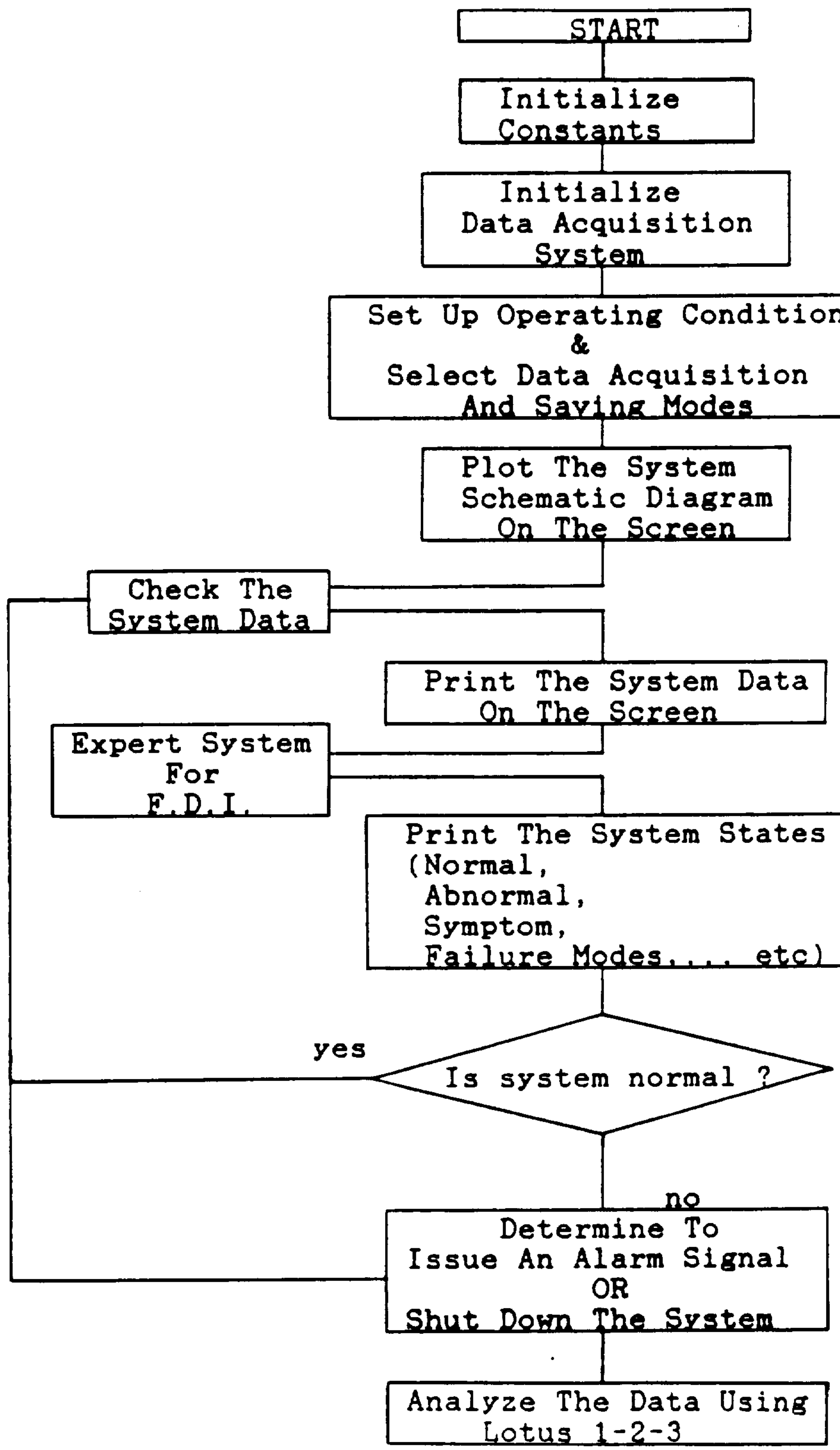


Fig. 3.5 Flow Chart of Program MODFDI2.BAS

OLD DOMINION UNIVERSITY, MECHANICAL ENGINEERING DEPARTMENT
NORFOLK, VIRGINIA

MARS OXYGEN DEMONSTRATION PROJECT (MOD)

MAIN MENU

F1 .... Real-Time Update Screen

F2 .... Expert System For F.D.I.

F3 .... Analyze Individual Input Readings

F4 .... General System Information

F10 .... Exit This Program

Fig. 3.6 Basic Functions of MODFDI2.BAS

```
Barometric pressure (mb)? 1013
The set point you use now is  0
Flow set point to be used(0-5)? 0.67024
Need 'MOD.LOG' flle (Y/N) ? y
Data to be printed (Y/N) ? y
Sampling period (sec) ? 10
Send data every ?th update ? 60[
```

Fig. 3.7 The Suggested System Initialization Parameters

Set Up Operating Conditions

```
                 The current system parameters are:
Cell Temperature(K) Tcn= 1194.64
Cell Voltage(V) Vzn= 2.263736
Feedgas Flowrate(gm/hr) F1n= 5.450035
Oxygen Flowrate(gm/hr) F2n= 8.166056E-02
Wastegas Flowrate(gm/hr) F3n= 4.426924
Cell Current(mA) Izn= 353.5911
Manifold Switch S1(P/R)=P
Delivery Pressure(psi) Pdn= 40
Cell Pressure(mb) Pcn= 125.897
Oxygen Prsssure(mb) Pzn= 82.98722
Pipeline #1 Pressure(mb) P1n= 130.4385
Pipeline #2 Pressure(V) P2n= 7.313798
Pipeline #3 Pressure(mb) P3n= 101.713
Tank Pressure(mb) Ptn= 0
Primary Bottle Pres.(psi) Pps= 909.2389
Reserved Bottle Pres.(psi)Prs= 236.8742
Is everything correct (Y/N)
EXPERT SYSTEM starts if (Y)? [
```

Fig. 3.8 Double Check The System Performance

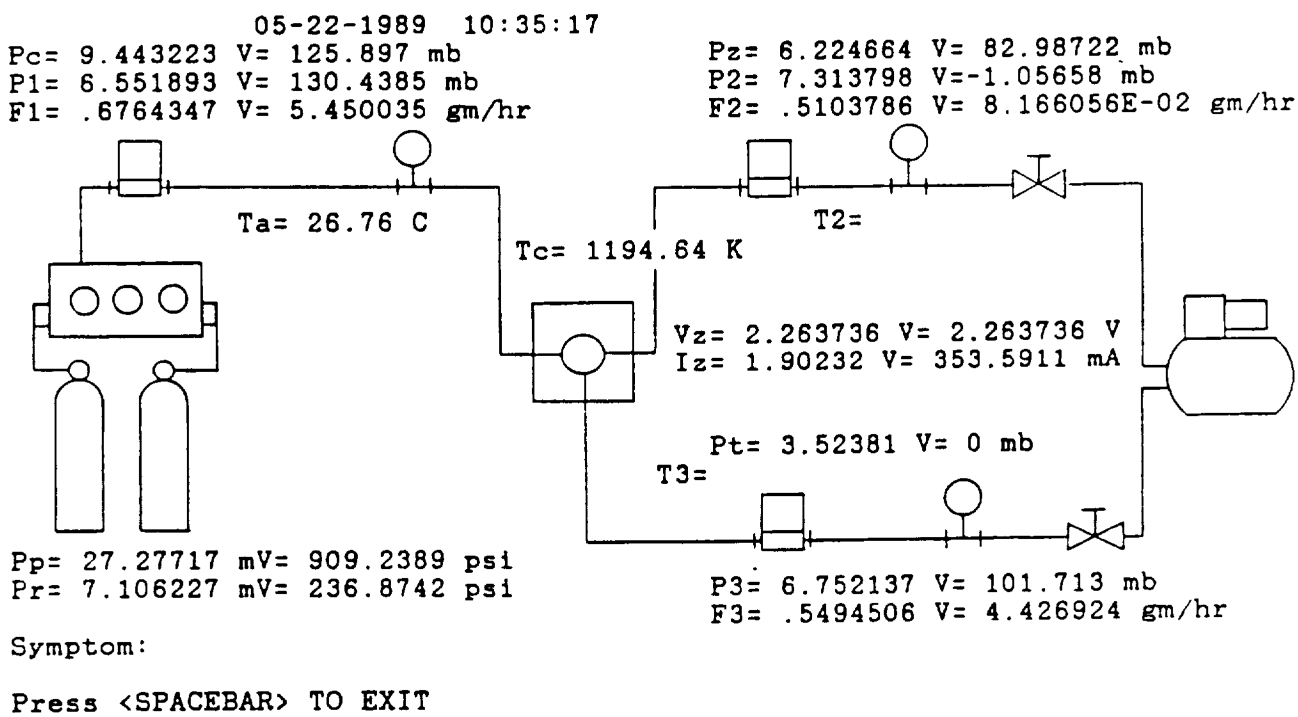


Fig. 3.9 Real-Time Update Screen

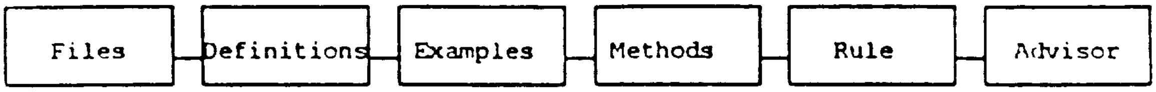


Fig. 3.10 Six Procedures to Develop A K.B with 1st-CLASS

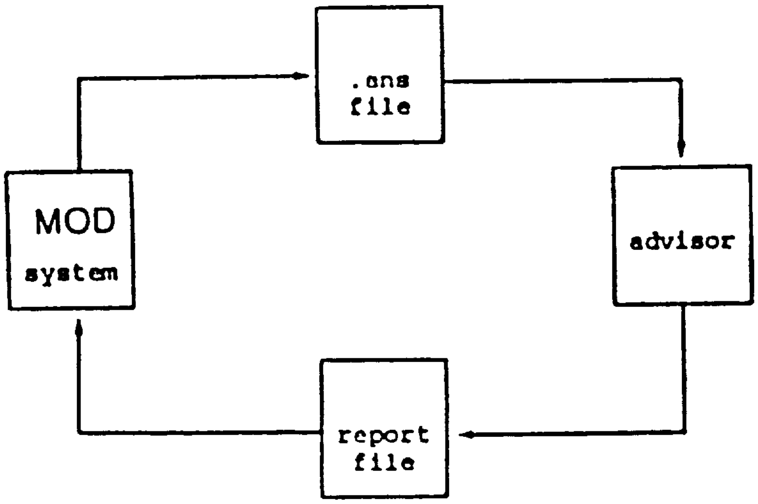


Fig. 3.11 The File Interface of MODFDI3.1SC

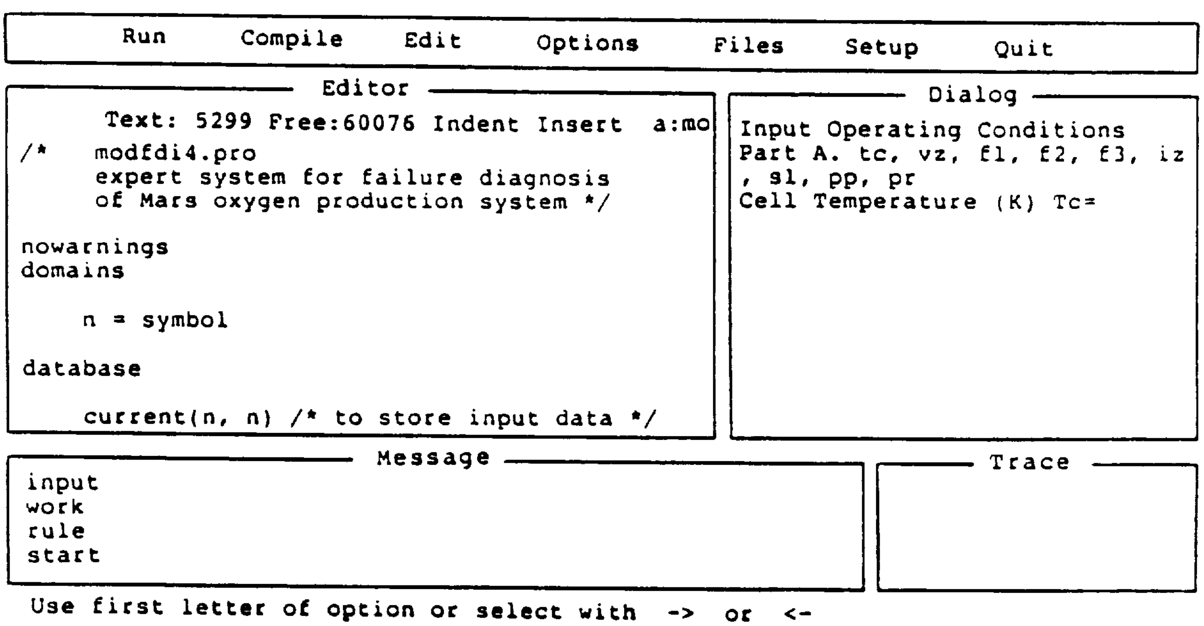


Fig. 3.12 A Screen of TurboPROLOG

Chapter 4

Model-Based Process Fault Detection

The Fault Detection and Isolation (F.D.I.) schemes developed in Chapter 3 are valid only for steady state conditions. In other words, the expert system detects errors in the system and determines the corresponding failure mode based on the assumption that the system with or without failures has been stablized. The expert system is unable to decide which failure mode has occured during the transition. Generally speaking, such an assumption is good enough for most of the process plants since the transition period is not critical. However, if the transient time is long enough to cause damage to some components of the system, then we have to detect it and try to isolate the failure during transition. On the other hand, although the MOD system is being operated around several fixed set points, the system operator might need to change the set points during operation at a later time (eg. change the flow rate). A dynamic F.D.I. scheme has to be applied to monitor the dynamic behavior of the system during transition. We begin this study by modeling the dynamics of the system.

4.1 Mathematical Model of MOD System Using Bond Graph

Static and dynamic models of the gas behavior in the pipelines require complex analysis of fluid dynamics. The Bond graph method is an easier alternative way to derive a mathematical model for the MOD system for further analysis. In order to use this method, it is assumed that the flow gas in the system is incompressible and the fluid velocity is low enough to ensure that the dynamic pressure is not significant. For the first assumption, although gases are considered compressible for most of the cases, gas flow may be considered incompressible provided the flow velocity is small relative to the speed of sound in the fluid. The ratio of the flow speed Vg to the speed of sound C in the fluid is defined as the Mach number M.

$$M = Vg / C$$

$$C = (k R T) / M$$

where M = Mach number

Vg = flow speed of working gas

C = speed of sound in the gas

k = the ratio of specific heats

R = gas constant

T = gas temperature

Gas flows with $M < 0.3$ can be treated as incompressible flows.

For the MOD system, assuming the maximum mass flow rate of feed-gas equals 100 sccm and the operating pressure in pipeline is about 120 mb, Then,

$$Qa = Qs (Ps / Pa) = 100(\text{sccm}) (1013.3(\text{mb}) / 120(\text{mb}))$$

$$= 844.42 \text{ accm} = 14.07 \text{ cm}^3\text{s}^{-1}$$

$$Vg = Qa / A = 14.07(\text{cm}^3\text{s}^{-1}) / 0.164(\text{cm}^2) = 85.79 \text{ cm s}^{-1}$$

$$= 0.86 \text{ m s}^{-1}$$

$$C = (k R T) / M$$

$$= (1.3)(8.31 \text{ J mol}^{-1}\text{K}^{-1})(298 \text{ K})/(44\text{E-3 Kg mol}^{-1})$$

$$= 270 \text{ m s}^{-1}$$

$$M = Vg / C = 0.86 / 270 = 3.185\text{E-3} << 0.3$$

where Qs = volumetric flow rate under Standard Conditions (70 F, and 1 atm)

Qa = volumetric flow rate under actual operating conditions

Ps = standard pressure = 1 atm = 1013.3 mb

Pa = actual pressure

Because M < 0.3, gas in the MOD system is considered to be incompressible.

Now consider the second assumption. The dynamic pressure Pdy is

$$Pdy = 1/2\ p\ (\ Q\ /\ A\ )^2 = (0.5)(0.213)(0.86)^2 \text{ Pascal}$$
$$= 7.9E-4 \text{ mb}$$

which is much smaller than the static pressure (120 mb), so the second assumption is acceptable. [30]

Consider another flow characteristic (laminar, transient or turbulent flow) of the flowing gas.

Reynolds number for a pipe :

$$Re = (\ V\ D\ )/v$$

where

V = flow velocity

D = diameter of the pipe

v = dynamic viscosity = u/p

u = absolute viscosity (Kg/ms)

At T = 1000 K, u = 4.85E-5 Kg/ms

$$v = u/p = 4.85E-5\ /\ 0.213\ Kg/m^3 = 2.27E-4\ m^2/s$$

$$Re = (\ V\ D\ )/v = (0.86(m/s)*0.00457(m))/2.27E-4(m^2/s)$$
$$= 17.31$$

Re << 2300, so it is a laminar flow.

From the above calculations, it is clear that we are dealing with a laminar, incompressibe and static-pressure dominated gas flow. Based on this conclusion, we can use the Bond graph method to derive a dynamic model for the system. However, there is one complicating factor. Although it can be considered an incompressible flow, the density of the gas varies with the pressure along the pipeline. The flow meters used currently are mass flow meters which measure the flow rate in units of sccm. From the viewpoint of power variables in the Bond graph, one should use volume flow-rate instead of mass flow-rate.

Since volume flow-rate varies with density, the flow-rate conservation law will not be true for volume flow rate. We choose to use mass flow rate and treat it as a class of pseudo bond graphs [28] (similar to the pseudo bond graph being used for thermal systems which use temperature and heat flow rate as the effort and flow variables). Pseudo bond graph still works well as long as the effort and flow variables are related correctly. For the MOD system, all the relations between variables are determined via experimental testing.

The proposed block diagram for flow controllers and Bond graph of the MOD system are shown in Fig. 4.1 and Fig. 4.2.

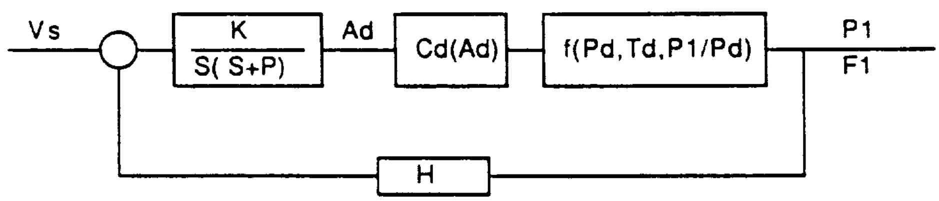


Fig. 4.1 Block diagram of the flow controller

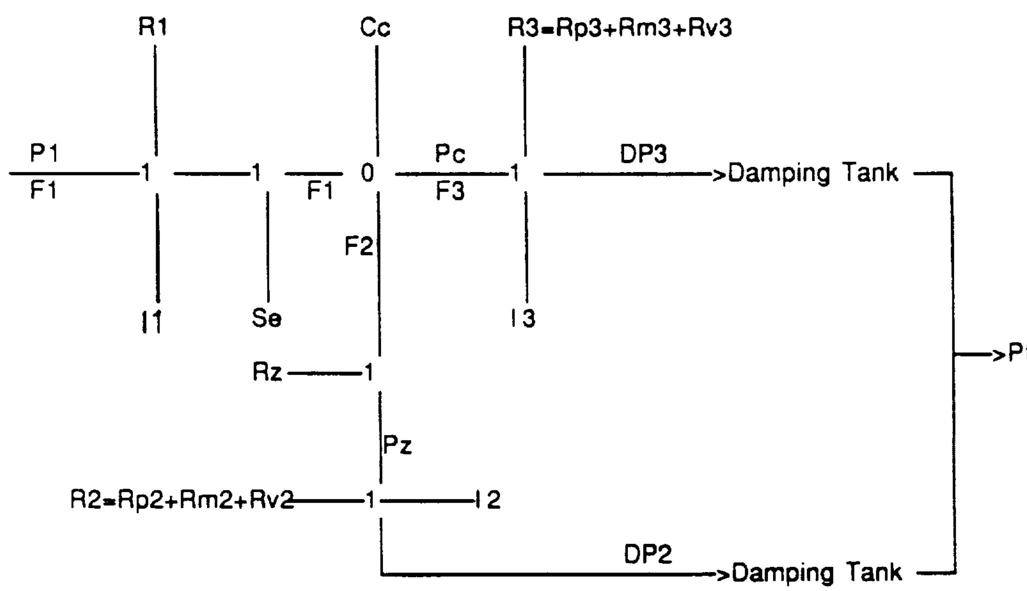


Fig. 4.2 Bond Graph Representation of The MOD System

----- Table 4.1 Nomenclature of Fig. 4.1 and 4.2 --------------------

Cc : fluid capacitance of the cell (sccm sec/mb)

D_{P1} : offset of linearized pressure-flowrate relation of pipeline #1

D_{P2} : offset of linearized pressure-flowrate relation of pipeline #2

D_{P3} : offset of linearized pressure-flowrate relation of pipeline #3

F1 : mass flow-rate of feed-gas (sccm)

H : transfer function of the mass flow-rate feedback (V/sccm)

I1 : fluid inertance of pipeline #1

I2 : fluid inertance of pipeline #2

I3 : fluid inertance of pipeline #3

K/s(s+p) : dynamic model of the flow controller

P1 : outlet pressure of the flow controller (mb)

Pd : delivery pressure (psig)

Pt : pressure of damping tank

R1 : overall fluid resistance of pipeline #1

R2 : overall fluid resistance of pipeline #2

R3 : overall fluid resistance of pipeline #3

R_{m2} : fluid resistance of flow meter #2

R_{m3} : fluid resistance of flow meter #3

R_{p2} : fluid resistance of pipeline segment #2

R_{p3} : fluid resistance of pipeline segment #3

R_{v2} : fluid resistance of metering valve #2

R_{v3} : fluid resistance of metering valve #3

Rz : simulant fluid resistance through the zirconia membrane

Se : external effort gained by height difference (mb)

Vs : mass flow-rate set point of feed-gas (Volts)

Derivation of system dynamic equations :

The dynamics of the mass flow controller can be modeled as a DC motor with flow rate set point input and fluid resistance output. The basic equation for the flow of a gas through an orifice is [29] :

$$F1 = Cd\ Ad\ f(Pd,\ Td,\ Pd/P1) \qquad (4.1\text{-}1)$$

where

F1 is mass flow rate

Cd is a dimensionless discharge coefficient

Ad is the orifice area which can be adjusted by the valve driven by a dc motor

Td is the upstream stagnation temperature

Pd is the upstream stagnation pressure (= delivery pressure)

P1 is the downstream pressure

The function f can be expressed as :

$$f = C1\ Pd/(Td)^{1/2}\ (P1/Pd)^{1/k}\ (\ 1 - (P1/Pd)^{(k-1)/k}\ )^{1/2} \qquad (4.1\text{-}2)$$

where C1 is function of k, the ratio of specific heats, and gas constant. For the MOD, we assume that Td = room temperature = constant, Cd=constant and k=1.3 for carbon dioxide. Then,

$$\begin{aligned} F1 &= Cd\ Ad\ C1\ Pd/(Td)^{1/2}\ (P1/Pd)^{1/k}\ (\ 1 - (P1/Pd)^{(k-1)/k}\ )^{1/2} \\ &= Ct\ Ad\ Pd\ (\ (P1/Pd)^{2/k} - (P1/Pd)^{(k+1)/k}\ )^{1/2} \\ &= Ct\ Ad\ Pd\ (\ (P1/Pd)^{2/1.3} - (P1/Pd)^{2.3/1.3}\ )^{1/2} \\ &= Ad\ G(Pd,\ P1) \qquad (4.1\text{-}3) \end{aligned}$$

where $G(Pd,P1) = Ct\ Pd\ (\ (P1/Pd)^{2/1.3} - (P1/Pd)^{2.3/1.3}\ )^{1/2}$

$Ct = Cd\ C1/(Td)^{1/2}$

Before moving on, we first justify whether Se (external effort gained by height difference) is negligible.

$$Se = \rho\ g\ h = (\ 0.213\ kg/m^3\ )\ (\ 9.8\ m/s^2\ )\ (\ 1.32\ m\ )$$

$$= 2.755\ N/m^2 = 0.0275\ mb$$

which is on the order of measurement noise and thus is negligible.

The flow controller is a typical feedback control system (see the block diagram in Fig. 4.1). Considering the dynamics of the motor, Vs is the input voltage and the output is the orifice area (Ad). The mass flow-rate can be obtained simply by multiplying Ad by the discharge coefficient (Cd) and the function (f). H is the feed-back transfer function. The equation has been simplified so that the overall transfer function is :

$$\frac{F1}{Vs} = \frac{K/S(S+P)\ G(Pd,\ P1)}{1 + K/S(S+P)\ H\ G(Pd,\ P1)} = \frac{K\ G(Pd,\ P1)}{S^2 + S\ P + K\ H\ G(Pd,\ P1)}$$

with $\ddot{F1} + P\,\dot{F1} + K\ H\ G(Pd,\ P1)\ F1 = K\ Vs\ G(Pd,\ P1)$, (4.1-4)

where $F1_d = \dot{F1}$, so that Eq. 4.1-4 becomes:

$$\dot{F1}_d = K\ Vs\ G(Pd,\ P1) - P\ F1_d - K\ H\ G(Pd,\ P1)\ F1 \qquad (4.1\text{-}5)$$

Based on Fig. 4.2, the primary variables are chosen to be Pc, P1, F1, $F1_d$, F2 and F3.

$$\dot{Pc} = (1/Cc)F1 - (1/Cc)F2 - (1/Cc)F3 \qquad (4.1\text{-}6)$$

$$\dot{F1} = F1_d \qquad (4.1\text{-}7)$$

$$\dot{F1}_d = K\ Vs\ G(Pd,\ P1) - P\ F1_d - K\ H\ G(Pd,\ P1)\ F1 \qquad (4.1\text{-}8)$$

where $G(Pd,P1) = Ct\ Pd(\ (P1/Pd)^{2/1.3} - (P1/Pd)^{2.3/1.3}\ )^{1/2}$

$$P1 = Pc + R1\ F1 + D_{P1} + I1\ F1_d \qquad (4.1\text{-}9)$$

$$\dot{F3} = (1/I3)Pc - (1/I3)(R_{p3} + R_{m3} + R_{v3})F3 - (1/I3)D_{P3}$$

$$= (1/I3)[\ Pc - R3\ F3 - D_{P3}\ ] \qquad (4.1\text{-}10)$$

F2 = Iz/(4 F) gm-mole/sec = [Iz/(4 F)]*32*60*60 gm/hr

= {Vz - (R Tc)/(4 F) ln[Pz/(Pc*f)]}*115200/(4 F Rz) gm/hr ----- (4.1-11)

where F is Faraday's constant (96484.56 Amp.S/mole)

R is gas constant (8.31434 J/(gm-mole K))

f is the mole fraction of carbon dioxide dissociated

f and Rz are both functions of Tc

D_{P1}, D_{P2} and D_{P3} are constants and can be obtained by experiment

The latter part of Eq. 4.1-11 is derived from Eq. 2.2-2

$$Pz = (R_{p2} + R_{m2} + R_{v2})F2 + D_{P2} + I2\ \dot{F}2$$

or $$Pz = R2\ F2 + D_{P2} + I2\ \dot{F}2$$

$$= R2\ F2 + D_{P2} \quad \text{(assume I2 } \dot{F}2 \text{ is negligible)} \qquad (4.1\text{-}12)$$

In order to obtain the dynamics of F2, we may take the time derivative of Eq. 4.1-11

$$\dot{F}2 = \{Vz - (R\ Tc)/(4\ F)\ \ln[Pz/(Pc*f)]\}'*115200/(4\ F\ Rz) \qquad \text{-----}\ (4.1\text{-}13)$$

If Vz is supplied by an independent, constant-voltage power supply then Vz is constant and $\dot{V}z = 0$, then

$$\dot{F}2 = \{-(R\ Tc)/(4\ F)\ [(Pc*f)/Pz\ (Pz/Pc*f)']\}*115200/(4\ F\ Rz)$$

Substitution of Eq. 4.1-12 for Pz and assuming f is constant, then

$$\dot{F}2 = \{-(R\ Tc)/(4\ F)\ [(R2\ F2 + D_{P2})(Pc*f)]^{-1}[(Pc*f)(R2\ F2 + D_{P2})'$$

$$- (R2\ F2 + D_{P2})(Pc*f)']\}*115200/(4\ F\ Rz)$$

$$= -(R\ Tc)*115200/(16\ F^2\ Rz)\ [(R2\ \dot{F}2)/(R2\ F2 + D_{P2})$$

$$- (\dot{P}c/Pc)]$$

$$= -C(Tc,Rz)\ [R2/(R2\ F2 + D_{P2})\ \dot{F}2 - (1/Pc)\ \dot{P}c]$$

where $C(Tc,Rz) = (R.Tc)*115200/(16\ F^2\ Rz) > 0$

$$\{C(Tc,Rz)\ [R2/(R2\ F2 + D_{P2})] + 1\}\ \dot{F}2 = C(Tc,Rz)\ [\dot{P}c/Pc]$$

$$\dot{F}2 = C(Tc,Rz)\ [\dot{P}c/Pc]/\{C(Tc,Rz)\ [R2/(R2\ F2 + D_{P2})] + 1\}$$

$$\dot{F2} = \frac{C(Tc,Rz)\ [(1/Cc)(F1 - F2 - F3)/Pc]}{C(Tc,Rz)\ [R2/(R2\ F2 + D_{P2})] + 1} \qquad (4.1\text{-}14)$$

However, this equation is not quite correct because Eq.(4.1-11) can be satisfied only when the system reaches steady-state. Eq.(4.1-13) thus is a hypothesis only. Observation from experiments shows that the rate of change of oxygen flow-rate as the system states change is very slow. An accurate dynamic model of the electro-chemical reaction of the zirconia cell itself has to be studied in the future.

Although F2 cannot be one of the system states of the dynamic model, we can linearize Eq.(4.1-11) about the normal operating set points of Pz and Pc by using Taylor Series and solve for F2 as the system output for simulation purposes. This is demonstrated as follows.

$$\begin{aligned} F2 &= \{Vz - (R\ Tc)/(4\ F)\ \ln[Pz/(Pc*f)]\}*115200/(4\ F\ Rz) \\ &= \{Vz - (R\ Tc)/(4\ F)\ [\ln(Pz) - \ln(Pc*f)]\}*115200/(4\ F\ Rz) \\ &= \{Vz - (R\ Tc)/(4\ F)\ [\ln(MPz) + (1/MPz)(Pz - MPz) - \ln(MPcf) \\ &\quad - (1/MPcf)(Pc*f - MPcf)]\}*115200/(4\ F\ Rz) \end{aligned}$$

where MPz is the mean operating value of Pz

MPcf is the mean operating value of Pc*f

$$F2 = Vz*115200/(4\ F\ Rz) - C(Tc,Rz)[\ln(MPz) + (1/MPz)(R2\ F2 + D_{P2} - MPz) - \ln(MPcf) - (1/MPcf)(Pc*f - MPcf)]$$

$$\begin{aligned} F2[1 + C(Tc,Rz)(R2/MPz)] = {} & Vz*115200/(4\ F\ Rz) - C(Tc,Rz)[\ln(MPz) \\ & + (1/MPz)(D_{P2} - MPz) - \ln(MPcf) \\ & - (1/MPcf)(Pc*f - MPcf)] \end{aligned}$$

$$F2 = \frac{C'(Vz,Tc,Rz,MPz,DP2,MPcf) + C(Tc,Rz)(1/MPcf)(Pc*f - MPcf)}{[1 + C(Tc,Rz)(R2/MPz)]}$$

or $F2 = f_{F2}(Pc) = a\ Pc + b$ (4.1-15)

where $C'(Vz,Tc,Rz,MPz,DP2,MPcf) = Vz*115200/(4\ F\ Rz) - C(Tc,Rz)$

$*[\ln(MPz) + (1/MPz)(D_{P2} - MPz) - \ln(MPcf)]$

a and b are constants

Because the rate of change of F2 as the system states change is very slow and small compared with those of F1 and F3, Eq.(4.1-6) can be modified to be:

$$\dot{Pc} = (1/Cc)\ [\ F1 - MF2 - F3\ ] \qquad (4.1\text{-}16)$$

where MF2 is the mean operating value of F2.

Therefore, the system dynamics can be fully described by four state variables (Pc, F1, F1d, and F3), four state equations (Eq. 4.1-7, 4.1-8, 4.1-10, 4.1-16) and one output equation (Eq. 4.1-15).

In addition, the temperature keeping subsystem can be modeled as shown in Fig. 4.3 which shows a basic thermal system consisting of a mass of material which is assumed to have a nearly uniform temperature, Tc, and thermal capacity C [30]. It is surrounded by insulation with thermal resistance R. We neglect any capacitance in the insulation, but in R we include the resistance associated with the two interfaces between the insulation and the block of material and the insulation and the surrounding atmosphere at temperature Ta. We assume that ambient temperature remains at temperature Ta even if heat flows to and from the surroundings. An electric-resistance heating element is embedded in the block. From the electrical point of view, the power dissipated is e . i, and from the thermal point of view this power becomes the source flow Qs'. The pseudo bond graph is assembled also in Fig. 4.3.

Based on the Bond graph and the on-off control nature of the oven controller, the following two dynamic equations and control laws can be derived [30].

$$\dot{Tc} = -(1/R\ C)(Tc - Ta) + ie/C \qquad \text{when } Tc < Ts - dT$$

$$\dot{Tc} = -(1/R\ C)(Tc - Ta) \qquad \text{when } Tc > Ts + dT$$

where Ts is the temperature set-point and dT is the deviation threshold for on-off control.

Determination of System Process Parameters :

Determination of process parameters includes determination of the values of Cc, K, P, H, Ct, Pd, R1, D_{P1}, I1, R3, D_{P3}, I3, Vz, Tc, Rz, R2, D_{P2}, f, ...etc. They have to be evaluated through experimental tests. Testing procedures and results are given in Chapter 5. Numerical simulations are also provided.

4.2 Process Fault Detection Based on Modeling and Estimation

A method presented by Rolf Isermann in 1983 [31] can be applied to detect and isolate system failure using least-squares parameter estimation if the system can be described by linear differential equations.

$$X' = f(M, X, U) \quad (4.2\text{-}1)$$

$$Y = g(M, X) \quad (4.2\text{-}2)$$

where M are the model parameters and X are the state variables
U are the input variables and Y are the output variables

The relationship between the model parameters M and the physical process coefficients P is:

$$M = h(P) \quad (4.2\text{-}3)$$

Measurements of the input, state and output signals are made and all required derivatives are determined at discrete times $t_k = k\ T_0$, $k=0,1,2,...N$ with T_0 the sampling time. The model parameters can be estimated as a result of measurements of the signals. Then the process coefficients P are calculated and their changes can be determined from

$$P = h^{-1}(M) \quad (4.2\text{-}4)$$

Possible process faults can be pinpointed by pattern recognition in which the relationship between process faults and changes in the coefficients has been established.

Hence, the basis of this method is the combination of theoretical modeling and parameter estimation. A block diagram is given in Fig. 4.4.

If there is nonlinearity in Eq.(4.2-1), we may either use a priori probablistic information and apply an Extended Kalman Filter to estimate the model parameters or try to linearize the equations and apply Least-squares estimation techniques. The second approach was used here.

Rewrite the four state equations and one output equation of MOD system as follows.

$$\dot{Pc} = (1/Cc)\ [\ F1 - MF2 - F3\ ] \tag{4.2-5}$$

$$\dot{F1} = F1_d \tag{4.2-6}$$

$$\dot{F1_d} = K\ Vs\ G(Pd,P1) - P\ F1_d - K\ H\ G(Pd,P1)\ F1 \tag{4.2-7}$$

$$\dot{F3} = (1/I3)\ [\ Pc - R3\ F3 - D_{P3}\ ] \tag{4.2-8}$$

$$F2 = a\ Pc + b \tag{4.2-9}$$

Because Pd is constant and P1 is a function of Pc only, G(Pd,P1) can be replaced by G(Pc). Linearize G(Pc) about the operating set point of Pc (i.e. MPc) using a Taylor Series. Then, Eq.(4.2-7) becomes

$$\dot{F1_d} = K\ Vs\ [G(MPc) + \dot{G}(MPc)(Pc - MPc)] - P\ F1_d$$

$$-K\ H\ F1\ [G(MPc) - \dot{G}(MPc)(Pc - MPc)] \tag{4.2-10}$$

The assumption of small deviations around the steady-state (MPc, MF1, $MF1_d$, MF3, MF2) leads to

$$Pc = MPc + \delta Pc \tag{4.2-11}$$

$$F1 = MF1 + \delta F1 \tag{4.2-12}$$

$$F1_d = MF1_d + \delta F1_d \tag{4.2-13}$$

$$F3 = MF3 + \delta F3 \tag{4.2-14}$$

$$F2 = MF2 + \delta F2 \tag{4.2-15}$$

$$Vs = MVs + \delta Vs \tag{4.2-16}$$

Substitute these equations into the system state equations and neglect the higher order terms to get:

$$\dot{Pc} = (1/Cc)\ [\ \delta F1 - \delta F3\ ] \tag{4.2-17}$$

$$\dot{F1} = \delta F1_d \tag{4.2-18}$$

$$\dot{F1}_d = K(MVs + \delta Vs)[G(MPc) + \dot{G}(MPc)(MPc + \delta Pc - MPc)] - P(MF1_d$$

$$+ \delta F1_d) - K\ H\ (MF1 + \delta F1)[G(MPc) + \dot{G}(MPc)(MPc + \delta Pc - MPc)]$$

$$\dot{F1}_d = K\ MVs\ \dot{G}(MPc)\ \delta Pc + K\ G(MPc)\ \delta Vs - P\ \delta F1_d$$

$$- K\ H\ MF1\ \dot{G}(MPc)\ \delta Pc - K\ H\ G(MPc)\ \delta F1 \tag{4.2-19}$$

$$\dot{F3} = (1/I3)\ [\ \delta Pc - R3\ \delta F3\ ] \tag{4.2-20}$$

$$\delta F2 = a\ \delta Pc \tag{4.2-21}$$

Write the result in state space form as :

$$\dot{X} = A\ \delta X + B\ \delta U$$

where

$$\dot{X}^T = [\ \dot{Pc}\quad \dot{F1}\quad \dot{F1}_d\quad \dot{F3}\ ]$$

$$\delta X = [\ \delta Pc\quad \delta F1\quad \delta F1_d\quad \delta F3\ ]$$

$$\delta U = \delta Vs$$

$$A = \begin{bmatrix} 0 & a(1,2) & 0 & a(1,4) \\ 0 & 0 & 1 & 0 \\ a(3,1) & a(3,2) & a(3,3) & 0 \\ a(4,1) & 0 & 0 & a(4,4) \end{bmatrix}$$

where $a(1,2) = 1/Cc$

$a(1,4) = -(1/Cc)$

$$a(2,3) = 1$$

$$a(3,1) = K\ MVs\ \dot{G}(MPc) - K\ H\ MF1\ \dot{G}(MPc)$$

$$a(3,2) = -\ K\ H\ G(MPc)$$

$$a(3,3) = -\ P$$

$$a(4,1) = 1/I3$$

$$a(4,4) = -(R3/I3)$$

$$B^T = [\ 0 \quad 0 \quad b(3,1) \quad 0\ ]$$

where $b(3,1) = K\ G(MPc)$

Rewrite Eq. 4.2-17 to Eq. 4.2-20 in the following forms

$$\dot{Pc} = [\ \delta F1 \quad \delta F3\ ]\ [\ a(1,2) \quad a(1,4)\ ]^T \qquad (4.2\text{-}22)$$

$$\dot{F1} = [\ 1\ ]\ [\ \delta F1_d\ ]^T \qquad (4.2\text{-}23)$$

$$\dot{F1}_d = [\ \delta Pc \quad \delta F1 \quad \delta F1_d \quad \delta Vs\ ]\ [\ a(3,1) \quad a(3,2) \quad a(3,3) \quad b(3,1)\ ]^T$$

----- (4.2-24)

$$\dot{F3} = [\ \delta Pc \quad \delta F3\ ]\ [\ a(4,1) \quad a(4,4)\ ]^T \qquad (4.2\text{-}25)$$

From Eq. 4.1-9, $P1 = Pc + R1\ F1 + D_{P1} + I1\ F1_d$

$$\delta P1 = [\ \delta Pc \quad \delta F1 \quad \delta F1_d\ ]\ [\ p(1,1) \quad p(2,1) \quad p(3,1)\ ]^T \qquad (4.2\text{-}26)$$

where $p(1,1) = 1$; $p(2,1) = R1$; $p(3,1) = I1$

From Eq. 4.1-10, $F2 = a\ Pc + b$

$$\delta F2 = [\ \delta Pc\ ]\ [\ q(1,1)\ ]^T \qquad (4.2\text{-}27)$$

where $q(1,1) = a$

Now, measurements of the input and output signals are made and all required derivatives are determined at discrete time $t_k = k\ T_0$, $k=0,1,2,\ \ldots,N$ with T_0 the sampling time. Then N+1 equations from Eq. 4.2-22 to Eq. 4.2-27 result in the following general form.

$$Y_n = X_n^T\ M_n$$

where Mn (n=1 to 4) are the model parameter vectors.

The least-squares estimate of the parameter vector becomes the well-known nonrecursive estimation equation.

$$M_n = [X_n^T X_n]^{-1} X_n^T Y_n$$

The physical process coefficients can be estimated using the following relationship :

Cc = 1/a(1,2)

K = b(3,1)/G(MPc)

H = -a(3,2)/b(3,1)

P = -a(3,3)

I3 = 1/a(4,1)

R3 = -a(4,4)/a(4,1)

I1 = p(3,1)

R1 = p(2,1)

a = q(1,1)

Deviation of a certain estimated process coefficient from their normal values may indicate the existence of a correspondent failure mode. For example, a large deviation of estimated Cc means a change of the cell capacitance which might result from cell leakage. Rules and thresholds have to be set to justify each individual failure mode.

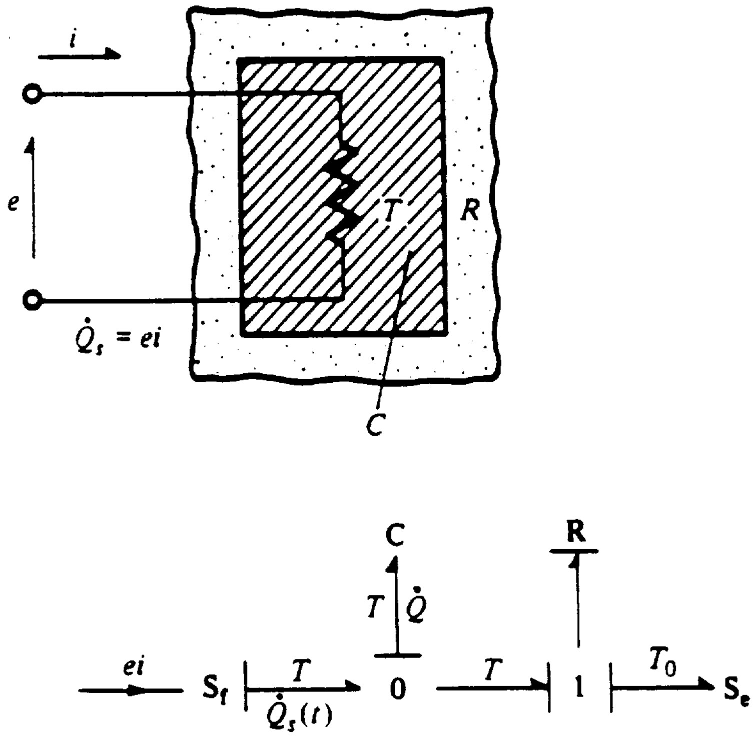


Fig. 4.3 Model of The Temperature Keeping Subsystem [30]

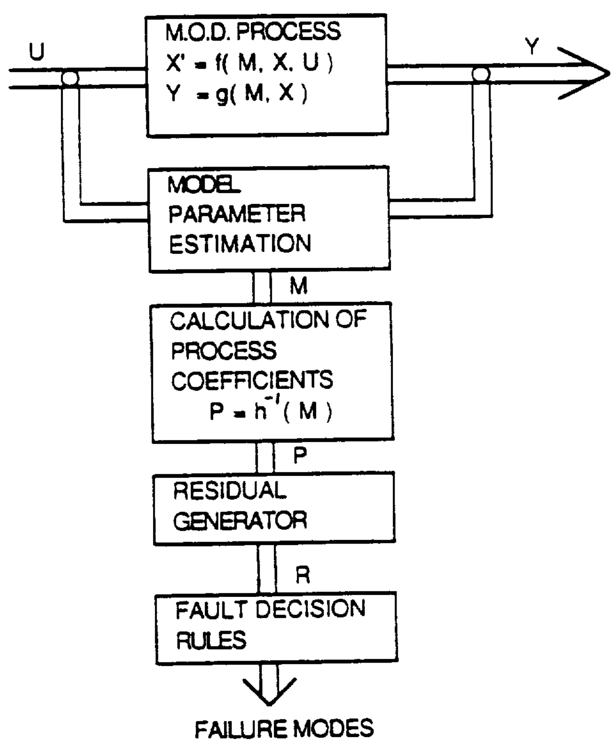


Fig. 4.4 Block Diagram of The Model Based Process Fault Detection [31]

Chapter 5

Experimental Results And Numerical Simulations

5.1 Preliminary Evaluation of System Performance

The whole MOD system had been assembled and set up completely by August 1988. System performance has been evaluated by operating the system using different input parameter sets since September, 1988. Table 5.1 lists the possible values that have been used for each individual input parameter (refer to Section 3.1).

Table 5.1 Values for Each Input Parameter

Tc	900 K,	1000 K,	1200K		
Vz	1.0 V,	1.2 V,	1.5 V,	2.0 V	
F1	50 sccm,	100 sccm,	150 sccm		
Pd	40 psig				
Pc	75 mb,	115 mb,	125 mb,	150 mb	
Pz	50 mb,	75 mb,	115 mb,	125 mb,	150 mb

At the beginning, it was observed that the current through the zirconia tubular membrane decreased monotonically over each 24 hour operating interval. Figure 5.1-1 shows a typical example of such a problem. After getting advice from Cermatic Inc., another test was performed by running the system without turning the vacuum pump on. Figure 5.1-2 shows how the current dropped as well as fluctuated. Reasons for that are not known and still being studied. The following possible reasons are considered.

1. Overpotential

2. Ionization/Polarization

3. Oxygen partial pressure

Instead of using carbon dioxide, simulated Mars gas was used as feed-gas for testing (in October, 1988). The stability of current output was improved (see Fig. 5.1-3), and might be caused by the 0.13 % of oxygen in the simulated Mars gas which helps to stabilize the current output. Figure 5.1-4 which follows directly from Fig. 5.1-3 shows the result of a proposed cell repair test which was implemented by reversing the cell polarity and conducting oxygen back into the feed-gas stream for 30 miniutes [32]. No observable advantage was achieved by reversing the potential.

Figure 5.1-5 shows the results obtained in November, 1988 from setting Pc = P2 = 150mb. Figure 5.1-6 has the same initialization system parameters except that cell #1 was used instead of cell #2.

The system operating condition and parameters which would be used for testing of the Expert System were finally determined in December 1988. Experience showed that current can be stablized faster by adjusting the metering valve in the oxygen line wide-open. Improved transient behavior is probably due to the fact that the air which is initially in the oxygen line can be drawn out faster so that the partial pressure of the oxygen line can reach total pressure more rapidly. The system operating parameters are listed as follows :

Tc = 1200 K, Vz = 1.0 V, F1 = 50 sccm, F2 = 0.32 sccm, F3 = F1,
Iz = 150 mA, S1$ = P, Pp = Pr > 100 psi, Pd = 40 psi,
Pc = 125 mb, Pz = 50 mb, P1 = 190 mb, P2 = 0 mb, P3 = 100 mb, Pt= 0 mb

---

Figure 5.1-7 shows the corresponding current response over a 24 hour operating period. Figure 5.1-8 shows the result obtained using the same operating condition as that of Fig. 5.1-7 except introducing the feed-gas through a preheating pipe-coil before conducting the feed-gas into the cell.

A leakage at the cell cap was identified and fixed in March 1989. In the meantime, we consulted Dr. K.G. Brown in the Chemistry Department about the cell performance. He gave us advice that changed our operating philosophy from conservative to nonconservative. He told us that we can increase the cell voltage, due to the fact that pure oxygen in the oxygen line and pure carbon dioxide in the feed-gas line make the pressure potential (Eq. 2.2-2) very high. We have to overcome the pressure potential before the whole applied voltage can be used to generate oxygen. Therefore we increased the cell voltage step by step. Figures 5.1-9 to 5.1-12 show the current and/or conversion efficiency versus cell voltage plots which were obtained at the end of April 1989.

All the experimental or simulation results given in the following sections were obtained before March 1989.

5.2 Statistical Analysis of System Parameters

In order to determine the failure thresholds for the Expert System, the normal range of each sensor output under normal operating condition has to be examined first. This job has been done by recording all the sensor outputs over a 24 hour operating period. The mean value and standard deviation (sd) of each sensor output are calculated and listed in Table 5.2. Figures 5.2a to 5.51 shows how well the system parameters can be maintained.

Table 5.2 Statistics of Each Individual Sensor Output

sensor	Tc(K)	Vz(v)	F1(gm/hr)	F2(gm/hr)	F3(gm/hr)	Pp(psig)
mean	1200.241	1.014947	5.406548	0.026945	4.726044	934.0432
sd	1.358235	0.003172	0.038737	0.000856	0.030729	7.799892

sensor	Pc(mb)	Pc(mb)	Pz(mb)	P1(mb)	P2(v)	P3(mb)
mean	124.3519	127.3614	48.89628	192.3832	7.374239	99.79853
sd	1.177941	0.295978	1.113997	1.301610	0.235801	1.449098

Note : 1. Refer to Fig. 2.1 for symbolic representation.

2. Pc has two sets of data.

3. P1, P2, and P3 are all regular pressure transducers which measure gage pressure instead of absolute pressure. Their readings may vary with barometric pressure.

5.3 Testing Results of Individual Failure Mode

Tests of each individual failure mode have been performed. The tests are divided into the following categories :

Tests of failure modes G01 & G02 (Tc too high or too low)

G01 was not tested because excessive high temperature levels might cause permanent cell failure.

G02 was tested by decreasing the temperature set point of the temperature controller intentionally. Three noticeable changes in system parameters (Iz, Vz and F2) occured as Tc was decreased and were recorded (see Fig. 5.3-1 to 5.3-3). The linear least-squares fitting equations are

$$\delta Iz(mA) = 0.193089\ \delta Tc(K) + 6.789808$$

$$\delta Vz(mV) = -1.932058\ \delta Tc(K) - 71.394406$$

$$\delta F2(1.0E\text{-}3\ gm/hr) = 0.19321\ \delta Tc(K) + 8.72695$$

Tests of failure modes G03 & G04 (Vz too high or too low)

G03 was tested by increasing the cell voltage intentionally. Three noticeable changes of system parameters (Iz, F2 and Pz) occured as Vz was increased and were recorded (see Fig. 5.3-4 to 5.3-6). The linear least-squares fitting equations are

$$\delta Iz(mA) = 0.134534\ \delta Vz(mV) - 0.041194$$

$$\delta F2(1.0E\text{-}3\ gm/hr) = 0.01505\ \delta Vz(mV) - 0.925258$$

$$\delta Pz(mb) = 0.014312\ \delta Vz(mV) - 0.473267$$

G04 was tested by decreasing the cell voltage intentionally. Three noticeable changes of system parameters (Iz, F2 and Pz) occured as Vz was decreased and were recorded (see Fig. 5.3-7 to 5.3-9). The linear least-squares fitting equations are

$\delta Iz(mA) = 0.131814\ \delta Vz(mV) - 0.276796$

$\delta F2(1.0E\text{-}3\ gm/hr) = 0.020841\ \delta Vz(mV) - 1.401878$

$\delta Pz(mb) = 0.025836\ \delta Vz(mV) - 0.479174$

Tests of failure modes G13 & G14 (F1 too high or too low)

G13 was tested by increasing the flow-rate set-point intentionally. Three noticeable changes of system parameters (P1, F3 and P3) occured as F1 was increased and were recorded (see Fig. 5.3-10 to 5.3-12). The linear least-squares fitting equations are

$\delta P1(mb) = 0.175738\ \delta F1(1.0E\text{-}2\ gm/hr) + 1.876937$

$\delta F3(1.0E\text{-}2\ gm/hr) = 1.043657\ \delta F1(1.0E\text{-}2\ gm/hr) - 1.803987$

$\delta P3(mb) = 0.081369\ \delta F1(1.0E\text{-}2\ gm/hr) + 0.845773$

G14 was tested by decreasing the flow-rate set-point intentionally. Four noticeable changes of system parameters (P1, F3, P3 and Pc) occured as F1 was decreased and were recorded (see Fig. 5.3-13 to 5.3-16). The linear least-squares fitting equations are

$\delta P1(mb) = 0.296773\ \delta F1(1.0E\text{-}2\ gm/hr) + 7.805504$

$\delta F3(1.0E\text{-}2\ gm/hr) = 0.807503\ \delta F1(1.0E\text{-}2\ gm/hr) - 20.774352$

$\delta P3(mb) = 0.133342\ \delta F1(1.0E\text{-}2\ gm/hr) + 2.666432$

$\delta Pc(mb) = 0.159179\ \delta F1(1.0E\text{-}2\ gm/hr) + 2.711193$

Tests of failure modes G15 & G16 (Vw set-point too high or too low)

G15 was tested by adjusting the metering valve for the waste-gas pipeline more open than the original set-point intentionally. Two noticeable changes of system parameters (P3 and Pc) occured as P1 was changed and were recorded (see Fig. 5.3-17 to 5.3-18). The linear least-squares fitting equations are

$\delta P3(mb) = 1.719925\ \delta P1(mb) + 0.32853$

$\delta Pc(mb) = 1.232156\ \delta P1(mb) + 0.259908$

G16 was tested by adjusting the metering valve for the waste-gas pipeline more closed than the original set-point intentionally. Two noticeable changes of system parameters (P3 and Pc) as P1 was changing were recorded (see Fig. 5.3-19 to 5.3-20). The linear least-squares fitting equations are

$\delta P3(mb) = 1.817595\ \delta P1(mb) + 0.931269$

$\delta Pc(mb) = 1.19001\ \delta P1(mb) + 0.06019$

Tests of failure mode L01 (Leakage at feed-gas pipeline)

L01 was tested by slightly loosening one fitting screw along the feed-gas pipeline intentionally. Three noticeable changes of system parameters (P3, F3 and Iz) occured as P1 was changed and were recorded (see Fig. 5.3-21 to 5.3-23). The linear least-squares fitting equations are

$\delta P3(mb) = 0.431824\ \delta P1(mb) - 1.042762$

$\delta F3(1.0E\text{-}2\ gm/hr) = 2.992506\ \delta P1(mb) - 7.282831$

$\delta Iz(mA) = 0.064961\ \delta P1(mb) - 0.26193$

Tests of failure mode G17 (Vo set-point too high)

G17 was tested by adjusting the metering valve on the oxygen pipeline more open than the original set-point intentionally. Changes in Pz, as P2 was changing, were recorded (see Fig. 5.3-24). The linear least-squares fitting equation is

$\delta Pz(mb) = 0.568362\ \delta P2(mb) - 1.958184$

5.4 Experimental Data of System Dynamics

In order to apply the model-based process fault detection technique presented in Chapter 4, a good mathematical model of the system is required. The simplified model using the Bond graph technique was also described in Chapter 4. The validity of that model can be examined through a comparison between the experimental data and the simulation results of the model. The responses of some system parameters when the flow-rate set-point of the flow controller changed from one value to another are compared between experiments and simulations. The change of the F1 set-point is treated as a step input to the system. Figures 5.4(a to f) show the experimental results of the following cases :

(a) F1 set-point is changed from 50 sccm to 0 sccm (Fig. 5.4a)

(b) F1 set-point is changed from 50 sccm to 25 sccm (Fig. 5.4b)

(c) F1 set-point is changed from 25 sccm to 0 sccm (Fig. 5.4c)

(d) F1 set-point is changed from 25 sccm to 50 sccm (Fig. 5.4d)

(e) F1 set-point is changed from 0 sccm to 25 sccm (Fig. 5.4e)

(f) F1 set-point is changed from 0 sccm to 50 sccm (Fig. 5.4f)

Note : 1 sccm = 0.108 gm/hr (for CO_2 gas)

5.5 Simulation Results of System Model

As presented in Chapter 4, the system dynamics can be described by the following four equations with for state variables (Pc, F1, F1d and F3).

$$\dot{Pc} = (1/Cc)\ [\ F1 - MF2 - F3\ ] \qquad (5.5\text{-}1)$$

$$\dot{F1} = F1_d \qquad (5.5\text{-}2)$$

$$\dot{F1}_d = K\ Vs\ G(Pc)\ -\ P\ F1_d\ -\ K\ H\ G(Pc)\ F1 \qquad (5.5\text{-}3)$$

$$\dot{F3} = (1/I3)\ [\ Pc - R3\ F3 - D_{P3}\ ] \qquad (5.5\text{-}4)$$

where $G(Pc) = Ct\ Pd(\ (P1/Pd)^{2/1.3} - (P1/Pd)^{2.3/1.3}\ )^{1/2}$

$P1 = Pc + R1\ F1 + D_{P1} + I1\ F1_d$

In order to perform numerical simulation, the values of process parameters have to be determined by the following procedures.

Determination of Cell Capacitance Cc :

From Eq. 5.5-1, the capacitance, Cc, can be defined as the change in the mass of CO_2 in the cell required to make a unit change in pressure. A Test was performed by introducing a constant flow-rate (10 sccm) of CO_2 gas into the cell with two outlets closed (oxygen outlet and waste gas outlet) and recording the pressure increase in the cell. Figure 5.5-1 shows the result. The rate of pressure increase is

$$\dot{Pc} = (9.008547 - 1.477412)/(174.56 - 34.92)$$

$$= 0.05393\ V/sec = 0.5393\ torr/sec = 0.7190\ mb/sec$$

$$0.7190\ mb/sec = (1/Cc)*10\ sccm$$

$$C_c = 13.9082\ (\text{sccm})/(\text{mb/sec}) = 1.502\ (\text{gm/hr})/(\text{mb/sec})$$

Determination of the values of Ct, H, K and P :

For Eq. 5.5-3, we have to know the dynamic characteristics of the flow controller which is modeled as a valve actuated by a dc motor. The output flow-rate is affected not only by the valve position but also by the pressure conditions on both sides of the controller. Therefore, it is difficult to obtain an accurate model unless we could take the unit apart and examine the motor dynamics itself. Here, we assume that the flow controller with a transfer function of the following standard second-order form has 10% overshoot and 1.0sec settling time if the pressure conditions can be held constant at the normal operating pressure set-point. That is,

$$\frac{F1(V)}{Vs(V)} = \frac{Wn^2}{S^2 + 2\zeta\, Wn\, S + Wn^2}$$

according to the specifications,

$$Mp = \exp[-\pi\,(\zeta/(1-\zeta^2)^{1/2})] = 10\% \qquad \zeta = 0.5912$$

$$Ts = 4/(\zeta\, Wn) = 1.0 \qquad Wn = 6.7659$$

therefore

$$\frac{F1(V)}{Vs(V)} = \frac{45.7774}{S^2 + 8\,S + 45.7774} \tag{5.5-5}$$

$$= \frac{F1(\text{gm/hr})\ H}{Vs(V)} = \frac{K\ G(Pd,P1)\cdot H}{S^2 + P\,S + K\,H\,G(Pd,P1)} \tag{5.5-6}$$

where $H = 0.1225\ V/(\text{gm/hr})$

$$G(Pd,P1) = Ct\ Pd(\ (P1/Pd)^{2/1.3} - (P1/Pd)^{2.3/1.3}\ )^{0.5}$$

The operating set-point of Pd is 40 psig (or 2757.92 mb).

Also, from the block diagram shown in Fig. 4.1

$$F1\ (\text{gm/hr}) = Ad\ G(Pd,P1)$$

The current operating set-points for F1 (5.47 gm/hr) and P1 (135 mb) are considered to be a special case for computing Ct.

F1 (gm/hr) = 5.47 (gm/hr) = Ad G(Pd,P1)

assign arbitrarily Ad = 1 (unit of area)

$Ct*2757.92[(135/2757.92)^{2/1.3} - (135/2757.92)^{2.3/1.3}]^{0.5} = 5.47$

Ct = 0.02852

compare the numerators of Eq. 5.5-6 and 5.5-7

45.7774 = K G(Pd,P1) H = K * 5.47 * 0.1225

K = 68.2995

P = 8.0

Determination of the values of R1, D_{P1} and I1 :

P1 can be expressed in terms of Pc, F1 and $F1_d$

$P1 = Pc + R1\ F1 + D_{P1} + I1\ F1_d$

For steady state

$P1_{ss} - Pc_{ss} = R1\ F1 + D_{P1}$

Figure 5-5.2 shows that both $P1_{ss}$ and Pc_{ss} are linear functions of F1. R1 and D_{P1} can be figured out from the least-squares linear fitting equations of Fig. 5.5-2a and 5.5-2b.

R1 = 1.8284

$D_{P1} = 0.69979$

The gas inertance in a pipe refers to the change in pressure required to make a unit rate of change in mass flow-rate and can be evaluated by dividing the length of the pipe by its cross section area [34]. In this case,

I1 = L1 / A

where L1 = length of feed-gas pipeline = 70 in

$A = \pi\ (0.18/2)^2\ in^2$

$I1 = 2750.8262\ in^{-1} = 108302.3695\ m^{-1} = 3.0084E-4$ (mb sec/gm hr)

$P1(mb) = Pc(mb) + 1.8284\ F1(gm/hr) + 0.69979 + 3.0084E-4\ F1_d$

Determination of the values of R3, D_{P3} and I3 :

R3 and D_{P3} can be figured out from Fig. 5.5-2c and 5.5-2d.

R3 = 15.9179

D_{P3} = 39.6636

I3 is evaluated by the same way as for I1.

I3 = L3 / A

where L3 = length of waste-gas pipeline = 112 in

$A = \pi\ (0.18/2)^2\ in^2$

I3 = 4.8134E-4 (mb sec/gm hr)

Finally, the following four equations are ready for computer simulation.

$$\dot{Pc} = (1/1.502)\ [\ F1 - MF2 - F3\ ]$$

$$\dot{F1} = F1_d$$

$$\dot{F1_d} = 68.2995\ Vs\ G(P1) - 8.0\ F1_d - 68.2995*0.1225\ G(P1)\ F1$$

$$\dot{F3} = (1/4.8134E-4)\ [\ Pc - 15.9179\ F3 - 39.6636\ ]$$

where

$$G(P1) = 0.02852*2757.92*[(P1/2757.92)^{2/1.3} - (P1/2757.92)^{2.3/1.3}\ )^{0.5}$$

$$P1 = Pc + 1.8284\ F1 + 0.69979 + 3.0084E-4\ F1_d$$

A software called "DE" is used to realize the computer simulations for this case. "DE" has to be linked with "MODYNA1.FOR" to generate an executable program file. The numerical simulation results are shown in Fig. 5.5-3a to Fig. 5.5-3f). Simulation results are comparable to those obtained from experiments.

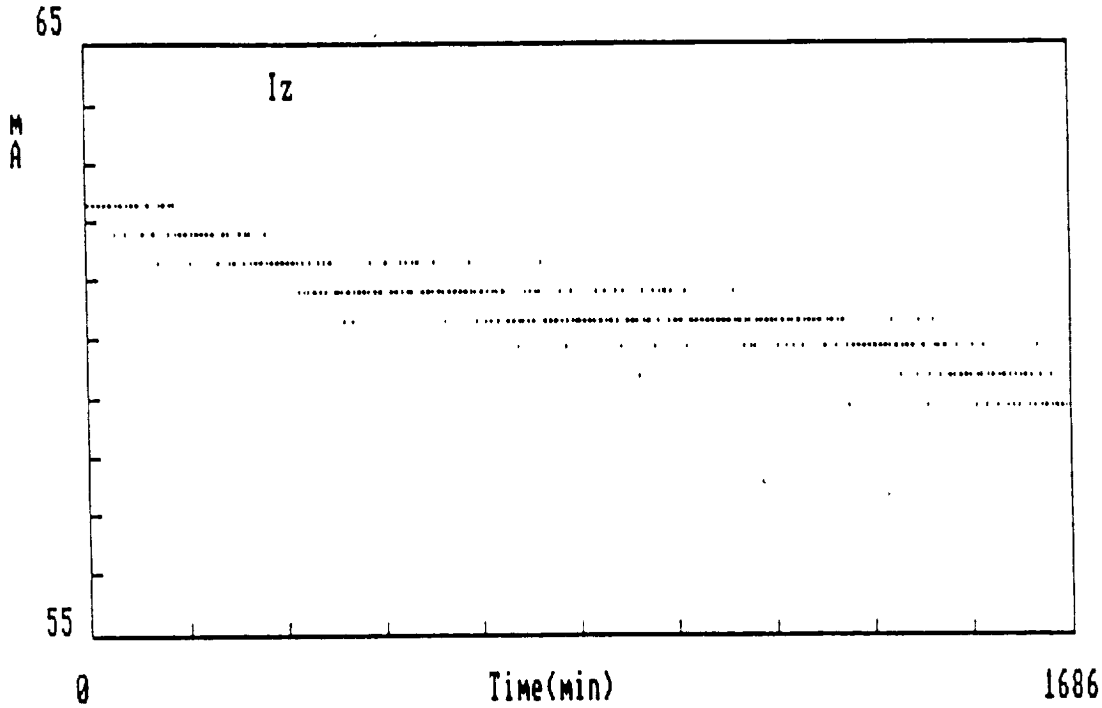


Fig. 5.1-1 Performance of Cell #2 (current vs. time) ,

CO_2,T=1200K,Vz=1.5V,F1=50sccm,Pc=115mb,P2=68mb

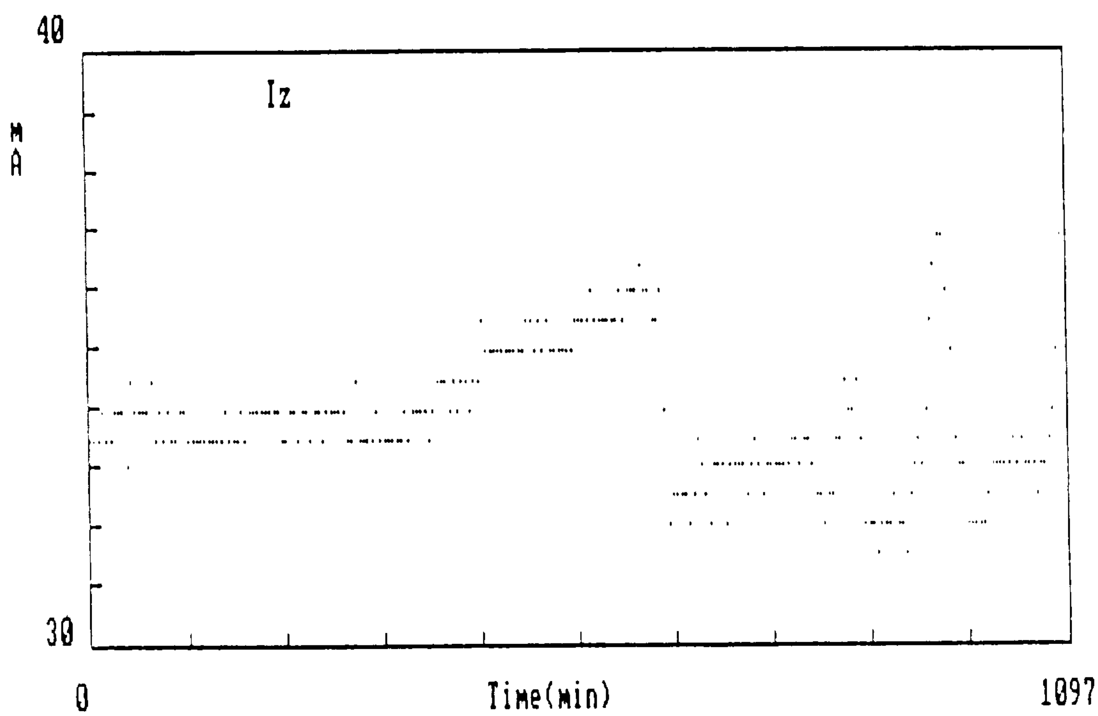


Fig. 5.1-2 Performance of Cell #2 (current vs. time) ,

CO_2,T=1200K,Vz=1.1V,F1=100sccm,Pc=P2=1atm

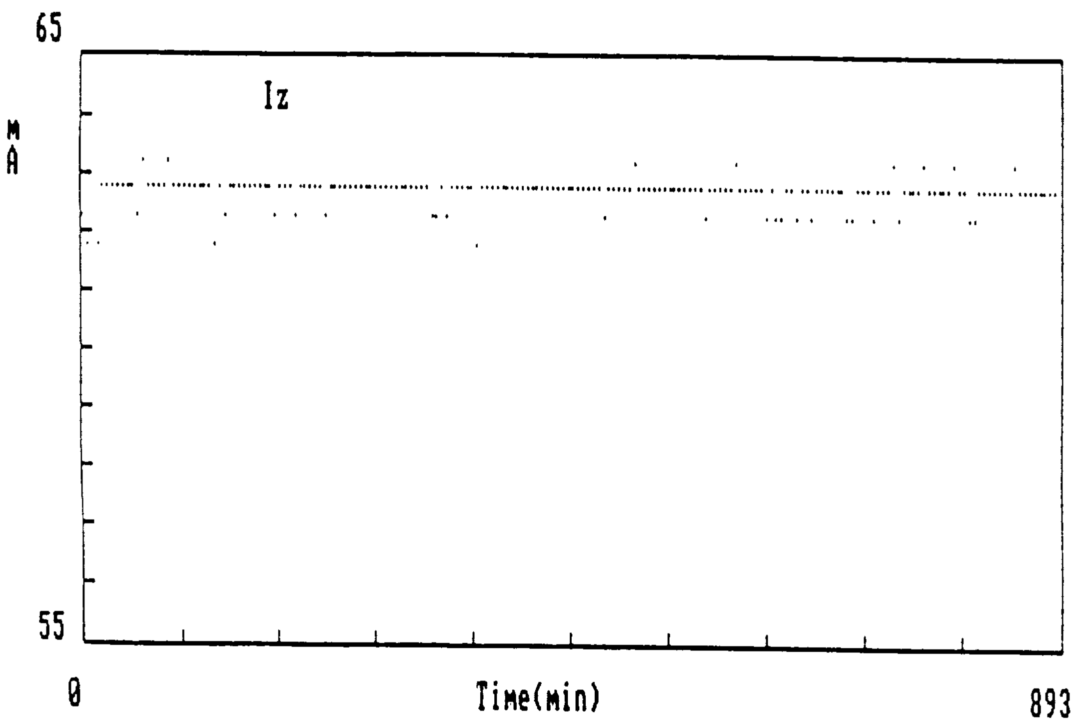


Fig. 5.1-3 Performance of Cell #2 (current vs. time),

Mars Gas, T=1200K, Vz=1.0V, F1=100sccm, Pc=115mb, P2=78mb

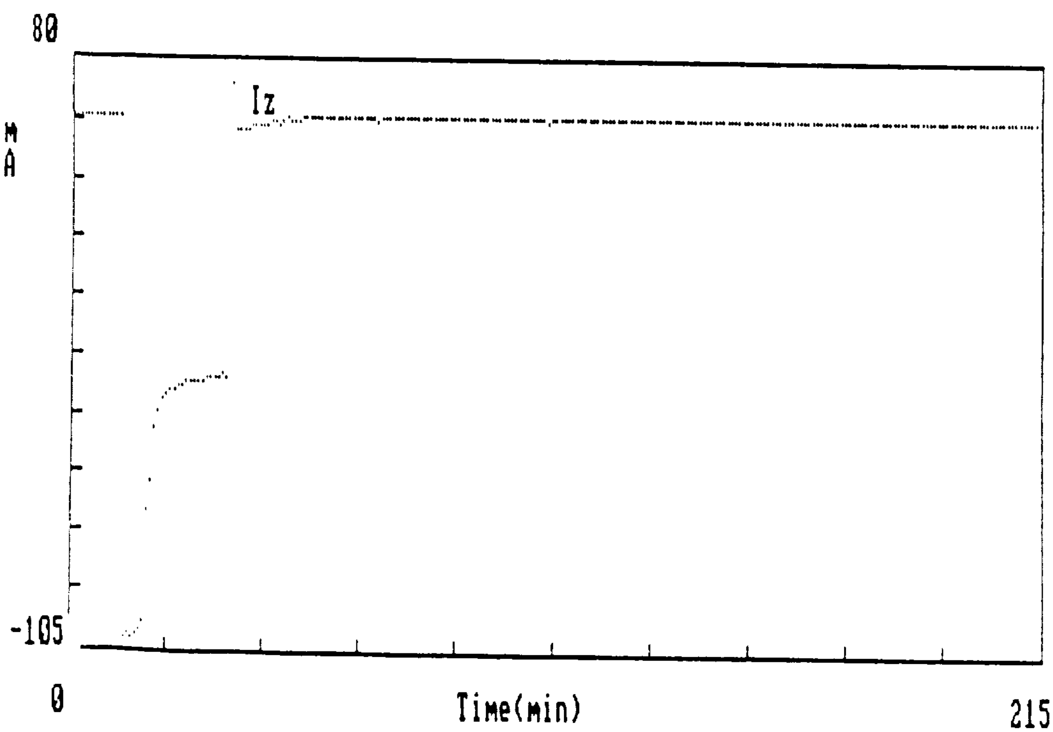


Fig. 5.1-4 Performance of Cell #2 (current vs. time),

Experiment of cell polarity reverse

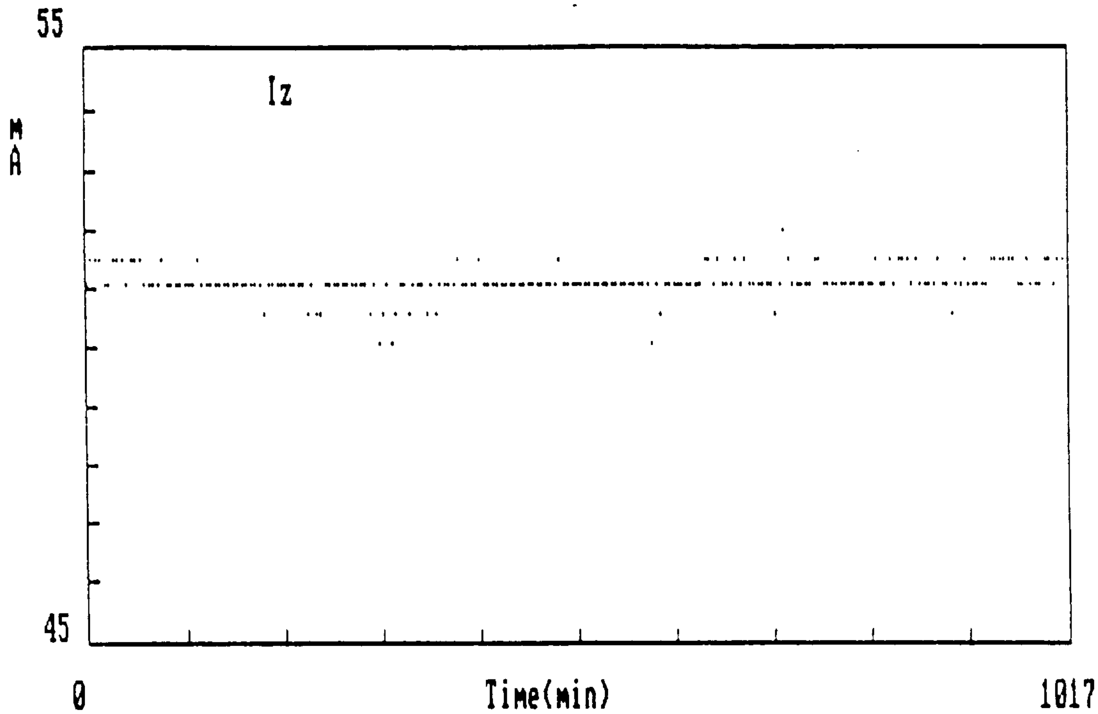


Fig. 5.1-5 Performance of Cell #2 (current vs. time),

Mars Gas, T=1200K, Vz=1.0V, F1=50sccm, Pc=P2=150mb

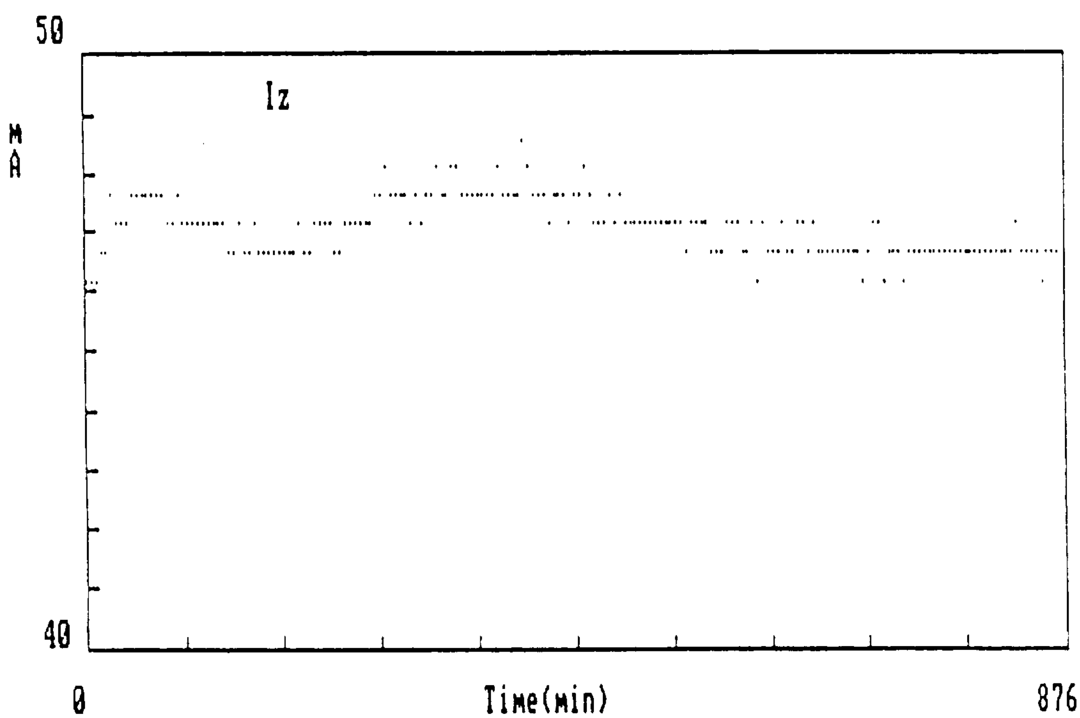


Fig. 5.1-6 Performance of Cell #1 (current vs. time),

Mars Gas, T=1200K, Vz=1.5V, F1=50sccm, Pc=P2=150mb

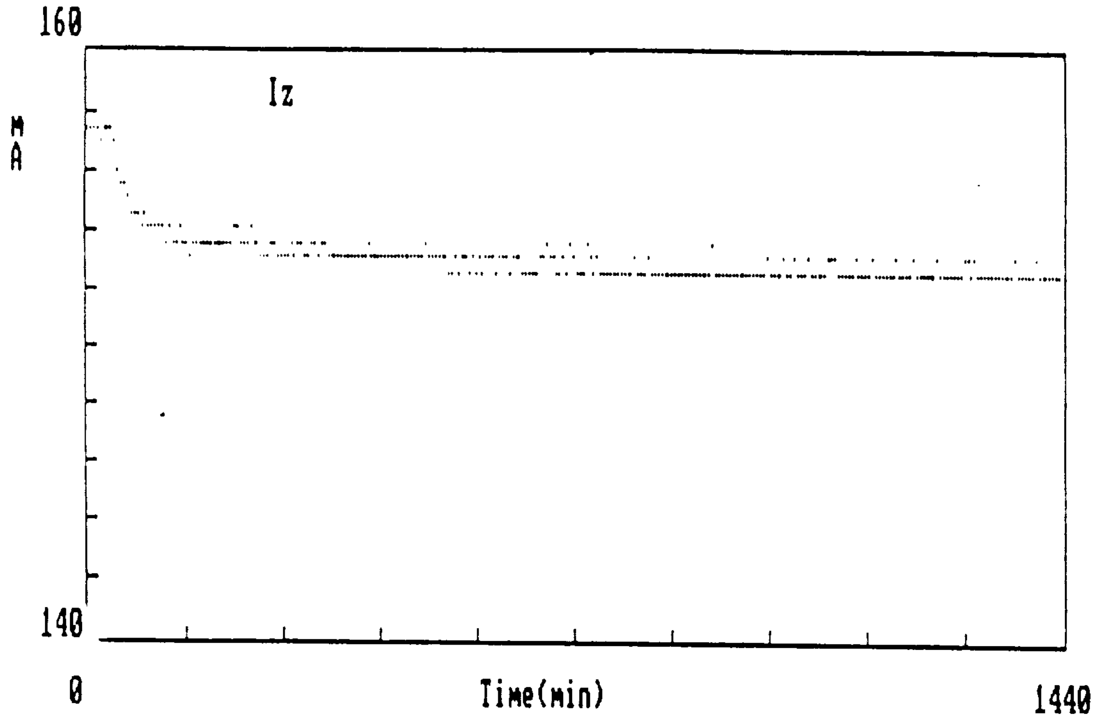


Fig. 5.1-7 Performance of Cell #2 (current vs. time),

CO_2, T=1200K, Vz=1.0V, F1=50sccm, Pc=127mb, Pz=47mb

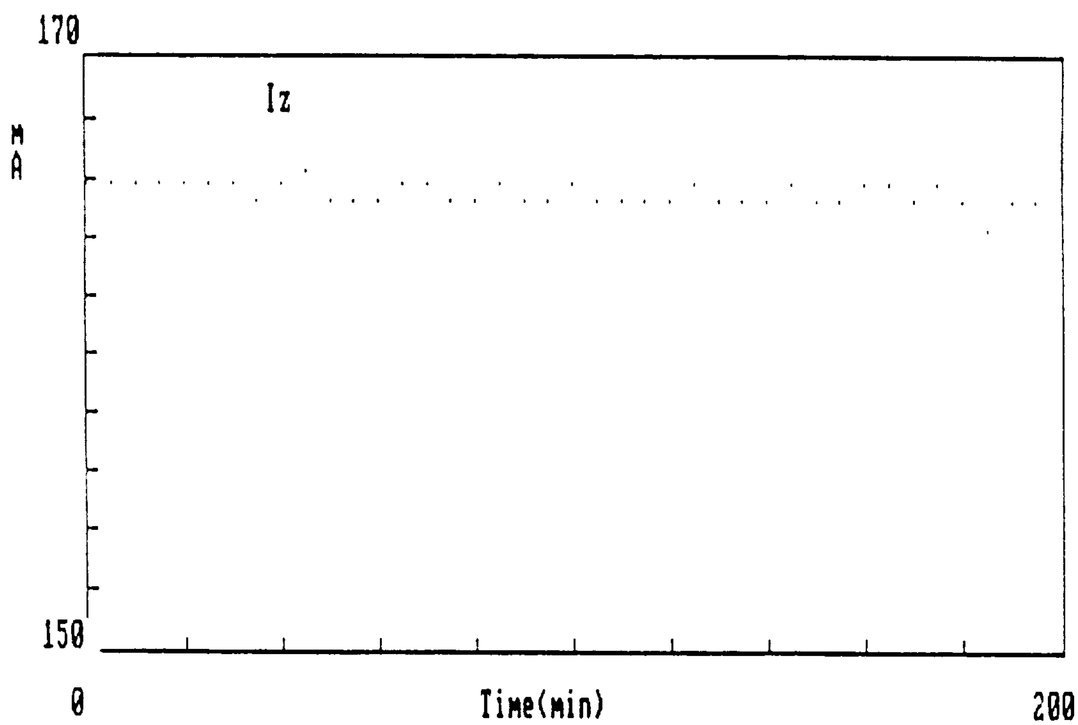


Fig. 5.1-8 Performance of Cell #2 (current vs. time),

CO_2, same parametric values as for Fig. 5.1-7

(with preheating pipe-coil)

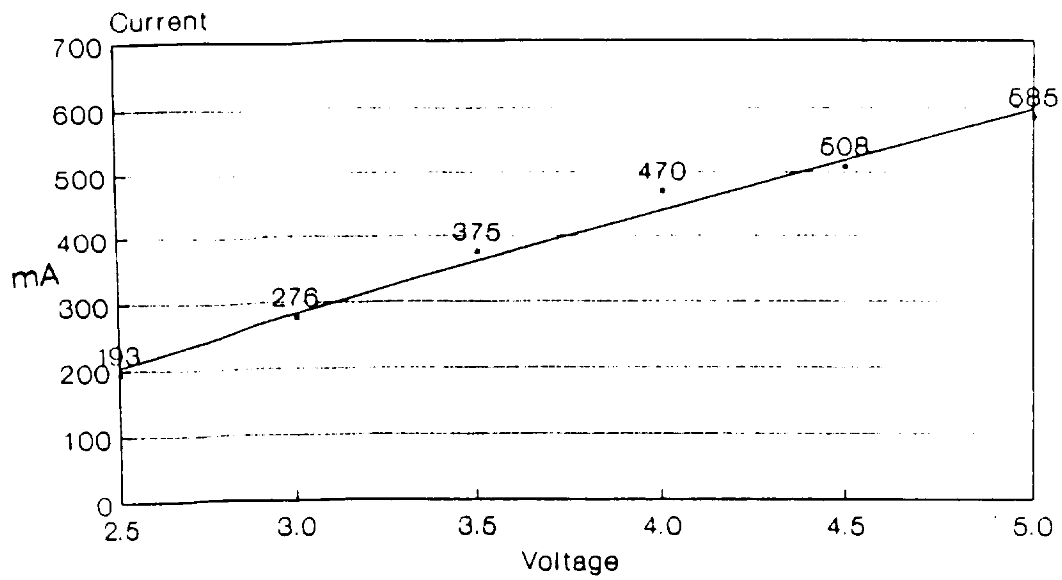


Fig. 5.1-9 Current vs Voltage (CO_2 Gas)

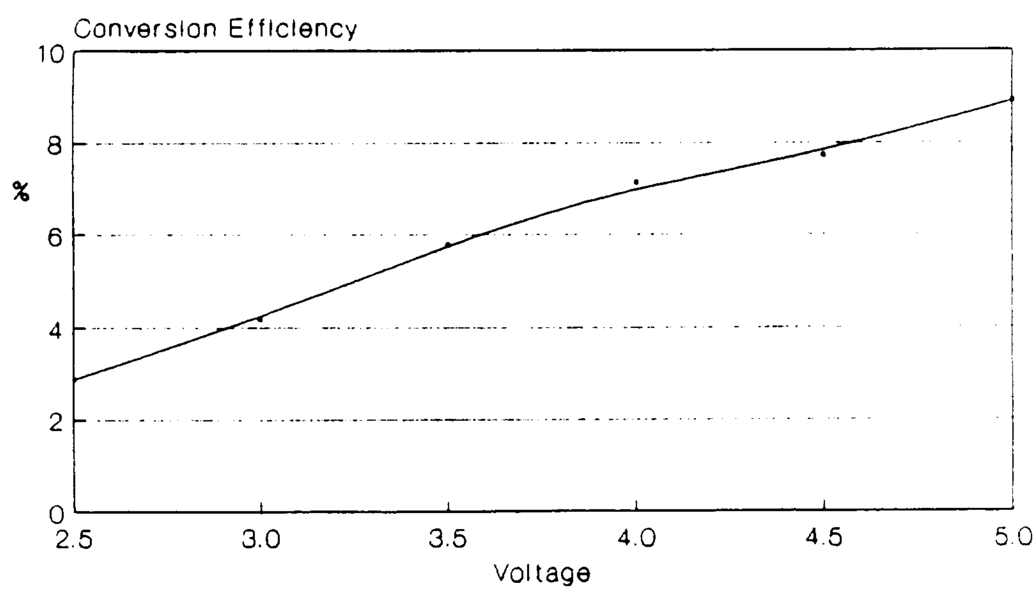


Fig. 5.1-10 Conversion Efficiency of CO_2

ORIGINAL PAGE IS
OF POOR QUALITY

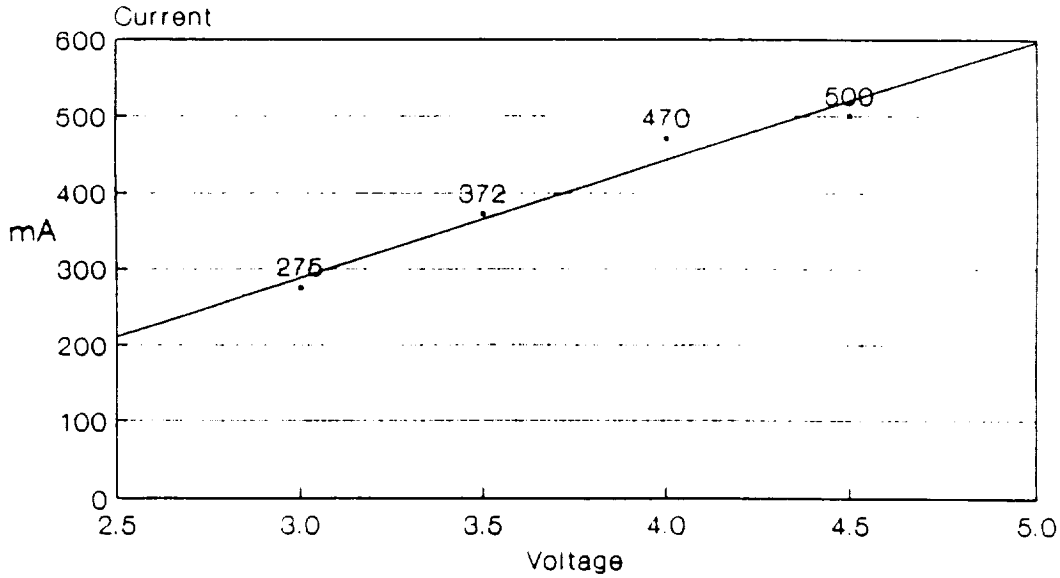


Fig. 5.1-11 Current vs Voltage (Mars Gas)

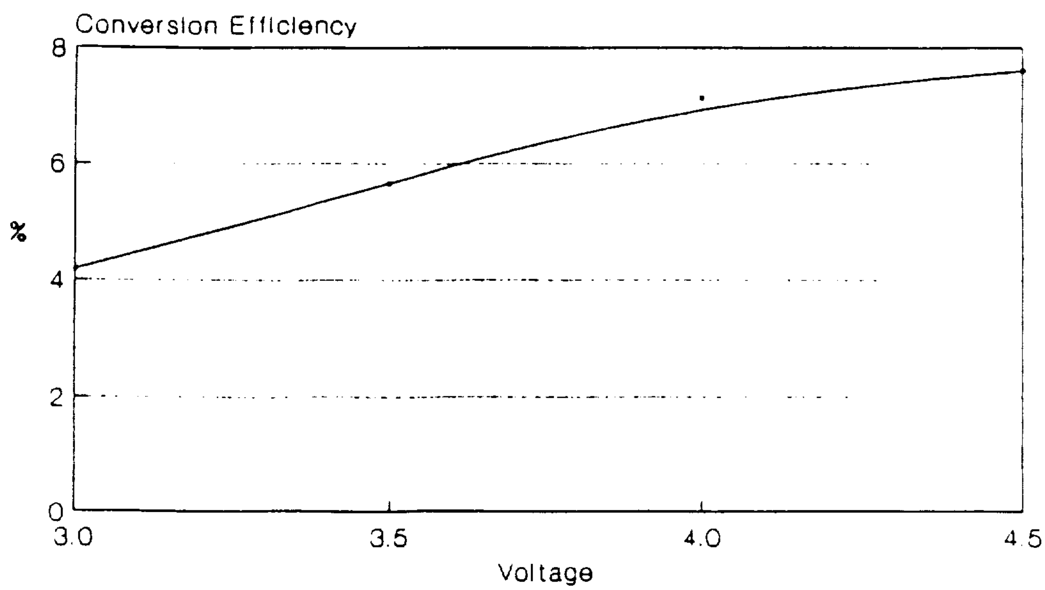
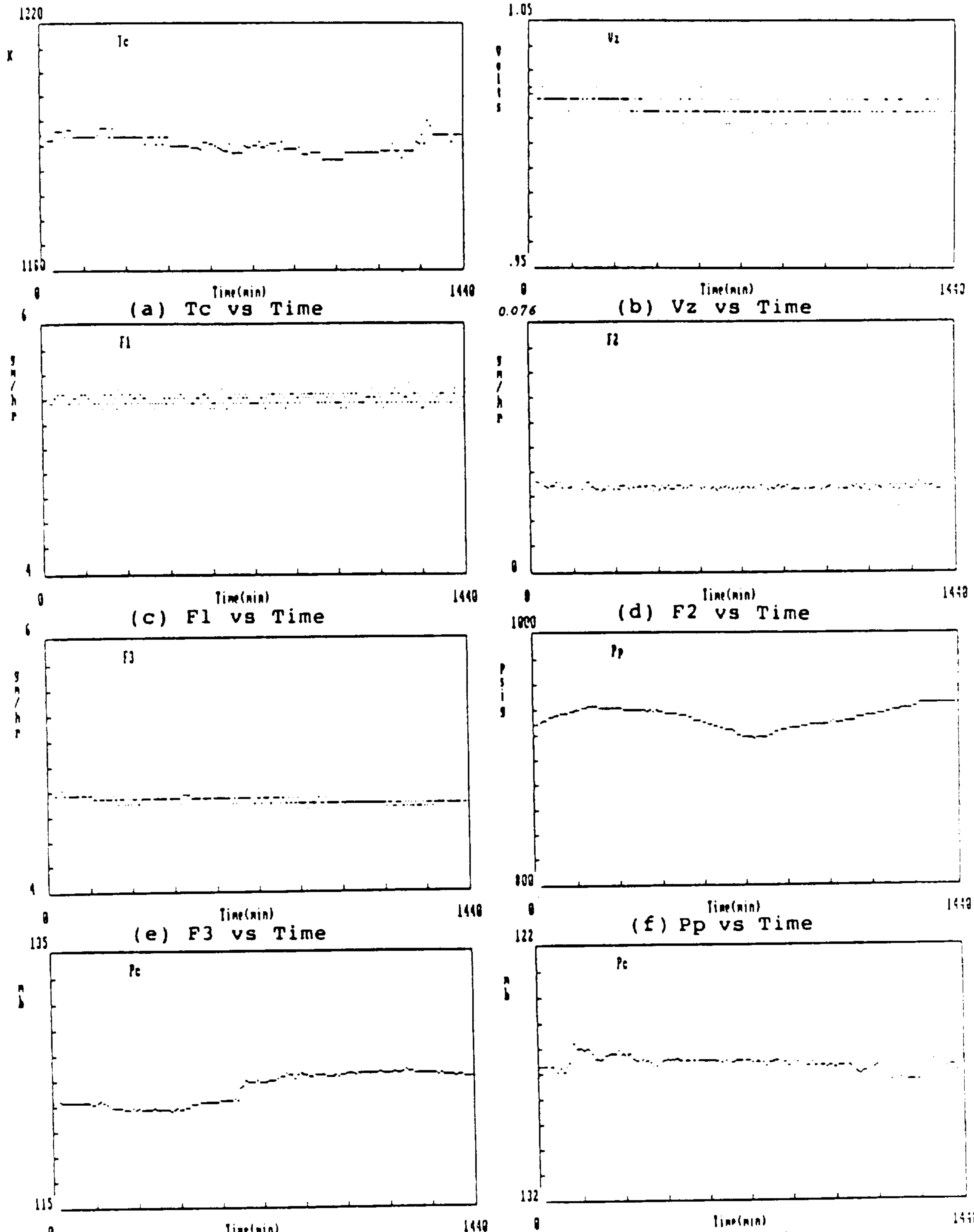


Fig. 5.1-12 Conversion Efficiency of Mars Gas

ORIGINAL PAGE IS OF POOR QUALITY



(a) Tc vs Time

(b) Vz vs Time

(c) F1 vs Time

(d) F2 vs Time

(e) F3 vs Time

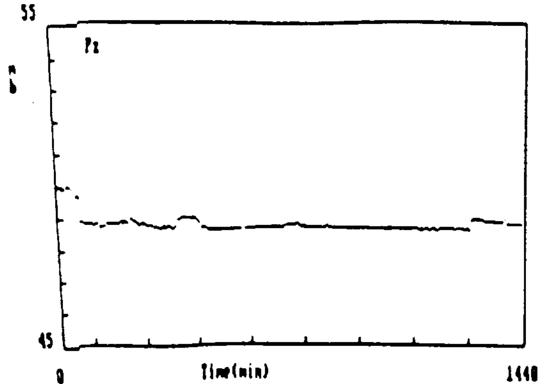
(f) Pp vs Time

(g) Pc vs Time

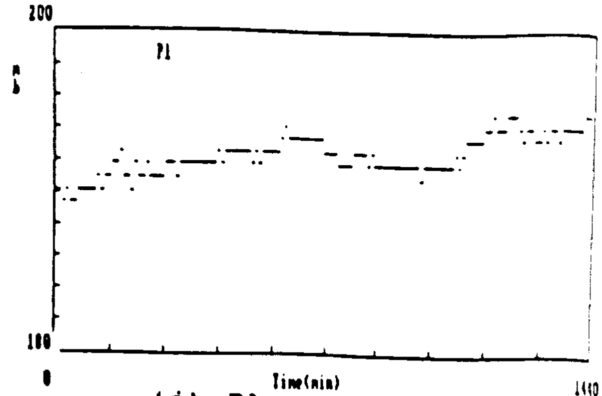
(h) Pc vs Time

Fig. 5.2(a to h) System Performance over 24 hours
(Refer to Fig. 2.1 for symbolic representation)

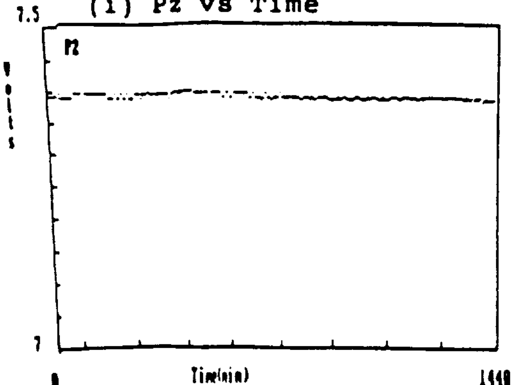
ORIGINAL PAGE IS
OF POOR QUALITY



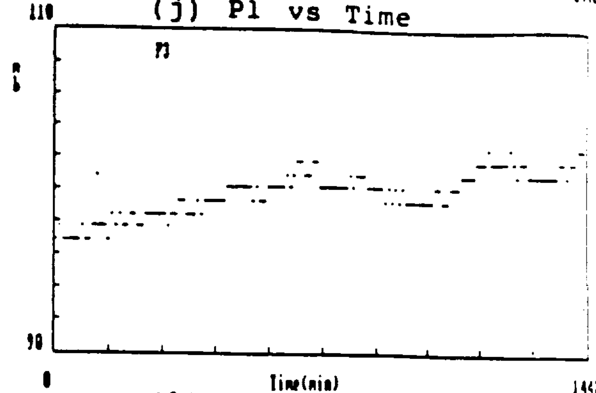
(i) Pz vs Time



(j) P1 vs Time



(k) P2 vs Time



(l) P3 vs Time

Fig. 5.2(i to l) System Performance over 24 hours

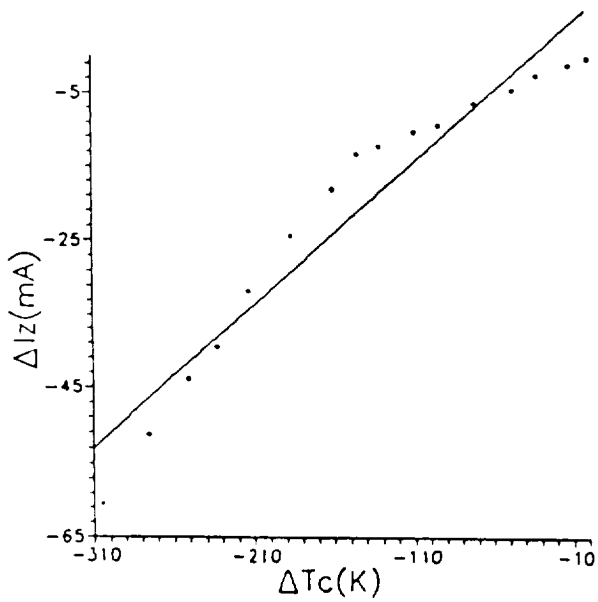


Fig. 5.3-1

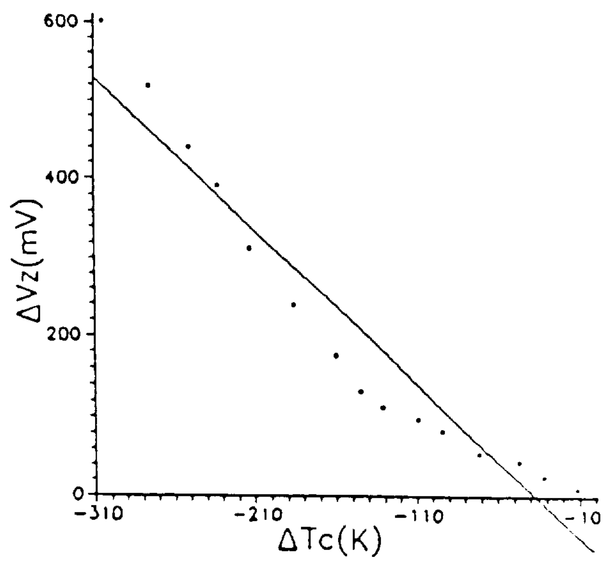


Fig. 5.3-2

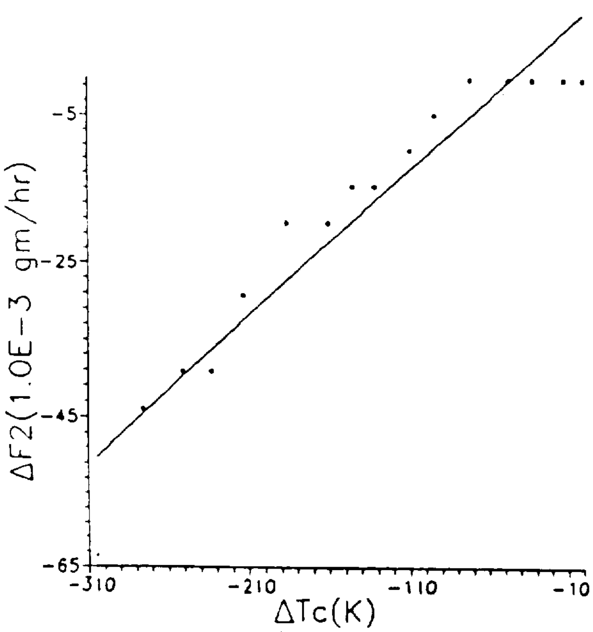


Fig. 5.3-3

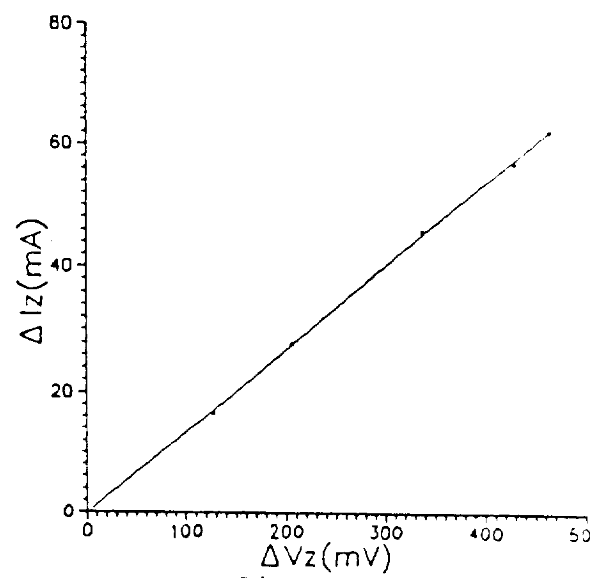


Fig. 5.3-4

Fig. 5.3-1 to 5.3-24 Testing Results of Failure Modes

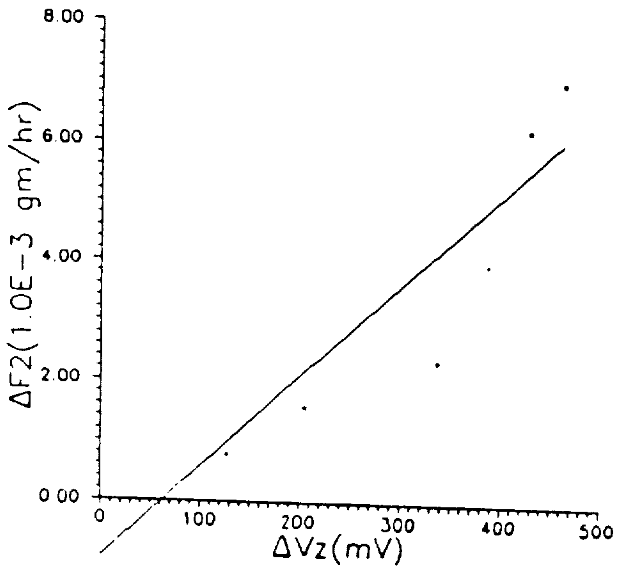


Fig. 5.3-5

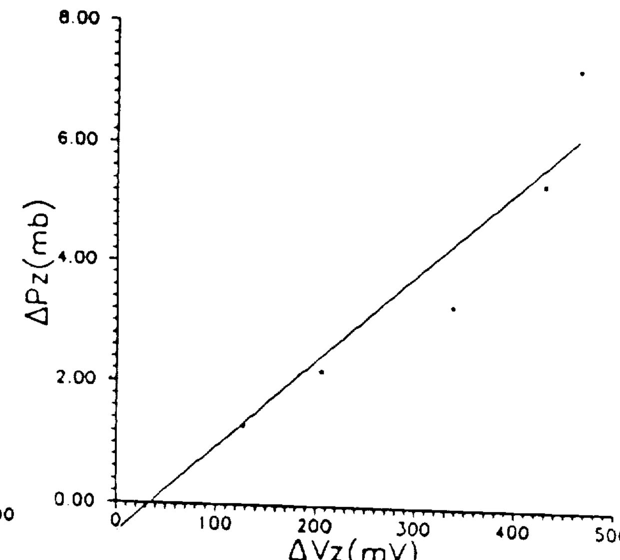


Fig. 5.3-6

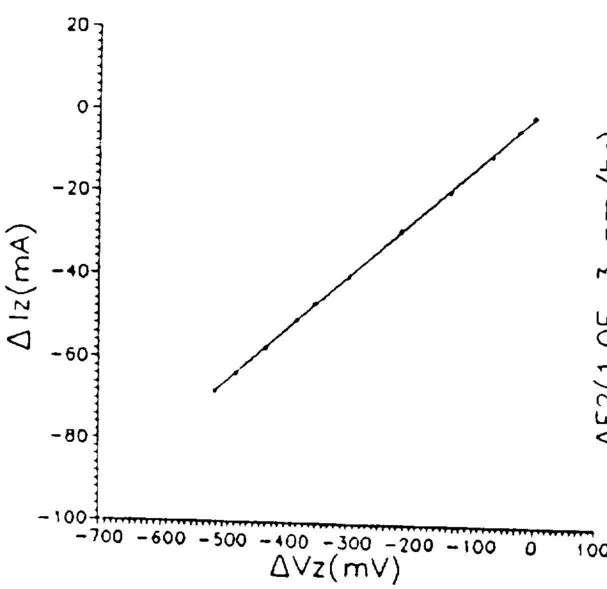


Fig. 5.3-7

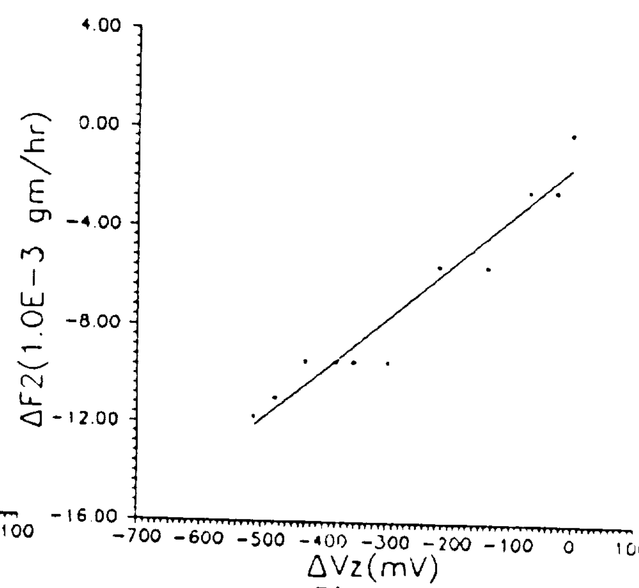


Fig. 5.3-8

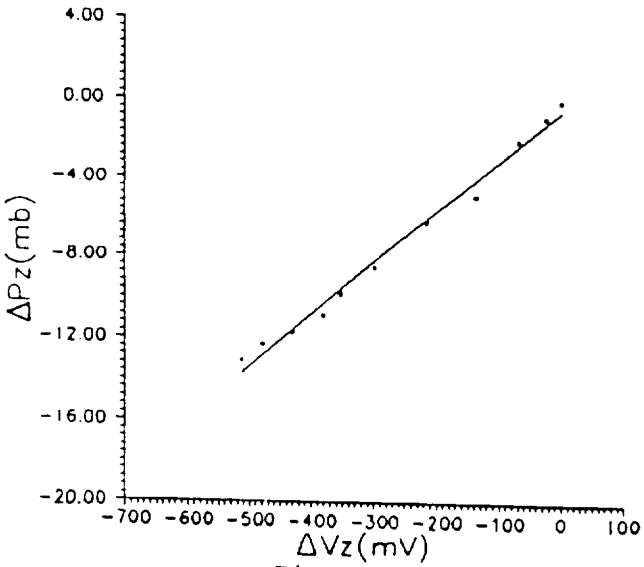


Fig. 5.3-9

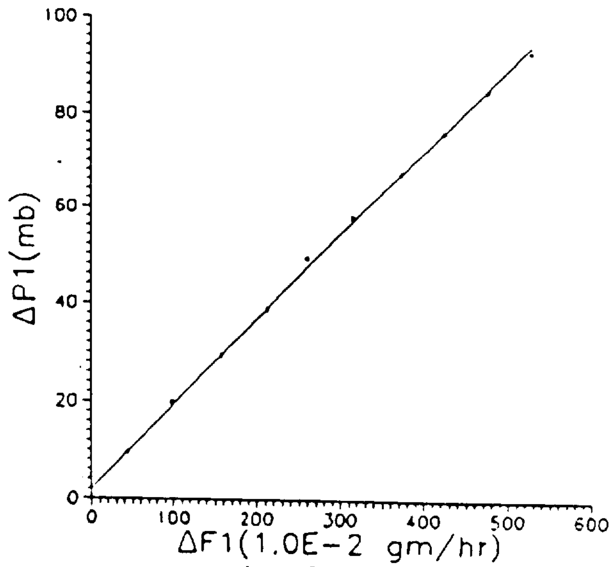


Fig. 5.3-10

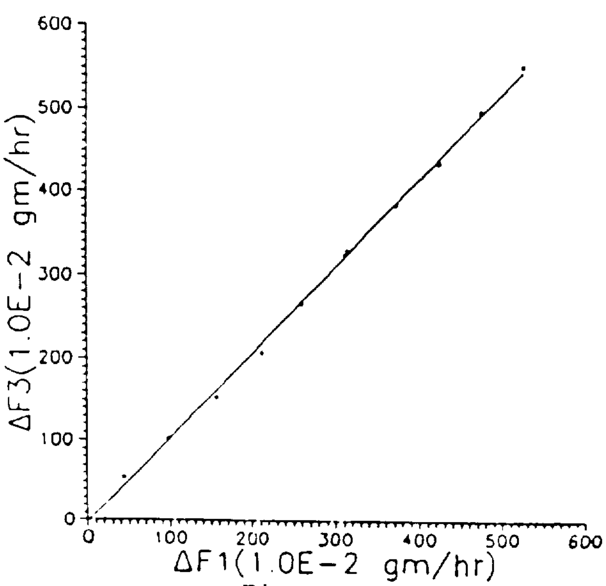


Fig. 5.3-11

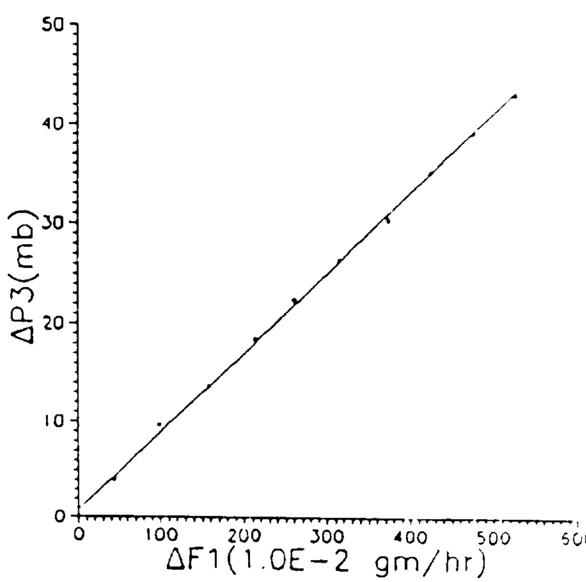


Fig. 5.3-12

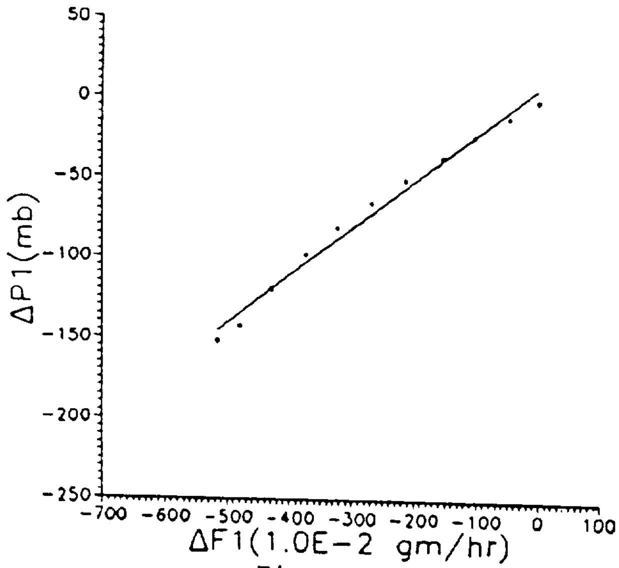


Fig. 5.3-13

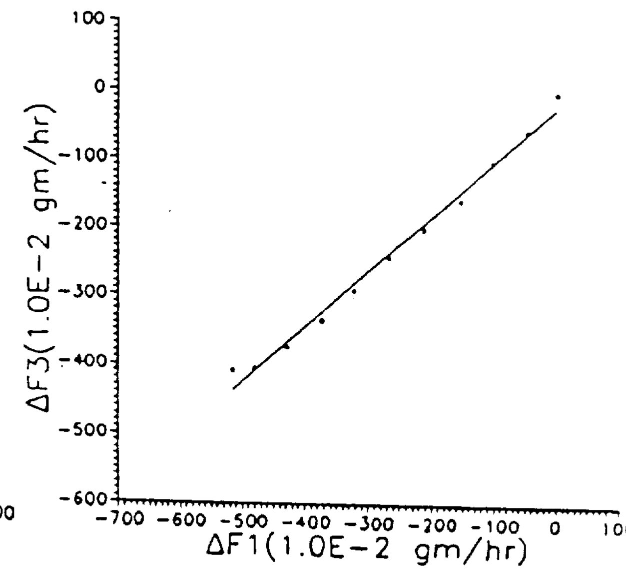


Fig. 5.3-14

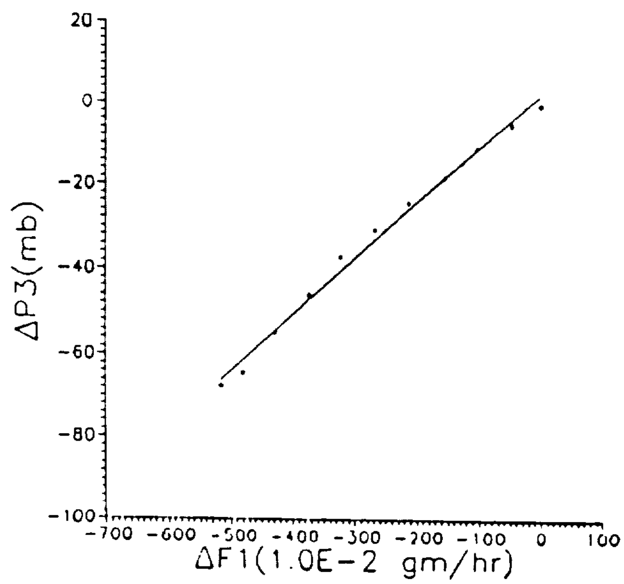


Fig. 5.3-15

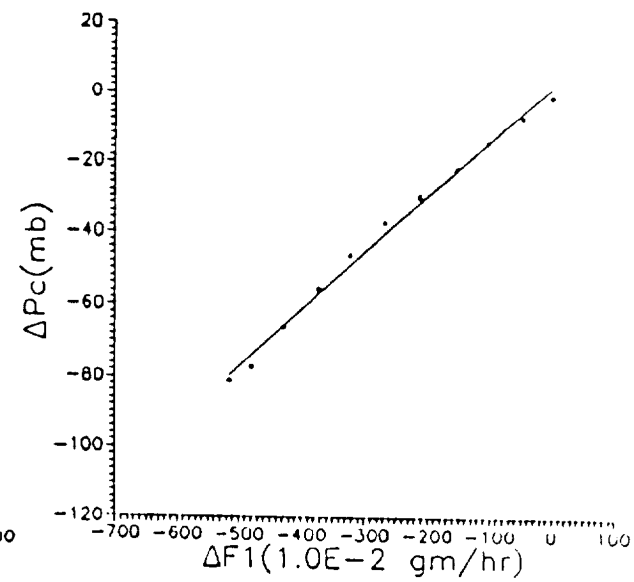


Fig. 5.3-16

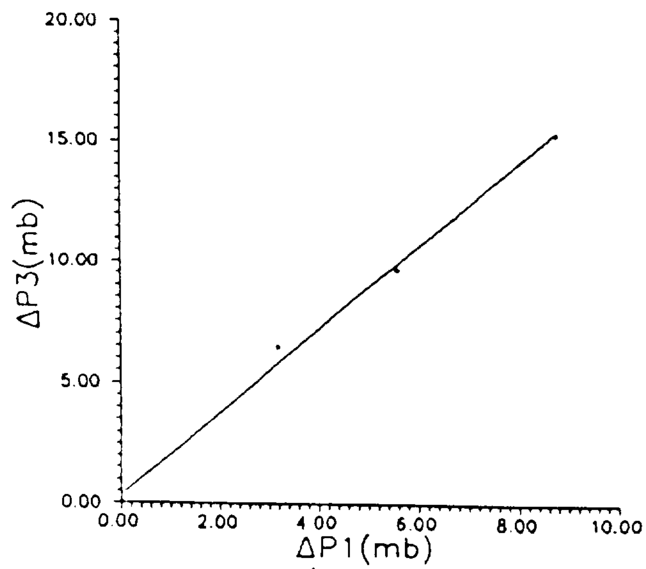


Fig. 5.3-17

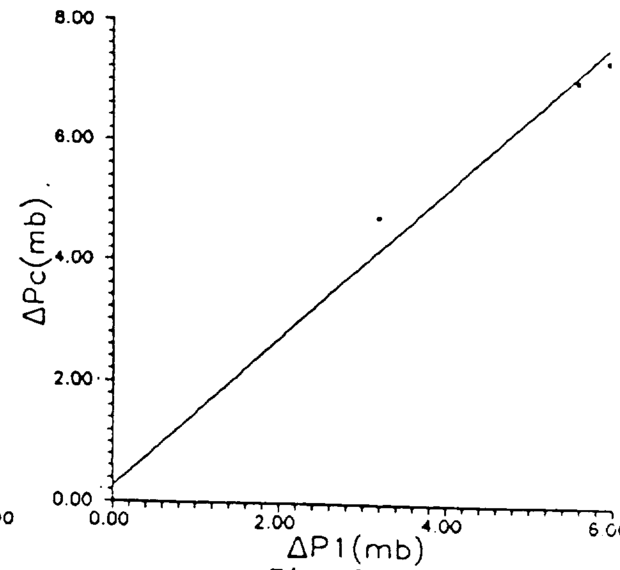


Fig. 5.3-18

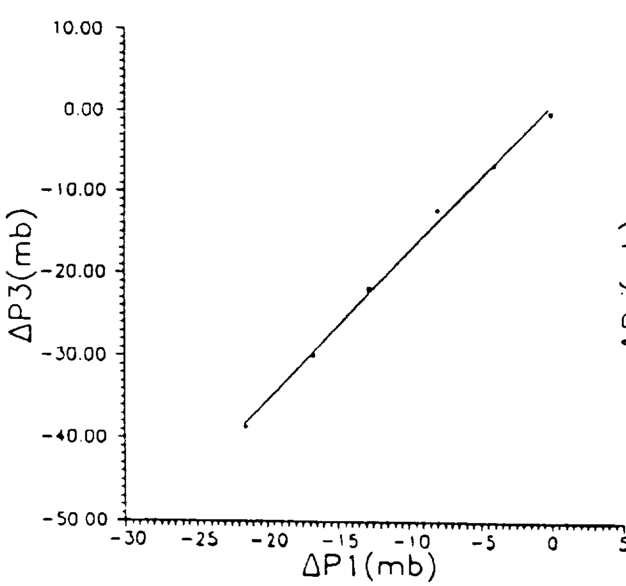


Fig. 5.3-19

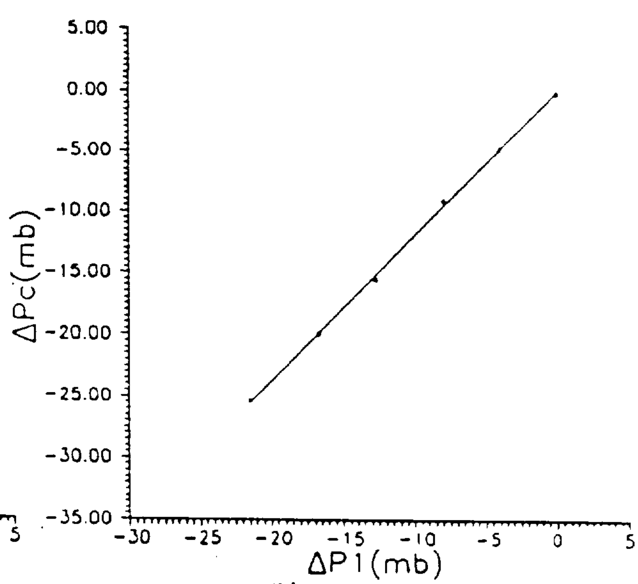


Fig. 5.3-20

C2

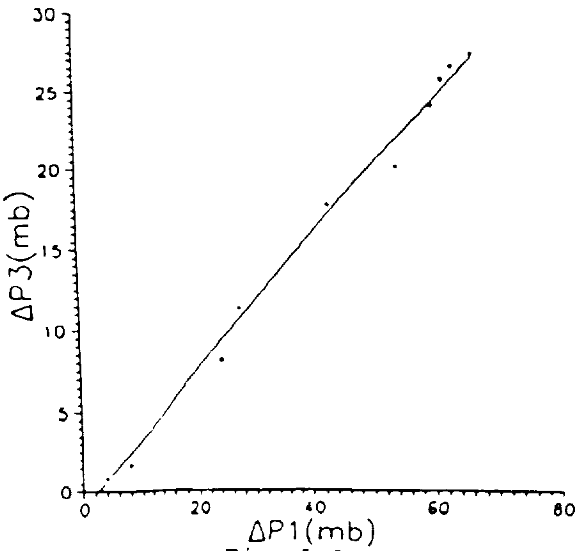


Fig. 5.3-21

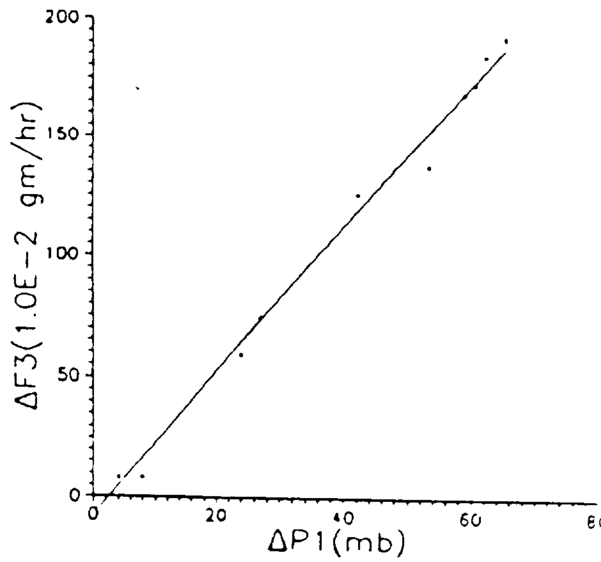


Fig. 5.3-22

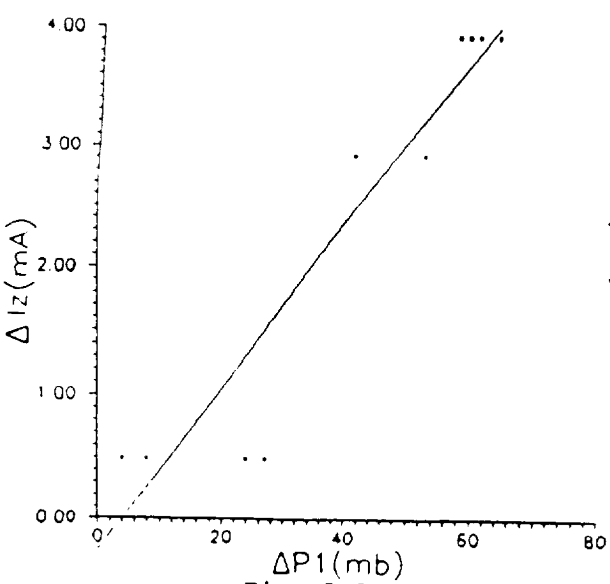


Fig. 5.3-23

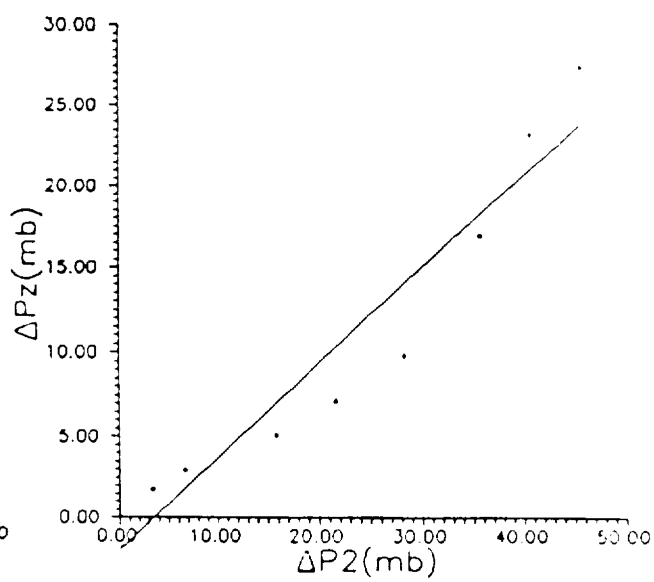


Fig. 5.3-24

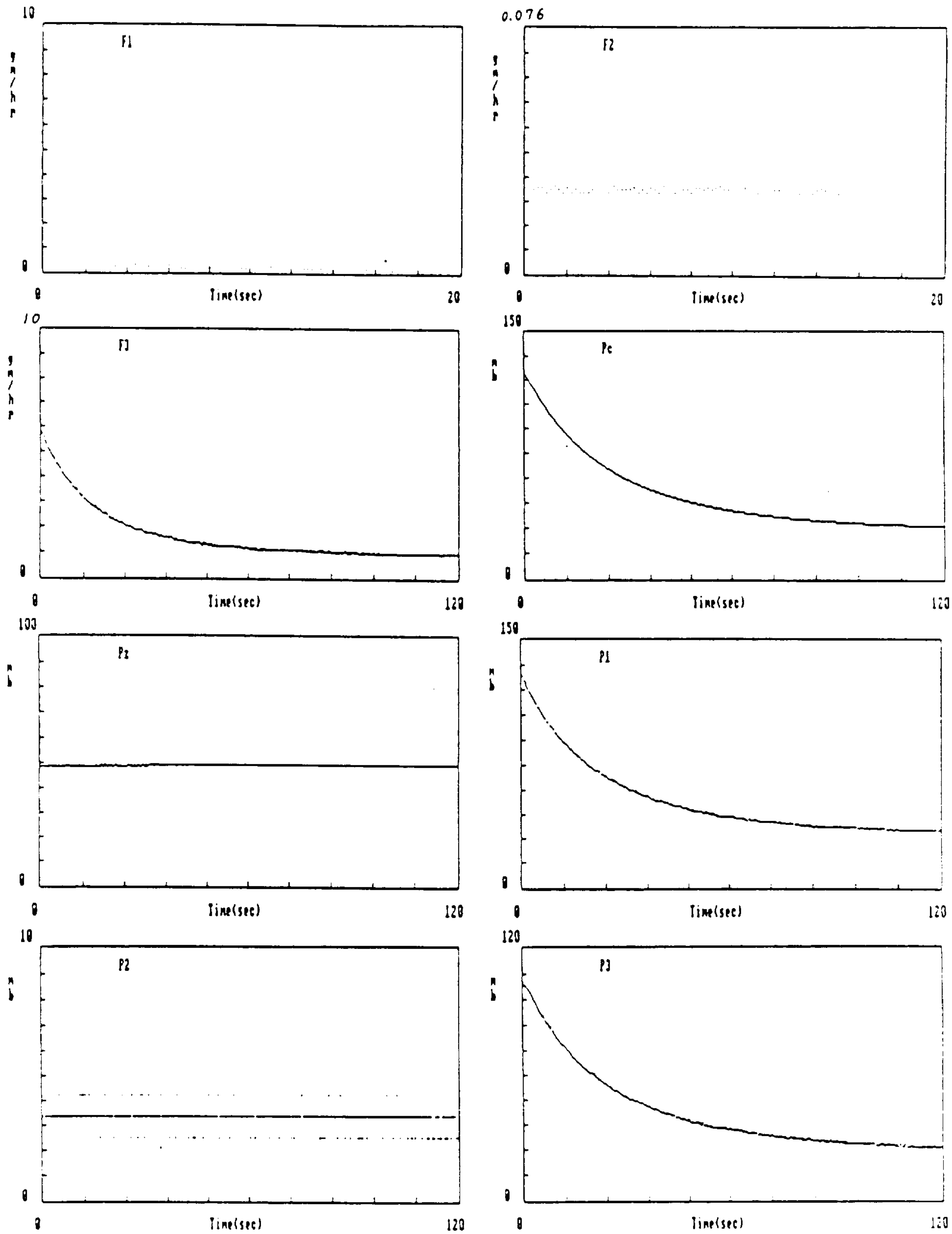


Fig. 5.4a System Response (F1 is changed from 50 to 0 sccm)

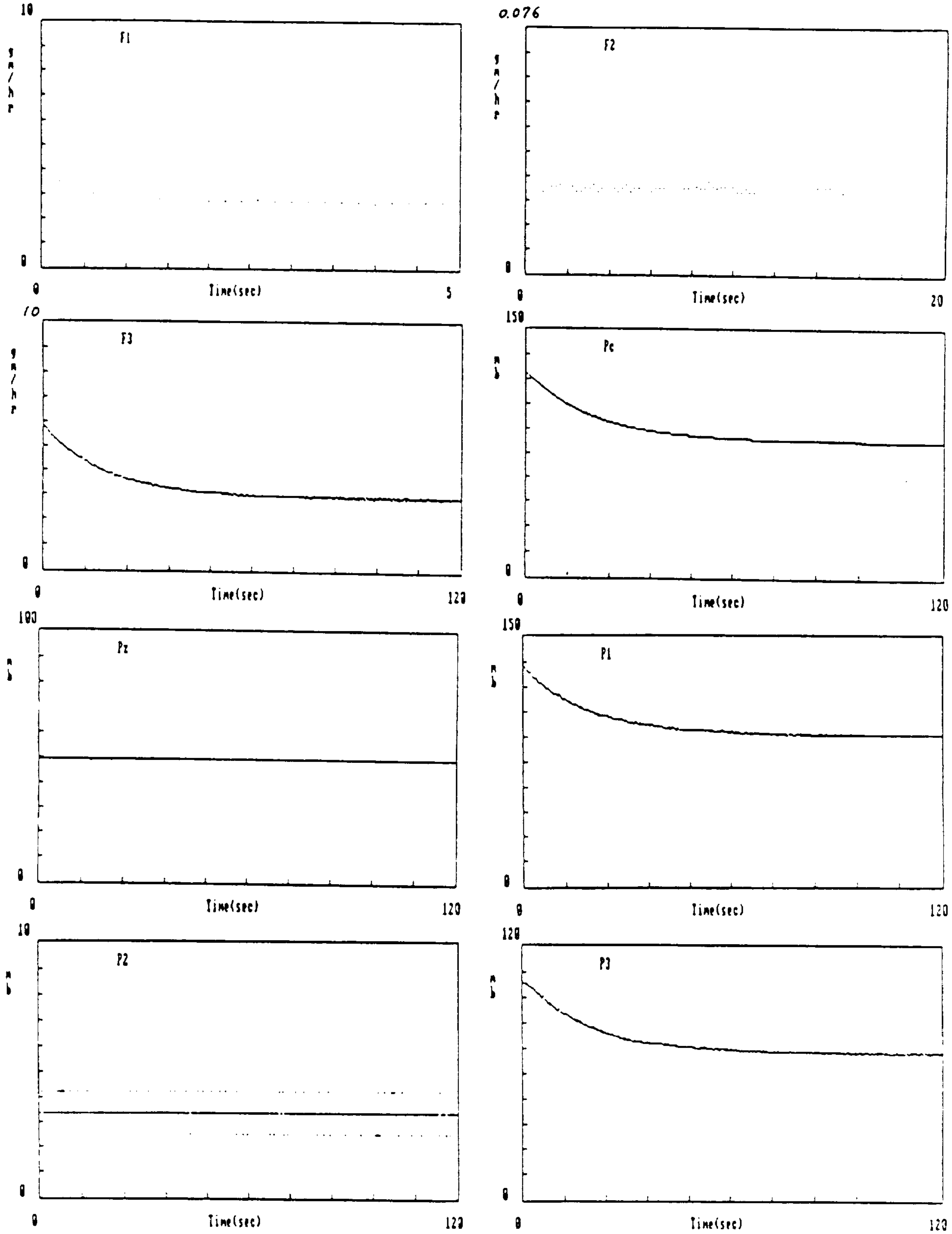


Fig. 5.4b System Response (F1 is changed from 50 to 25 sccm)

ORIGINAL PAGE IS
OF POOR QUALITY

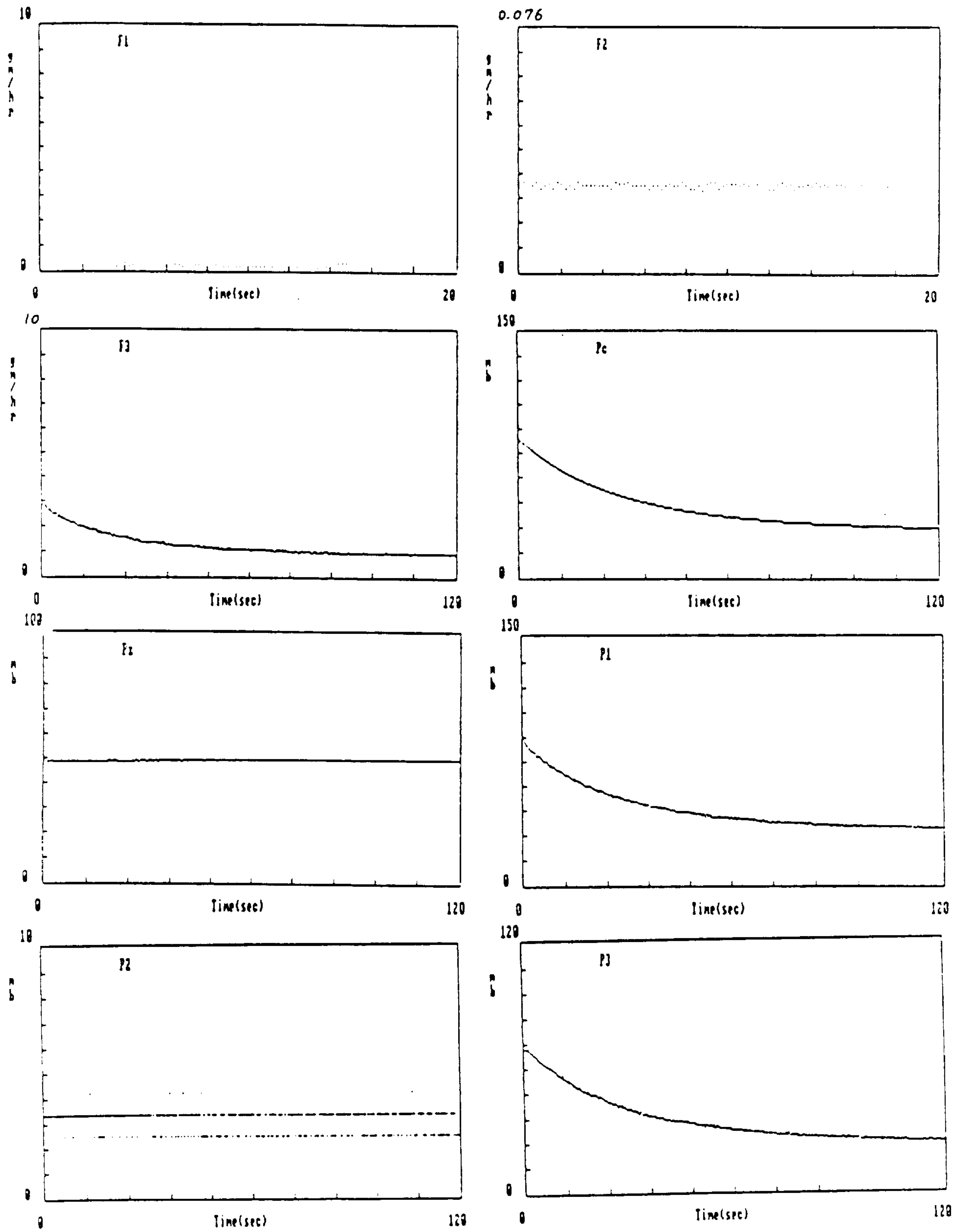


Fig. 5.4c System Response (F1 is changed from 25 to 0 sccm)

ORIGINAL PAGE IS
OF POOR QUALITY

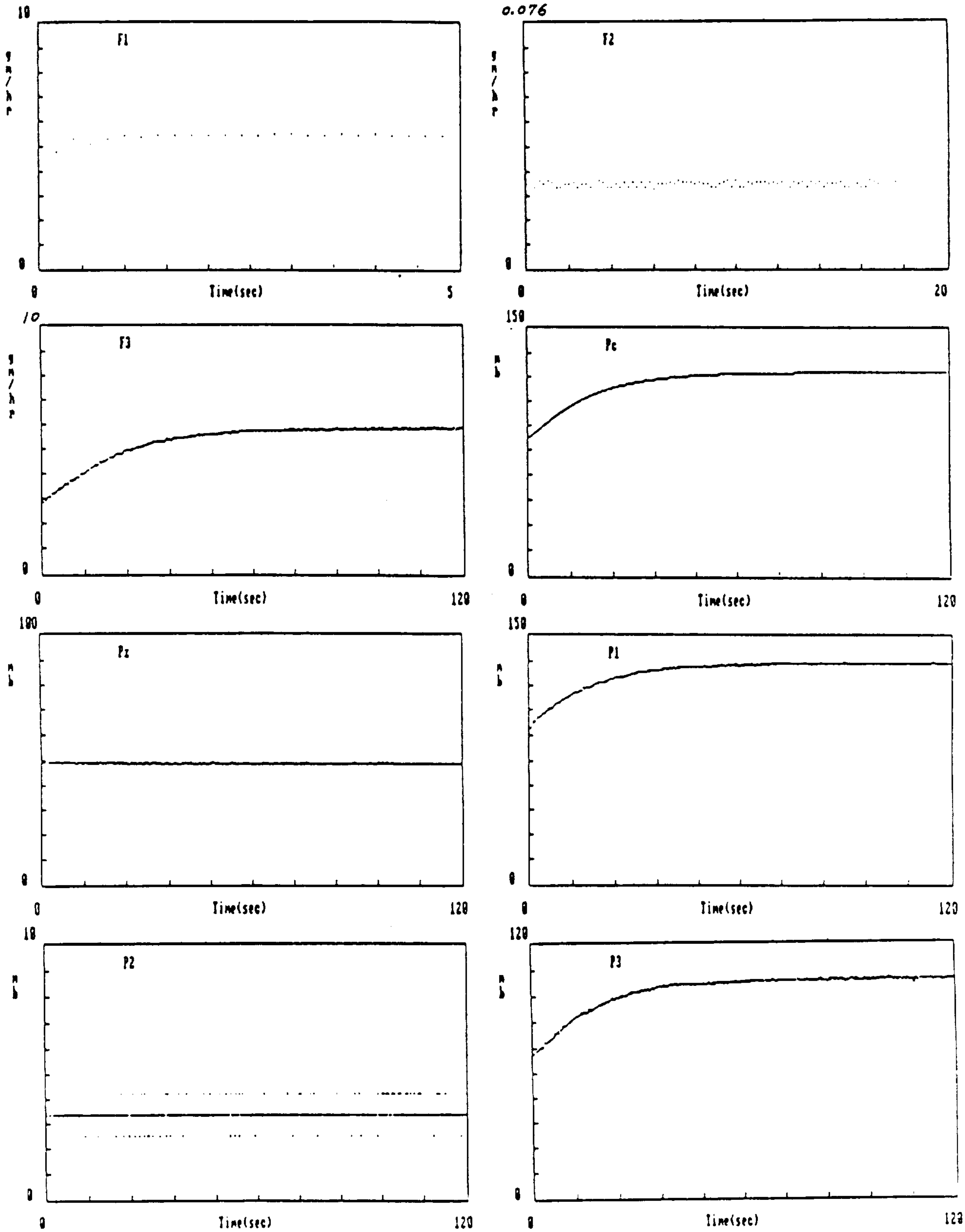


Fig. 5.4d System Response (F1 is changed from 25 to 50 sccm)

ORIGINAL PAGE IS
OF POOR QUALITY

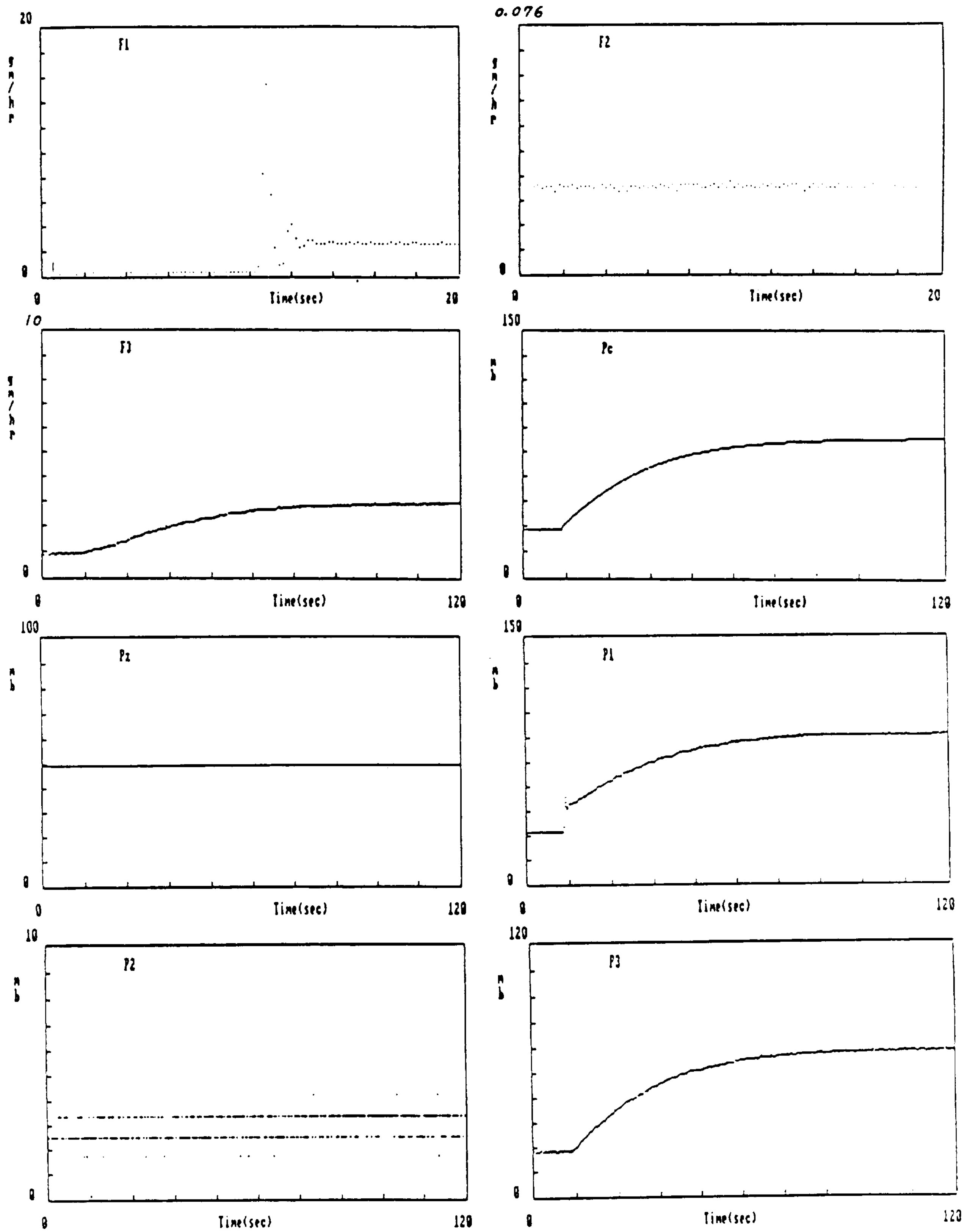


Fig. 5.4e System Response (F1 is changed from 0 to 25 sccm)

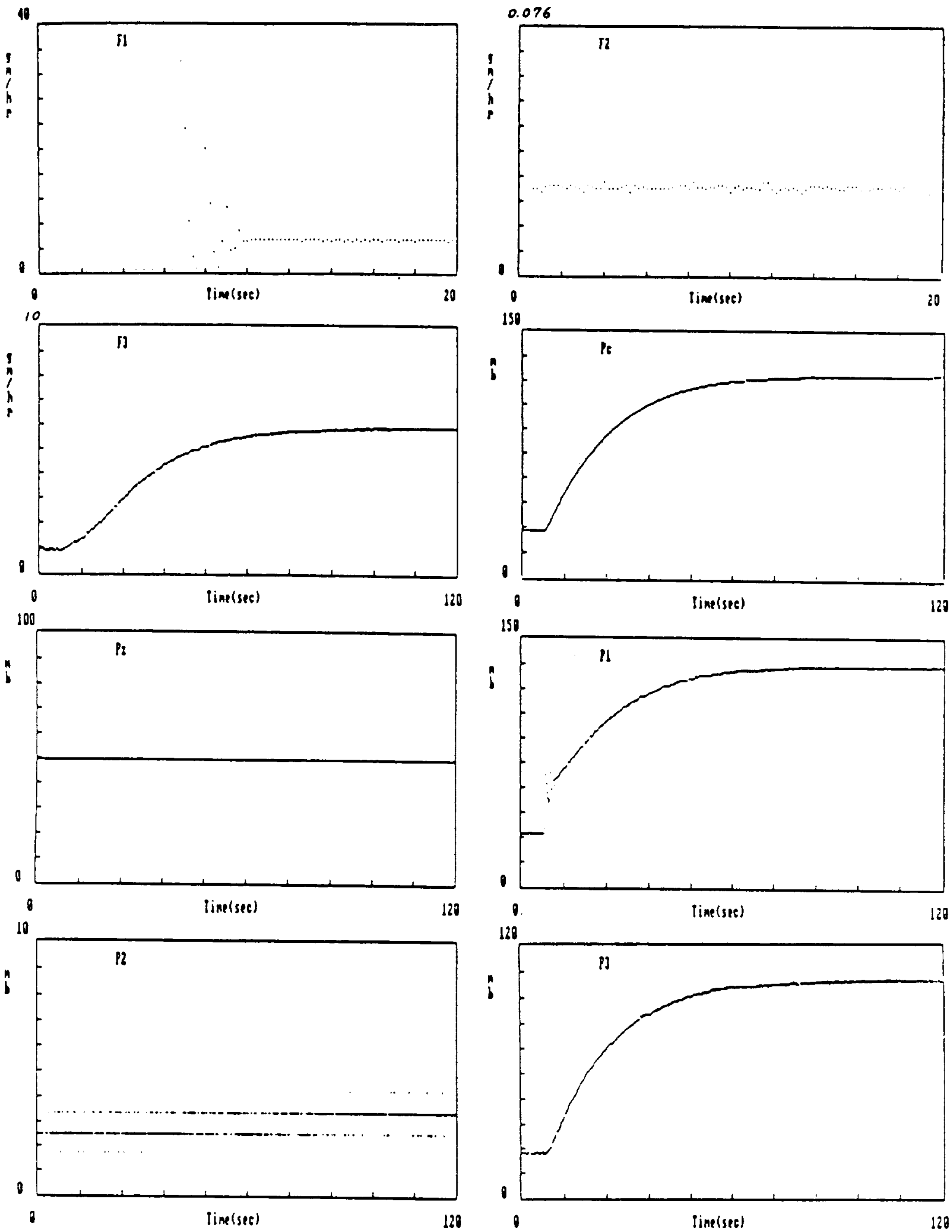


Fig. 5.4f System Response (F1 is changed from 0 to 50 sccm)

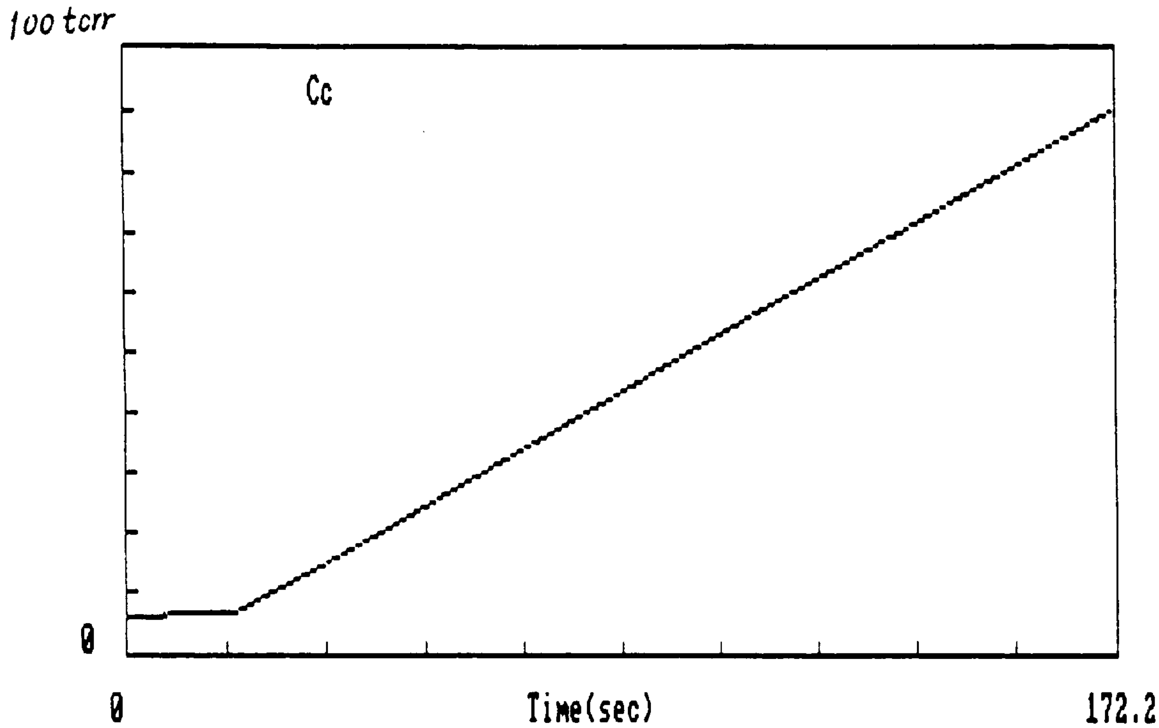


Fig. 5.5-1 Cell Capacitance Experiment

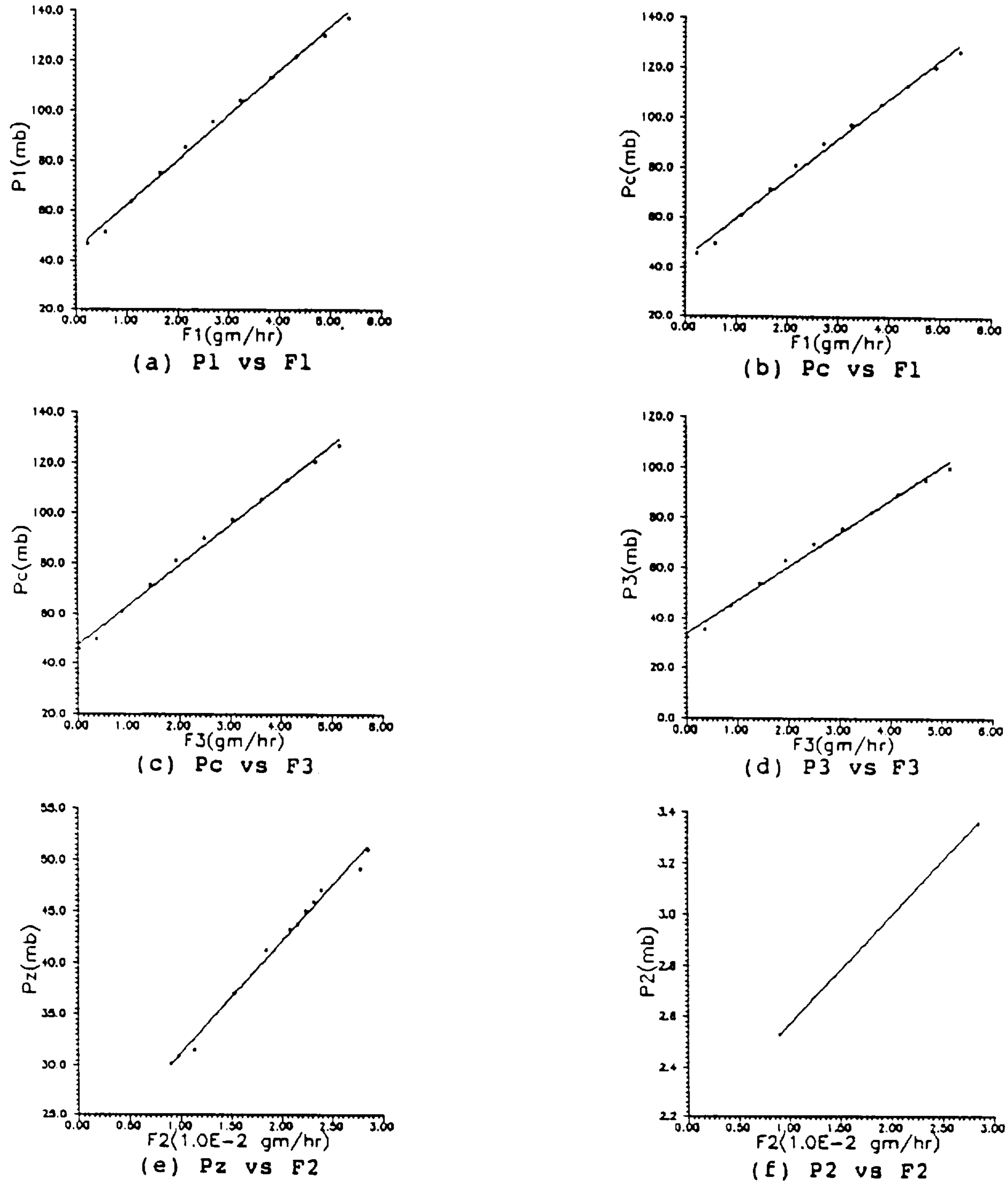


Fig. 5.5-2(a to f) Pipeline Resistance Experiments

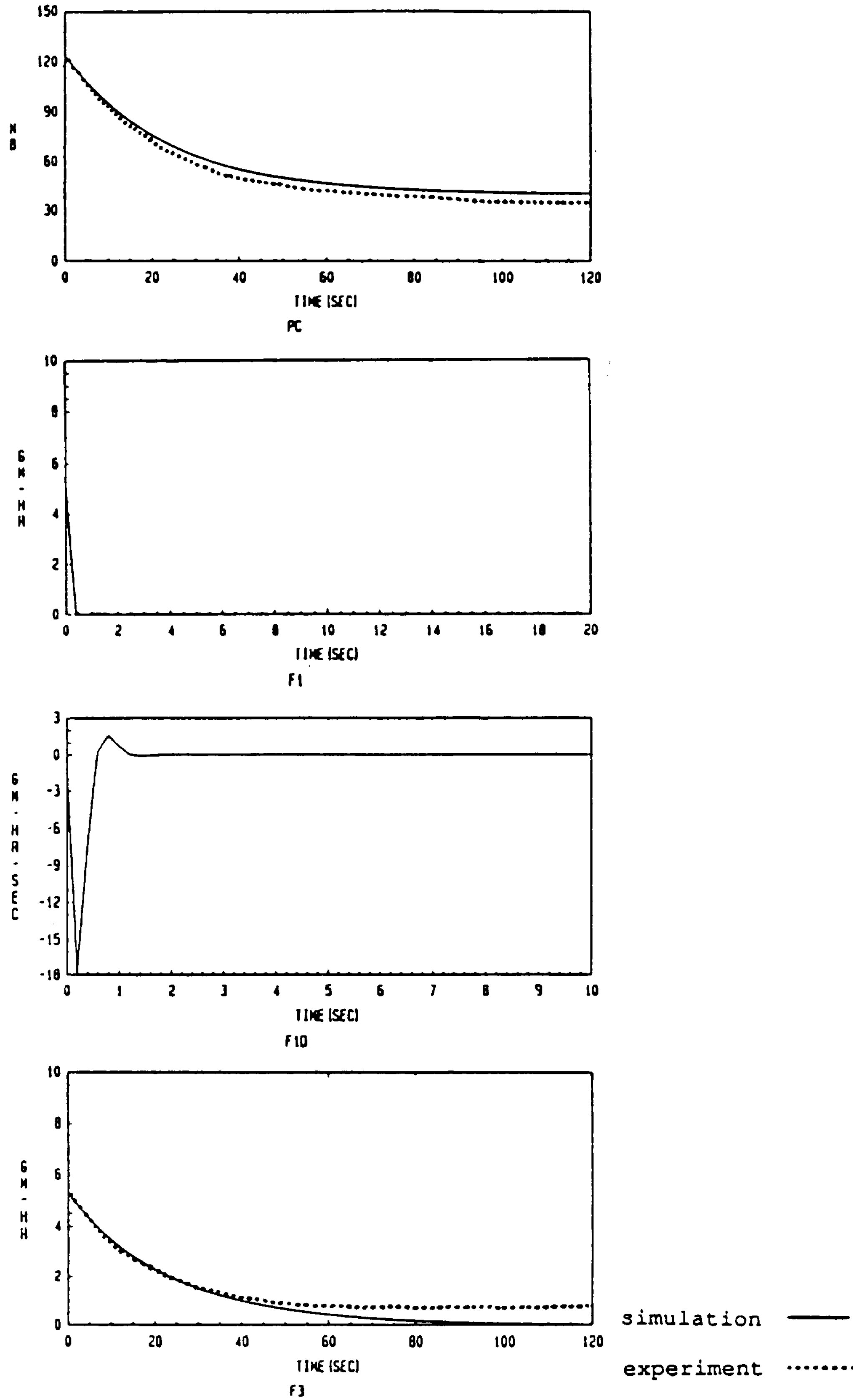


Fig. 5.5-3a Numerical Simulation(F1 is changed from 50 to 0 sccm)

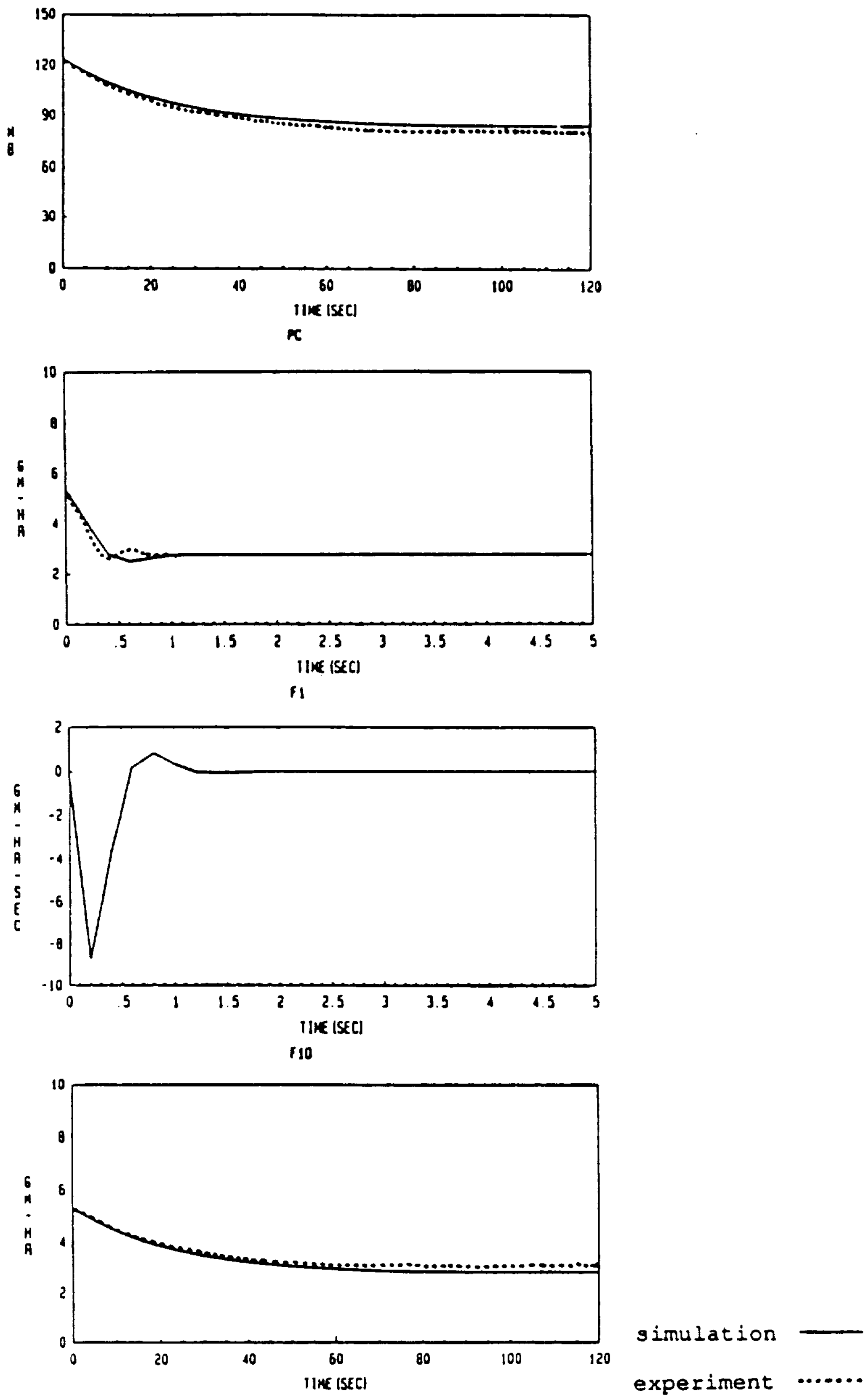


Fig. 5.5-3b Numerical Simulation(F1 is changed from 50 to 25 sccm) 102

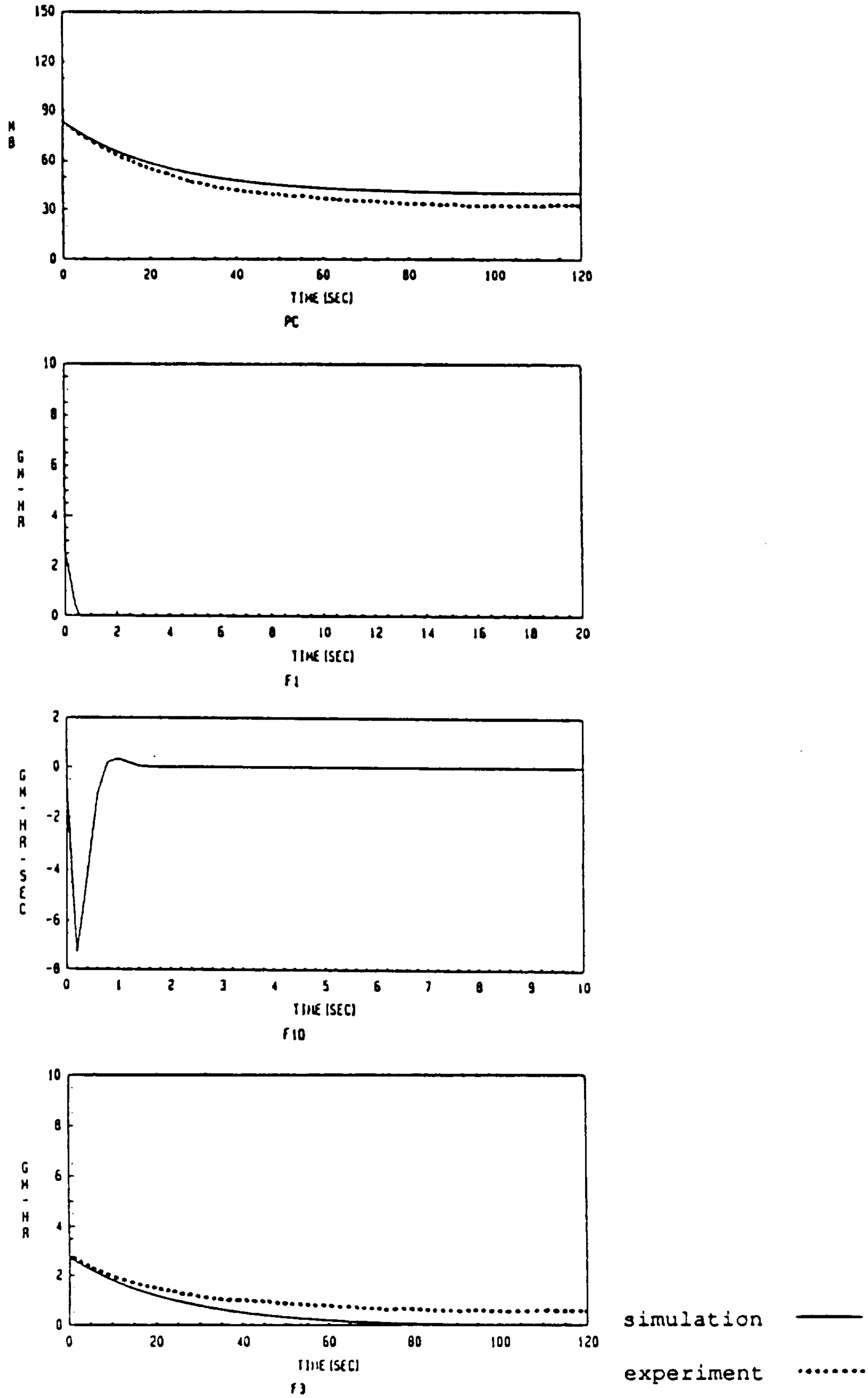


Fig. 5.5-3c Numerical Simulation(F1 is changed from 25 to 0 sccm) 103

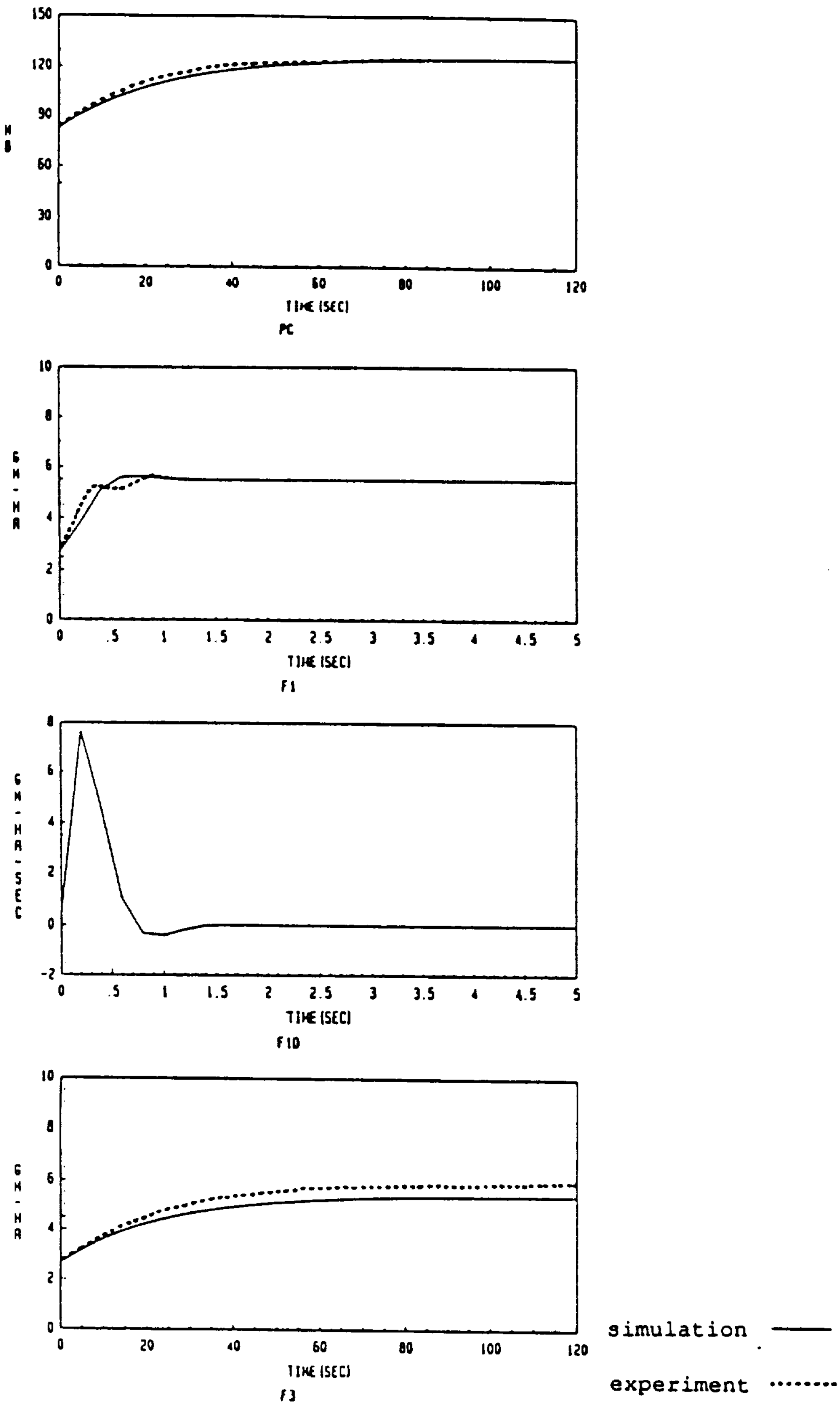


Fig. 5.5-3d Numerical Simulation(F1 is changed from 25 to 50 sccm) 104

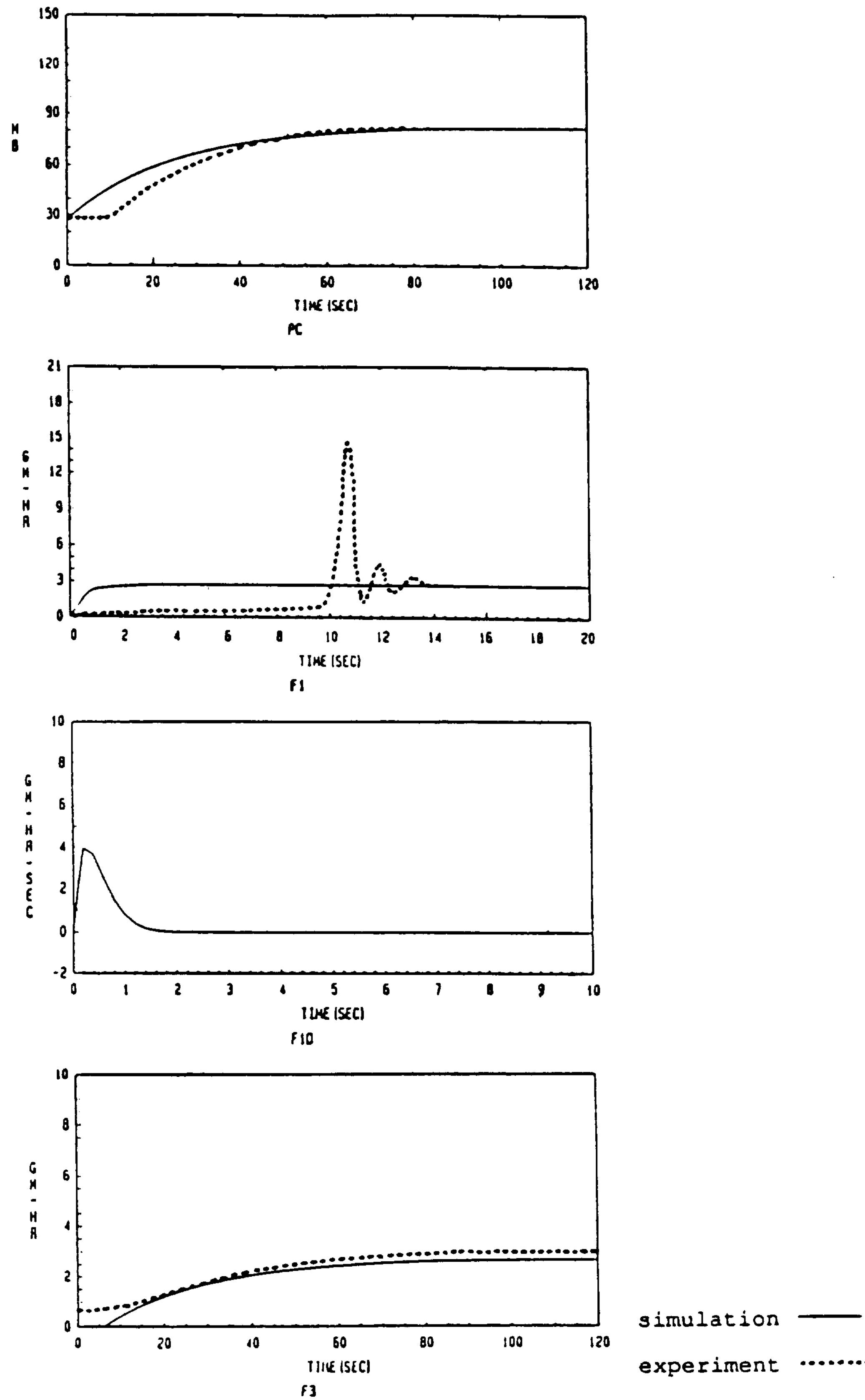


Fig. 5.5-3e Numerical Simulation(F1 is changed from 0 to 25 sccm)

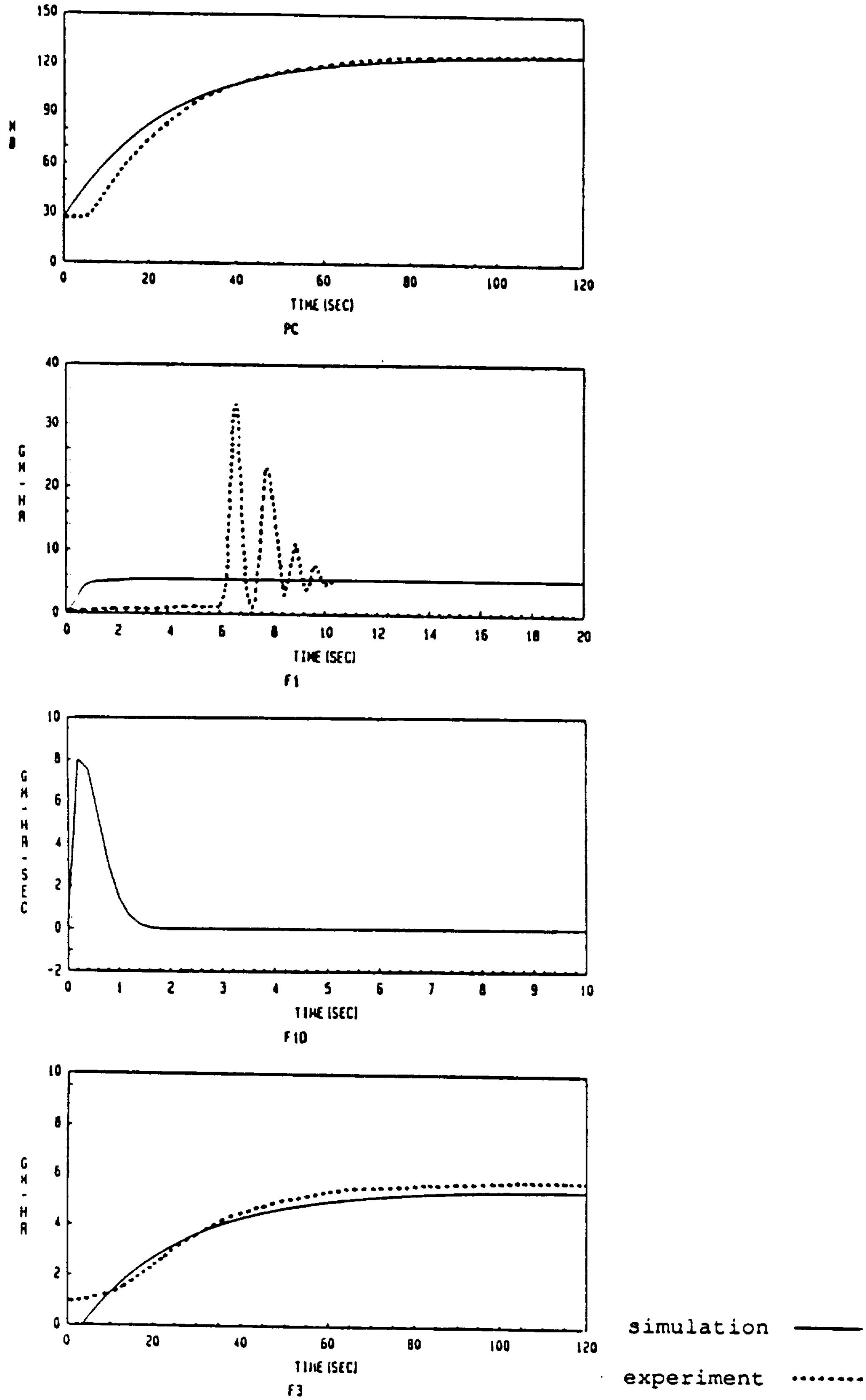


Fig. 5.5-3f Numerical Simulation(F1 is changed from 0 to 50 sccm)

Chapter 6

Preliminary Reliability Analysis

In order to develop a new technology as a viable option for supporting early missions to Mars, it is very important to show that this system can be operated for long periods of time with sufficiently high reliability [33]. The reliability of the MOD system is composed of two parts. One is the reliability of the monitoring (sensor) subsystem and the other is the reliability of the processor subsystem. One may consider that these two parts are both essential to assure the "healthy condition" of the overall system and therefore they are associated with each other in a series structure for reliability assessment (Fig. 6.1). Alternatively, one may consider that the system is in "healthy condition" as long as the oxygen cell matrix is functioning properly and reasonable amount of oxygen is being produced in the oxygen pipeline. Therefore, reliabilities of the monitoring subsystem and the processor subsystem can be studied individually and independently (Fig. 6.2). Figures 6.3 and 6.4 show the reliability block diagrams for the both subsystems in details. The series structure of the diagrams are for reliability assessment and should not be confused with functional block diagrams. The implication of the series arrangement is that each of the block functions must operate for mission success. The reliability of each subsystem is the product of the reliabilities of the individual units.

Reliability, as defined here, is the probability that the subsystems (or the units) will operate successfully during the phase of mission that requires oxygen production. If a constant failure rate is assumed for each subsystem (or unit) the reliability is given by

$R = \exp(-t/MTTF)$

where t = mission time

and MTTF = mean-time-to-failure for a certain subsystem (or unit).

Reliability of the overall system can be estimated by knowing the reliabilities of the individual units. The reliability of each individual unit may be obtained in several ways as follows :

(1) Data from laboratory testing where simulated operating conditions are applied to a number of the units of the same type.

(2) Field data from observation and historical records during controlled operational test.

(3) Field data from similar parts or components used in a variety of applications. This may be provided by the manufacturing company.

(4) Data by analysis and modeling

No comprehensive study of reliability analysis for the MOD system has been done so far. However, several failure types were observed and recorded during operational test. This information may be useful for future reliability analyses. Table 6-1 lists the failure types that occured and were repaired during the operational test period which began in August 1988 and ended in May 1989. The total operating time is about 750 hours or is equivalent to 2.5 hours per day.

Table 6-1

Failure type	Causes	Time happened	Repaired ?
Oven failure	Controller wires short circuit	October, '88	Yes
DAC channels failure	Overpotential	October, '88	Change to other channels
Computer hard disk failure		November, '88	Yes
F1 controller failure	Signal wire bad connection	December, '88	Yes
Leakage at cap of the cell		February, '89	Yes

Oven failure	Power cable melted	May, '89	Yes
Thermocouple Failure		May, '89	Yes

Note: Clip oxidization degrades the performance of the cell obviously and the clip has to be replaced every 150 hours of operating period.

Discussion of The Impact of Power Failure :

It is possible that a power failure might occur during operating period. Therefore, it is very important to design the system in a way that a power failure will not cause any severe damage to the system. Basically, a power failure will not hurt the MOD system immediately because both the feed-gas flow controller and the solenoid isolation valve will shut off automatically when power is removed. Therefore, the pipe network and the ballast tank will be isolated and kept at low pressure. However, a pressure difference will then be created between the tank and the vacuum pump (the pressure in the vacuum pump will tend toward atmospheric pressure). When power recovers again, the vacuum pump will be powered up and the solenoid valve will open automatically. It is possible that pump oil will flow into and contaminate the pipe system at this moment because of the existing pressure difference. Furthermore, the flow controller will not work unless the correct procedure to initialize the computer program is performed again These are the problems that need to be addressed in the future. The ideal design is that the computer will be able to execute the data acquisition and expert system program automatically each time when power fails and recovers so that feed-gas flow-rate will be reset correctly. Also, a pressure transducer measuring the vacuum pump pressure is necessary to compare the vacuum pump pressure and tank pressure so that the computer will be able to determine the appropriate time to open the solenoid valve to prevent undesired pump oil contamination. (Note that each time the SERIES 500 box is powered up, the control signal for the flow controller is set to -10V until the computer program is able to reset the output.)

Redundancy Consideration :

One way to improve reliability of a system is to use redundant components. Two types of redundancy can be distinguished by specifying the manner in which the redundant components of a system are configured. A system may be configured so that all redundant components are operating at all times, which is referred to as active redundancy. Another way is that all redundant components are standby until needed and are then switched into service, which is referred to as standby redundancy.

The equation for computing reliability of a system with parallel active redundancies is [33] :

$$R_s = 1 - (1 - R)^n$$

where R_s is system reliability and R is component reliability,

n is the total number of redundant components.

For example, in order for any subsystem of the MOD machine to work successfully for one year with reliability of 0.9, the mean time to failure can be derived as follows:

$1 - 365 * 24 * FR = 0.9 \longrightarrow FR = 1.14155\ E\text{-}5$

Mean Time To Failure (MTTF) = 1/FR = 87600.19 hrs

If the same subsystem reliability is desired and 3 redundant components are used, then :

$0.9 = 1 - (1 - R')^3 \longrightarrow R' = 0.53584$

$1 - 365 * 24 * FR' = 0.53584 \longrightarrow FR' = 5.2986\ E\text{-}5$

MTTF = 1/(5.2986E-5) = 18872.91 hrs

This means that we can achieve the same subsystem reliability by using three redundant components of smaller reliablity (about 0.59 times smaller than that of the simplex system) with cheaper prices.

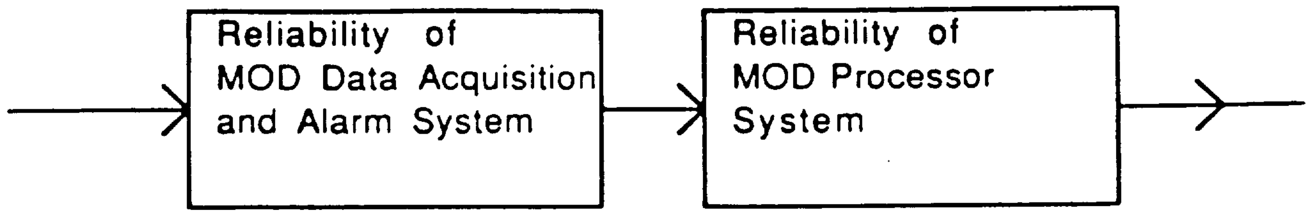


Fig. 6.1 Reliability Block Diagram (series structure)

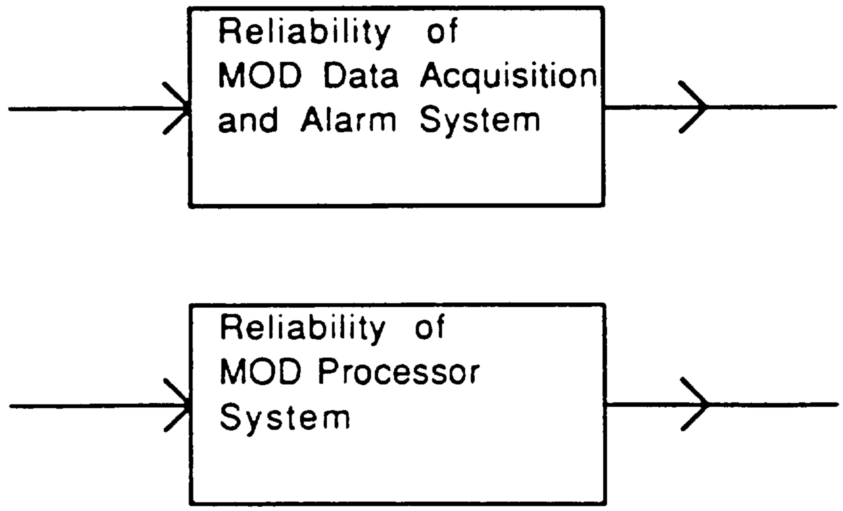


Fig. 6.2 Reliability Block Diagram (parallel structure)

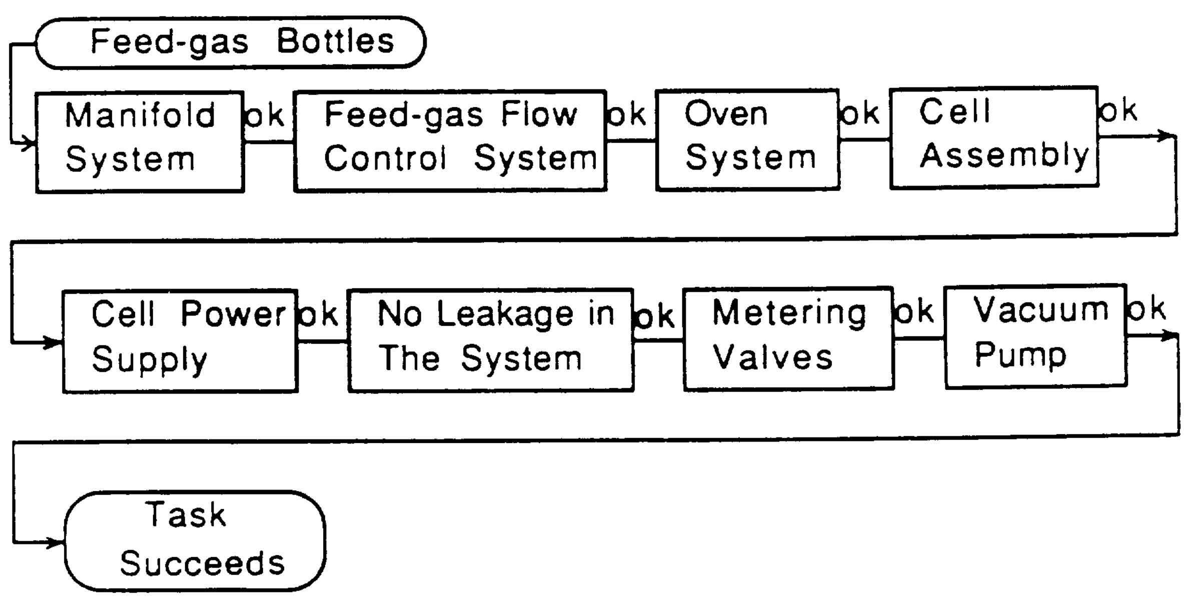


Fig. 6.3 Reliability Block Diagram of The Processor Subsystem

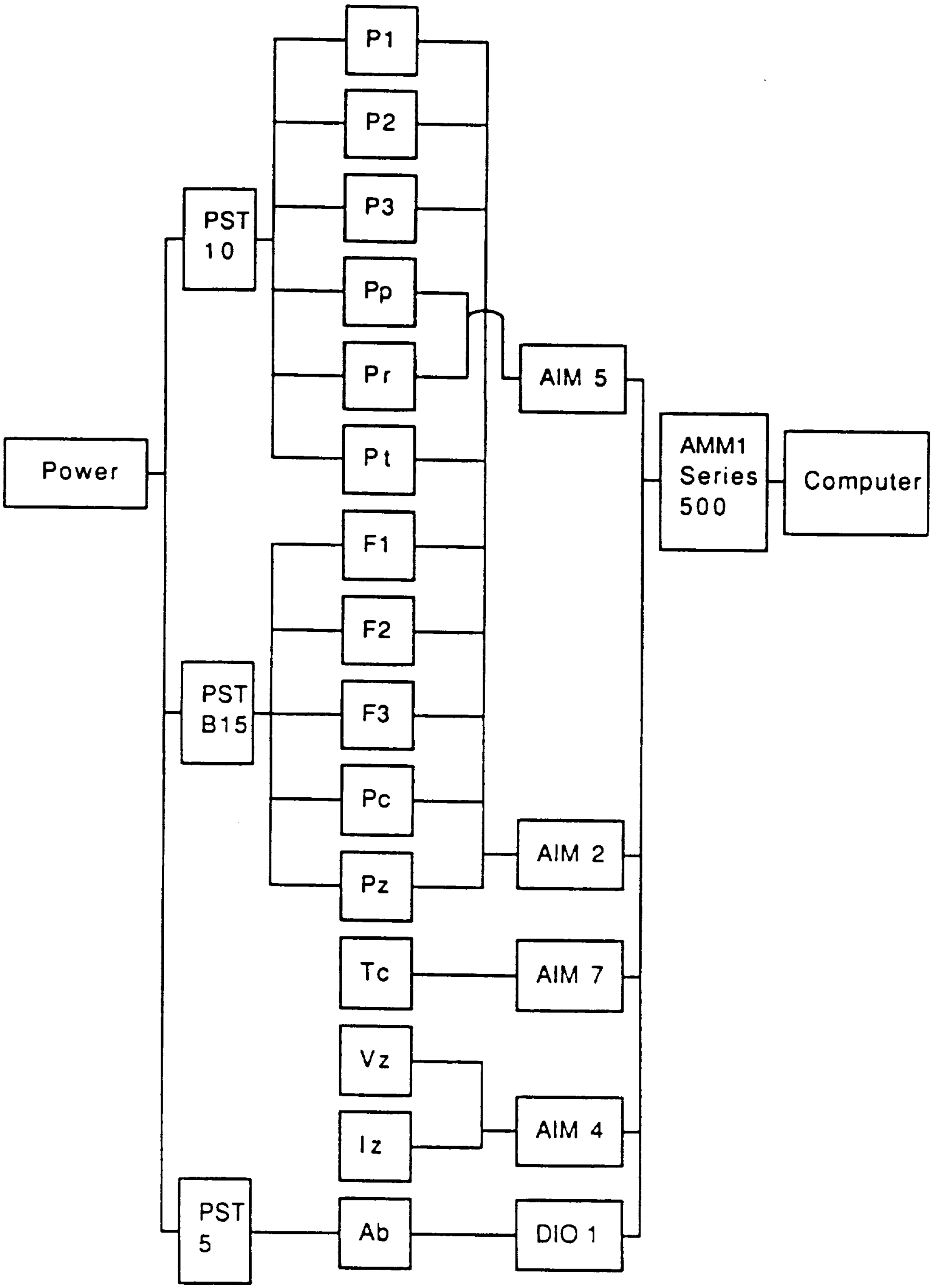


Fig. 6.4 Reliability Block Diagram of The Monitoring Subsystem

Chapter 7

Concluding Remarks

The hardware setup of a prototype Mars oxygen production system has been built. Several operation tests have been performed to obtain the optimal operating conditions which provide maximum oxygen production or production efficiency. The data acquisition system has been set up to acquire the system data and control the feed-gas flow-rate.

Several possible failure modes were tested and algorithms for failure detection and isolation were developed based on the failure tests. The method using forward chaining was more efficient but harder to modify. The method using backward chaining was more suitable for experimental tests but less efficient. The expert system shell "1st-Class" was very helpful in developing a knowledge-base but not so good in handling numerical data. The artificial intelligence language "Prolog" has similar features to those of "1st-Class" and is more flexible for future development.

An effort has been made to derive the dynamic model of the system using the Bond graph method. The results of numerical simulation were compared with the experimental data. Better agreement between the simulation results and the experimental data were achieved for the cases when feed-gas flow-rate was decreased rather than increased. Furthermore, for the cases when feed-gas flow-rate was increased, the agreement was not good. This phenomenon showed that the model was incomplete. The model needs to be modified by taking into account the real gas behaviors such as orifice effect, compressibility and thermal effects,...etc. These effects cause the nonlinear nature of the system which makes the controller dynamics complicated. Once a good model is derived, the model-based failure detection and isolation scheme by estimation presented in Chapter 4 can be applied.

A preliminary study on reliability analysis of the system was also provided. The reliability of each individual component has to be determined first and the overall system reliability can be assessed using technique of reliability of combinatorial components.

LIST OF REFERENCES

1. Paine, T.O., Pioneering the Space Frontier; Report of the National Commision on Space, Bantam Books, New York, 1986.

2. Ash, R.L., W.L. Dowler and G. Varsi, "Feasibility of Rocket Propellant Production on Mars," Acta Astronautica, Vol. 5, pp 705-724, 1978.

3. Stancati, M.L., et al, "In Situ Production for Improved Sample Return Mission Performance," Astrodynamics 1979, Vol. 40, Advances in Astronautical Sciences, pp 909-921, 1980.

4. Ash, R.L., et al, "Autonomous Oxygen Production for a Mars Return Vehicle," XXXIII Congress of the International Astronautical Federation, Paris, IAF paper No. 82-210, 1982.

5. Frisbee, R.H., J.R. French Jr. and E.A. Lawton, "A New Look at Oxygen Production on Mars -- In Situ Propellant Production (ISPP)," AIAA 25th Aerospace Science Meeting, Reno, NV, AIAA Paper No. 87-0236, 1987.

6. Owen, T. et al, "The Composition of the Atmosphere at the Surface of Mars," J. Geophysical Research, Vol.82, pp 4635-4639, 1977.

7. Richter, R., "Basic Investigation into the Production of Oxygen in a Solid Electrolyte Process," AIAA 16th Thermophysics Conference, Palo Alto, CA, AIAA Paper No. 81-1175, 1981.

8. Ash, R.L., J.-K. Huang, P.B. Johnson and W.E. Sivertson Jr., "Elements of Oxygen Production Systems Using Martian Atmosphere," AIAA/ASME/SAE/ASEE 22nd Joint Propulsion Conference, Huntsville, AL, AIAA Paper No. 86-1586, 1986.

9. San Giovanni, J.P. and H.C. Romans, "Expert Systems in Industry: A Survey," Chemical Engineering Progress, pp 52-59, Sept. 1987.

10. Malin, J.T. and N. Lance, Jr., "An Expert System Approach To Automated Fault Management in a Regenerative Life Support Subsystem," Presented at the AIAA 24th Aerospace Sciences Meeting, Reno, Nevada, AIAA paper 86-0321, 1986.

11. Aggarwal, B.B., J.C. Giordano and S.J. Przybylko, "Knowledge-based Systems for Rotodynamic Diagnostics," Presented at the 22nd AIAA/ASME/SAE/ASEE Joint Propulsion Conference, Huntsville, AL, Paper No. 86-1678, June 1986.

12. Liubakka, M.K., G. Rizzoni and W.B. Ribbens, "Failure Detection Algorithms Applied to Control System Design for Improved Diagnostics and Reliability," SAE International Congress and Exposition, SAE paper No. 88-0726, Detroit, Michigan, Feb. 1988.

13. Chien, T.T. and M.B. Adams, "A Sequential Failure Detection Technique and Its Application," IEEE Trans. Automatic Control Vol. AC-21, pp 750-757, Oct. 1976.

14. Clark. R.N., D.C. Fosch and V.M. Walton, "Detection Instrument Malfunctions in Control Systems," IEEE Trans. Aerosp. Electron. System, Vol. AES-11, pp 465-473, July 1975.

15. Degst, J.J. and J.C. Deckert, "Maximum Likelihood Failure Detection Techniques Applied To the Shuttle RCS Jets," J. Spacecraft, Rockets, Vol. 13, Feb. 1976.

16. Gilmore, J.P. and R.A. Mckern, "A Redundant Strapdown Inertial Reference Unit (SIRU)," J. Spacecraft, Vol.9, No.1, pp 39-47, Jan. 1972.

17. Willsky, A.S., "A Survey of Design Methods for Failure Detection in Dynamic Systems," Automatica, Vol.12, pp 601-611, 1976.

18. Murotsn, H.O.Y. and F. Terui, "Failure Detection and Identification in the Control of Large Space Structures," Proc. the 10th IFAC World Congress on Automatic Control, Munich, July 27-31, 1987.

19. Evans, F.A. and J.C. Wilcox, "Experimental Strapdown Redundant Sensor Inertial Navigation System," J. Spacecraft, Vol.7, pp 1070-1074, No.9, 1970.

20. Walker, B.K. and E.G. Gai, "Fault Detection Threshold Determination Technique Using Markov Theory," J. Guidance and Control, Vol.2, No.4, pp 313-319, July-August 1979.

21. Van Trees, H.L., Detection Estimation and Modulation Theory, Part 1, Wiley and Sons, New York, 1968.

22. Gai, E.G., "Determination of Failure Thresholds in Hybrid Navigation," IEEE Transaction on Aerospace and Electronic Systems, Vol. AES-12, No.6, pp 744-755, November 1976.

23. Marie, J.L., "Failure Detection and Correction in Low Orbit Satellite Attitute Control System," IFAC Automatic Control in Space, Noordwijkerhout, The Netherlands, 1982.

24. Ash, R.L., J.A. Werne and M.B. Haywood, "Design of a Mars Oxygen Processor," Case for Mars III, edited by C. Stoker, AAS Science and Technology Series, Vol.75, pp 479-487, 1989.

25. Richter, R., "Basic Investigation into the Electrical Performance of Solid Electrolyte Membranes," JPL publication 82-75, August 15, 1982.

26. Mollenkamp R.A., "Expert Systems in Process Plants," AIChE TODAY SERIES, Process Technology Corporation, New York, 1986.

27. Huang, J.-K., Y.-H. Wei, R.L. Ash and M.-T. Ho, "An Expert Systems Approach to Automated Maintenance for a Mars Oxygen Production System," SAE 18th Intersociety Conference on Environmental Systems, San Francisco, CA, SAE Paper No. 88-1056, July 11-13, 1988.

28. Karnopp, D.C., "State Variables and Pseudo Bond Graphs for Compressible Thermofluid Systems," Journal of Dynamic Systems, Measurement and Control, Vol.101, pp 201-204, September 1979.

29. Collacott, R.A., Mechanical Fault Diagnosis, Chapman and Hall, London, distributed in the U.S.A. by Halsted Press, New York, 1977.

30. Karnopp, D.C., Introduction to Physical System Dynamics , McGraw-Hill Book Company, 1983.

31. Isermann, R., "Process Fault Detection Based on Modeling and Estimation Methods - A Survey," Presented at the 6th IFAC Symposium

on Identification and System Parameter Estimation, Washington, pp 387-404, D.C., June 1982.

32. Ash, R.L., "Mars Oxygen Production System Design," to appear in Proceedings of the 4th Annual NASA/USRA University Advanced Design Program Conference, Kennedy Space Center, June 14-17, 1988.

33. Barlow, R.E. and F. Proschan, Statistical Theory of Reliability and Life Testing , Published by TO BEGIN WITH, 1981.

34. Ogata, K., System Dynamics , Prentice-Hall, Inc., 1978.

APPENDIX A. COMPUTER PROGRAM "MODFDI2.BAS"

(Refer to Section 3.3.2 for description.
The flow chart is shown in Fig. 3.5.)

```
500 '*********************************************************
502 '*                      PROGRAM 1                        *
504 '* This program is the common part to MODFDI1,           *
506 '* MODFDI2.BAS and MODFDI3.                              *
508 '*********************************************************
1000 KEY OFF    'Turn off the softkeys on line 25 of display
1010 FOR I = 1 TO 10 'Disable all the softkeys
1020   KEY I,""
1030 NEXT I
1040 SCREEN 9,0  'High-resolution color screen mode
1500 '*******************************************************
1505 '* Initialize constants that are going to be used in  *
1510 '* this program                                        *
1515 '*******************************************************
1670 STAT = 0
1680 FM1ROW = 4           'Flow meter 1 is displayed on row 4
1690 FM1COL = 4
1695 PTCROW = 2
1700 PT1ROW = 3
1705 PTCCOL = 4
1710 PT1COL = 4
1720 TC1ROW = 8
1730 TC1COL = 14
1740 S1ROW  = 3
1750 S1COL  = 66
1760 FM2ROW = 4
1770 FM2COL = 41
1775 PTZROW = 2
1780 PT2ROW = 3
1785 PTZCOL = 41
1790 PT2COL = 41
1800 TC2ROW = 8
1810 TC2COL = 47
1820 OTCROW = 9
1830 OTCCOL = 30
1840 FM3ROW = 22
1850 FM3COL = 33
1860 PT3ROW = 21
1870 PT3COL = 33
1880 TC3ROW = 17
1890 TC3COL = 38
1900 S2ROW  = 22
1910 S2COL  = 65
1920 TANROW = 15
1930 TANCOL = 62
1932 CEVROW = 12
1934 CEVCOL = 40
1936 CEIROW = 13
1938 CEICOL = 40
1940 BTAROW = 20
1942 BTACOL = 2
1944 BTBROW = 21
1946 BTBCOL = 2
1950 F1KEY$  = CHR$(59)
```

```
1960 F2KEY$  = CHR$(60)
1970 F3KEY$  = CHR$(61)
1980 F4KEY$  = CHR$(62)
1990 F5KEY$  = CHR$(63)
2000 F6KEY$  = CHR$(64)
2010 F7KEY$  = CHR$(65)
2020 F8KEY$  = CHR$(66)
2030 F9KEY$  = CHR$(67)
2040 F10KEY$ = CHR$(68)
2050 ERRMSGROW = 25
2060 ERRMSGCOL = 1
2975 '***********************************************************
2980 '* Initialize strings that are to be used to draw the *
2985 '* graphic symbols for the online-update screen.      *
2990 '***********************************************************
2995 '
3000 '********************************
3005 '*  Solenoid Isolation Valve    *
3010 '********************************
3015 SOLENOID1$ = "F1R1F1R1F1R1F1R1F2R1F1R1F1R1F1R1F1U10"
3020 SOLENOID2$ = "G1L1G1L1G1L1G1L1G2L1G1L1G1L1G1L1G1U10"
3025 SOLENOID$  = SOLENOID1$ + SOLENOID2$
3030 '
3035 '********************************
3040 '*  Sample Port Apparatus       *
3045 '********************************
3050 PORTAPP1$ = "R10G1D1G1D1G1D1G1D1G2D1G1D1G1D1G1D1G1R10"
3055 PORTAPP2$ = "H1U1H1U1H1U1H1U1H2U1H1U1H1U1H1U1H1"
3060 PORTAPP$  = PORTAPP1$ + PORTAPP2$
3065 '
3070 '********************************
3075 '*  Mars Gas - K size tanks     *
3080 '********************************
3084 'position cursor to middle of top of tank
3085 GASTANK1$ = "BR12"
3089 'upper right half of circle of tank
3090 GASTANK2$ = "R3F1R2F5D2F1D3"
3094 'bottom rectangle of tank
3095 GASTANK3$ = "D66L24U66"
3099 'upper left half of circle of tank
3100 GASTANK4$ = "U3E1U2E5R2E1R3"
3105 GASTANK$=GASTANK1$ + GASTANK2$ + GASTANK3$ + GASTANK4$
3110 '
3115 '********************************
3120 '*  1/4 Inch "T"                *
3125 '********************************
3130 NORMALT$ = "R26D6L10D4L6U4L10U6"
3135 '
3140 '********************************
3145 '*  1/4 Inch Upside Down "T"    *
3150 '********************************
3155 UPSIDET1$ = "BR10"
3160 UPSIDET2$ = "R6D4R10D6L26U6R10U4"
3165 UPSIDET$  = UPSIDET1$ + UPSIDET2$
```

```
3170 '
3175 '*********************************
3180 '*  Flow Meter                   *
3185 '*********************************
3189 'Position cursor to top corner of box
3190 FLOWM1$ = "BD13"
3194 'Draw box and return to top corner of box
3195 FLOWM2$ = "R26D10L26U10"
3199 'Move cursor to center of the first circle
3200 FLOWM3$ = "BR7BU6"
3204 'draw first circle
3205 FLOWM4$ = "BL4U2E1U1E2R2F2D1F1D4G1D1G2L2H2U1H1U2BR4"
3209 'Move cursor to start of drawing squiggly line
3210 FLOWM5$ = "BR3BU3"
3214 'Draw squiggly line above two circles
3215 FLOWM6$ = "R5E1R1E1R1E1U1"
3219 'Move cursor to center of second circle
3220 FLOWM7$ = "BL1BD8"
3224 'draw second circle
3225 FLOWM8$ = "BL3U2E1U1E1R2F1D1F1D4G1D1G1L2H1U1H1U2BR3"
3230 FLOWM$  = FLOWM1$+FLOWM2$+FLOWM3$+FLOWM4$+FLOWM5$
3235 'FLOWM$ = FLOWM$+FLOWM6$+FLOWM7$+FLOWM$8
3240 '*********************************
3245 '*   Thermocouple                *
3250 '*********************************
3255 THERMOC$ = "R2F4D2F1D5R1U3E1U3E1U2R1U1E1R1"
3260 '
3265 '*********************************
3270 '*   2 Tank Manifold             *
3275 '*********************************
3279 'Draw rectangle and left box
3280 TANKM1$ = "BR9R75D30L75U30BD16L9D9R9U9"
3284 'Position cursor to center of first circle
3285 TANKM2$ = "BU1BR18"
3289 'Draw first circle (radius=3)
3290 TANKM3$ = "BL3U1E2R2F2D2G2L2H2U1BR3"
3294 'Position cursor to center of second circle
3295 TANKM4$ = "BR19"
3299 'Draw second circle (radius=3)
3300 TANKM5$ = "BL3U1E2R2F2D2G2L2H2U1BR3"
3304 'Position cursor to center of third circle
3305 TANKM6$ = "BR19"
3309 'Draw third circle (radius=3)
3310 TANKM7$ = "BL3U1E2R2F2D2G2L2H2U1BR3"
3314 'Draw right box
3315 TANKM8$ = "BR19BD1R9D9L9U9"
3320 TANKM$  = TANKM1$+TANKM2$+TANKM3$+TANKM4$+TANKM5$
3325 'TANKM$ = TANKM$+TANKM6$+TANKM7$+TANKM8$
3330 '*********************************
3335 '*   Horizontal Heat Exchanger   *
3340 '*********************************
3344 'Position cursor to bottom of heat exchanger
3345 HHEATEX1$ = "BD5"
3350 HHEATEX2$ = "E5F1D1F1D1F1D1F1D1F1E1U1E1U1E1U1E1U1E1F5"
```

```
3355 HHEATEX$  = HHEATEX1$ + HHEATEX2$
3360 '
3365 '*****************************
3370 '*   Vertical Heat Exchanger    *
3375 '*****************************
3379 'position cursor to top of heat exchanger
3380 VHEATEX1$ = "BR5"
3385 VHEATEX2$ = "G5R2F2R2F2R1F1G1L1G2L2G2L2F5"
3390 VHEATEX$ = VHEATEX1$ + VHEATEX2$
3395 '
3400 '***********************************
3405 '*   High Accuracy Pressure Transducer *
3410 '***********************************
3415 HAPRESST$ = "BL6U2E1U1E2R1U1R4D1R1F2D1F1D4G1D1G2L1D1L
3420 'HAPRESST$=HAPRESST$+"L4U1L1H2U1H1U2BR6"
3425 '**********************
3430 '*   Zirconia Oxide Cell *
3435 '**********************
3440 OXIDEC1$ = "BL8U2E1U1E1R1E1R1E1R4F1R1F1R1F1D1F1D4"
3445 OXIDEC2$ = "G1D1G1L1G1L1G1L4H1L1H1L1H1U1H1U2BR8"
3450 OXIDEC$  = OXIDEC1$ + OXIDEC2$
3455 '
3460 '**********************
3465 '*   Pressure Transducer *
3470 '**********************
3475 PRESST$  = "BL5U1E3R4F3D2G3L4H3U1BR5"
3480 '
3485 '***************************
3490 '*   Carbon Steel Vacuum Tank *
3495 '***************************
3500 VACUUMT1$ = "BL26U2E1U1E4R2E1R2E1R30F1R2F1R2F4D1F1D4"
3505 VACUUMT2$ = "G1D1G4L2G1L2G1L30H1L2H1L2H4U1H1U2BR26"
3510 VACUUM$   = VACUUMT1$ + VACUUMT2$
3515 '
3520 '***************************
3525 '* Little Circle             *
3530 '***************************
3535 LCIRCLE$ = "BL3U1E2R2F2D2G2L2H2U1BR3"
3540 '
3545 '***************************
3550 '* Little Rectangle          *
3555 '***************************
3560 LRECT$ = "R8D8L8U8"
3565 '
3570 '***************************
3575 '* Medium Circle             *
3580 '***************************
3585 MCIRCLE$ = "BL7U1E1U1E2R1E1R4F1R1F2D1F1D2G1D1G2L1G1L
3590 'MCIRCLE$=MCIRCLE$+"L4H1L1H2U1H1U1"
3595 '***************************
3600 '* Belt Mount                *
3605 '***************************
3610 BELTMOUNT$ = "BR5R12F2D1F1D1L20U2E1R1E1R1E1"
3615 '
```

```
3620 '******************************
3625 '* Arrow Head                  *
3630 '******************************
3635 ARROWHEAD$ = "R1F1R1F1G1L1G1L1U4"
3640 '
3645 '******************************
3650 '* Oven                        *
3655 '******************************
3660 OVEN$ = "R50D40L50U40"
3665 '
3670 '********************************
3675 '* Upside Down Flow Meter       *
3680 '********************************
3685 UFLOWM1$ = "R26D10L26U10"
3690 UFLOWM2$ = "BR7BD16"
3694 'draw first circle
3695 UFLOWM3$ = "BL4U2E1U1E2R2F2D1F1D4G1D1G2L2H2U1H1U2BR4"
3699 'move cursor to start of drawing squiggly line
3700 UFLOWM4$ = "BR3BD3"
3704 'draw squiggly line
3705 UFLOWM5$ = "R5F1R1F1R1F1D1"
3709 'move cursor to center of second circle
3710 UFLOWM6$ = "BL1BU8"
3714 'draw second circle
3715 UFLOWM7$ = "BL3U2E1U1E1R2F1D1F1D4G1D1G1L2H1U1H1U2BR3"
3720 UFLOWM$  = UFLOWM1$+UFLOWM2$+UFLOWM3$+UFLOWM4$+UFLOWM5$
3725 'UFLOWM$ = UFLOWM$+UFLOWM5$+UFLOWM6$+UFLOWM7$
3730 '********************************
3735 '* Upside Down Thermocouple     *
3740 '********************************
3745 UTHERMOC1$ = "BD12"
3750 UTHERMOC2$ = "R2E4U2E1U5R1D3F1D3F1D2R1D1F1R1"
3755 UTHERMOC$  = UTHERMOC1$ + UTHERMOC2$
3760 '
```

```
10000 '****************************************************
10005 '*                    PROGRAM  2                    *
10010 '*                                                  *
10015 '*  1) Do the initialization for the Keithley Data  *
10020 '*     Acquisition System                           *
10025 '*  2) Bring up the main system menu                *
10030 '****************************************************
10035 '
10040 '**********************************
10043 '* Keith Lee Initialization       *
10045 '**********************************
10046 STAT%=0:C0=0:C1=0:C2=0:C3=0:C4=0:C5=0:C6=0:C7=0:C8=0
10048 A0=0:A1=0:A2=0:A3=0:A4=0:A5=0:A6=0:A7=0
10049 A8=0:A9=0:A10=0:A11=0:A12=0:A13=0:A14=0:A15=0
10050 H0=0:H1=0:L0=0:L1=0
10055 'C=AIM7  A=AIM2  H=AIM4  L=AIM5  V=AOM1
10058 CALL INIT
10060 CALL IONAME'("coldjunc",2,32,12)
10070 CALL IONAME'("C0",2,0,12)
10080 CALL IONAME'("C1",2,1,12)
10090 CALL IONAME'("C2",2,2,12)
10100 CALL IONAME'("C3",2,3,12)
10101 CALL IONAME'("C4",2,4,12)
10102 CALL IONAME'("C5",2,5,12)
10110 CALL IONAME'("A0",4,0,12,1)
10120 CALL IONAME'("A1",4,1,12,1)
10130 CALL IONAME'("A2",4,2,12,1)
10140 CALL IONAME'("A3",4,3,12,1)
10150 CALL IONAME'("A4",4,4,12,1)
10160 CALL IONAME'("A5",4,5,12,1)
10170 CALL IONAME'("A6",4,6,12,1)
10180 CALL IONAME'("A7",4,7,12,1)
10190 CALL IONAME'("A8",4,8,12,1)
10200 CALL IONAME'("A9",4,9,12,1)
10210 CALL IONAME'("A10",4,10,12,1)
10220 CALL IONAME'("H0",7,0,12,1)
10230 CALL IONAME'("H1",7,1,12,1)
10240 CALL IONAME'("L0",6,0,12,1)
10250 CALL IONAME'("L1",6,1,12,1)
10251 CALL IONAME'("A15",4,15,12,1)
10255 CALL IONAME'("V0",5,0)
10257 CALL IONAME'("D31",8,31)
10258 CALL DIGWRITE'("D31",1.0)
10259 '
10260 '
10500 '**************************
10510 '*  Main Menu Selections  *
10520 '**************************
10530 '
10535 CLS
10540 LOCATE 1,10
10550 PRINT "OLD DOMINION UNIVERSITY,
10551 'PRINT "MECHANICAL ENGINEERING DEPARTMENT"
10560 LOCATE 2,31
```

```
10570 PRINT "NORFOLK, VIRGINIA"
10580 LOCATE 4,20
10590 COLOR 3,0
10600 PRINT "MARS OXYGEN DEMONSTRATION PROJECT  (MOD)"
10610 COLOR 15,0
10620 'SET UP MENU
10630 STARTROW = 7
10640 STARTCOL = 10
10650 ENDROW = 22
10660 ENDCOL = 70
10670 GOSUB 30000  'Call the window routine!
10680 LOCATE 9,35
10690 PRINT "MAIN  MENU"
10700 LOCATE 12,14
10710 PRINT "F1 .................... Real-Time Update Screen"
10720 LOCATE 14,14
10730 PRINT "F2 ................... Expert System For F.D.I."
10740 LOCATE 16,14
10750 PRINT "F3 ......... Analyze Individual Input Readings"
10760 LOCATE 18,14
10770 PRINT "F4 ................. General System Information"
10780 LOCATE 20,14
10790 PRINT "F10 ......................... Exit This Program"
10795 '
10799 'Disable all the softkeys in case any were enabled
10800 FOR I = 1 TO 10
10810   KEY(I) OFF
10820 NEXT I
10829 'Enable all the softkeys allowed in this menu
10830 FOR I = 1 TO 4
10840    KEY(I) ON
10850 NEXT I
10860 KEY(10) ON
10865 '
10870 ON KEY(1) GOSUB 40000
10880 ON KEY(2) GOSUB 50000
10890 ON KEY(3) GOSUB 60000
10900 ON KEY(4) GOSUB 62000
10910 ON KEY(10) GOSUB 65000
10930 '
10990 GOTO 10930
```

```
20000 '*****************************************************
20001 '                     PROGRAM    3                    *
20002 '* Start checking the data from the Keith Lee system *
20004 '*****************************************************
20010 CALL ANREAD'("C0",C0,11)
20020 CALL ANREAD'("C1",C1,13)
20030 CALL ANREAD'("C2",C2,13)
20040 CALL ANREAD'("C3",C3,13)
20050 CALL ANREAD'("A0",A0,0)
20060 CALL ANREAD'("A1",A1,0)
20070 CALL ANREAD'("A2",A2,0)
20080 CALL ANREAD'("A3",A3,0)
20090 CALL ANREAD'("A4",A4,0)
20100 CALL ANREAD'("A5",A5,0)
20110 CALL ANREAD'("A6",A6,0)
20120 CALL ANREAD'("A7",A7,0)
20122 CALL ANREAD'("A8",A8,0)
20130 CALL ANREAD'("L0",L0,1)
20140 CALL ANREAD'("L1",L1,1)
20150 CALL ANREAD'("H0",H0,0)
20160 CALL ANREAD'("H1",H1,0)
20162 '
21000 TC=C0+273!
21001 VZ=H1
21002 F1=A1*8.057
21004 F2=A2*.16
21006 F3=A3*8.057
21007 IZ=(H0/5.38)*1000
21008 S1$="P"
21010 PP=(L0/30!)*1000!
21012 PR=(L1/30!)*1000!
21014 PD=40!
21016 PC=(A4*10)*1.3332
21018 PZ=(A8*10)*1.3332
21020 P1=(A5*(-163.1278)+186.2344)+PA
21022 P2=(A7*(-170.1535)+230.4117)+PA
21024 P3=(A6*(-165.0467)+203.1309)+PA
21025 PT=(A0*(0))
21026 T0=C1
21030 '
21100 CALL ANWRITE'("V0",V0,0)
21102 '
25000 RETURN
```

```
30000 '*****************************************************
30003 '*                    PROGRAM    4                    *
30005 '*                     SUBROUTINE                     *
30007 '*****************************************************
30008 'WINDOW SUBROUTINE
30010 LOCATE STARTROW,STARTCOL
30020 PRINT CHR$(201);
30030 FOR I=(STARTCOL+1) TO (ENDCOL-1)
30040    PRINT CHR$(205);
30050 NEXT I
30060 PRINT CHR$(187);
30070 FOR I=(STARTROW+1) TO (ENDROW-1)
30080    LOCATE I,STARTCOL
30090    PRINT CHR$(186);
30100    LOCATE I,ENDCOL
30110    PRINT CHR$(186);
30120 NEXT I
30130 LOCATE ENDROW,STARTCOL
30140 PRINT CHR$(200);
30150 FOR I=(STARTCOL+1) TO (ENDCOL-1)
30160    PRINT CHR$(205);
30170 NEXT I
30180 PRINT CHR$(188);
30190 RETURN
40000 '*****************************************************
40001 CLS
40010 FOR I = 1 TO 10
40020   KEY(I) OFF
40030 NEXT I
41000 LOCATE 11,10:INPUT "Barometric pressure (mb)";PA
41001 LOCATE 12,10:PRINT "The set point you use now is ";V0
41002 LOCATE 13,10:INPUT "Flow set point to be used(0-5)";V0
41003 IF V>5 OR V<0 THEN BEEP : GOTO 41002
41004 CALL ANWRITE'("V0",V0,0)
41005 LOCATE 14,10:INPUT "Need 'MOD.LOG' flle (Y/N) ";L$
41006 IF L$="Y" OR L$="y" THEN 41007 ELSE 41009
41007 OPEN "O",#1,"A:MOD.LOG"
41008 CLOSE #1
41009 LOCATE 15,10:INPUT "Data to be printed (Y/N) ";P$
41010 LOCATE 16,10:INPUT "Sampling period (sec) ";SP
41011 SP=SP*3000
41020 LOCATE 17,10:INPUT "Send data every ?th update ";TH
41021 DONE=0
42009 CLS
42010 '*****************************************************
42020 '*          Original Screen Initialization           *
42030 '*****************************************************
42040 DRAW "BM10,124"
42050 DRAW "XTANKM$;"
42053 'draw a pressure transducer above first gas tank
42055 DRAW "BM27,178"
42057 DRAW "XPRESST$;"
42060 DRAW "BM15,182"
42070 DRAW "XGASTANK$;"
```

```
42073 'draw a pressure transducer above second gas tank
42075 DRAW "BM85,178"
42077 DRAW "XPRESST$;"
42080 DRAW "BM73,182"
42090 DRAW "XGASTANK$;"
42100 LINE (10,145)-(5,145)
42110 LINE (5,145)-(5,169)
42120 LINE (5,169)-(27,169)
42130 LINE (27,169)-(27,174)
42170 '
42180 LINE (103,145)-(108,145)
42190 LINE (108,145)-(108,169)
42200 LINE (108,169)-(85,169)
42210 LINE (85,169)-(85,174)
42220 '
42230 LINE (28,124)-(28,114)
42240 DRAW "BM23,96"
42250 DRAW "XPORTAPP$;"
42260 LINE (28,96)-(28,86)
42300 'FLOW METER
42310 LINE (75,124)-(75,74)
42320 LINE (75,74)-(80,74)
42330 DRAW "BM80,56"
42340 DRAW "XFLOWM$;"
42400 'THERMOCOUPLE
42410 LINE (106,74)-(152,74)
42420 DRAW "BM122,62"
42430 DRAW "XTHERMOC$;"
42500 '
42510 '1/4 INCH T AND HIGH ACCURACY PRESSURE TRANSDUCER
42530 DRAW "BM152,67"
42540 DRAW "XUPSIDET$;"
42550 LINE (165,67)-(165,64)
42560 DRAW "BM165,57"
42570 DRAW "XHAPRESST$;"
42600 '
42610 'REGULAR T AND SAMPLE PORT APPARATUS
42620 LINE (178,74)-(192,74)
42630 DRAW "BM192,71"
42640 DRAW "XNORMALT$;"
42650 LINE (205,81)-(205,89)
42660 DRAW "BM200,89"
42670 DRAW "XPORTAPP$;"
42680 DRAW "BM205,111"
42690 DRAW "XLCIRCLE$;"
42700 '
42710 'OVEN AND CELL
42720 LINE (218,74)-(226,74)
42730 LINE (226,74)-(226,174)
42740 LINE (226,174)-(253,174)
42750 DRAW "BM261,174"
42760 DRAW "XOXIDEC$;"
42770 LINE (269,174)-(296,174)
42780 LINE (296,174)-(296,74)
```

```
42790 LINE (296,74)-(309,74)
42800 DRAW "BM236,154"
42810 DRAW "XOVEN$;"
42820 DRAW "BM254,142"
42830 DRAW "XTHERMOC$;"
42900 '** Draw horizontal heat exchanger **
42910 DRAW "BM309,69"
42920 DRAW "XHHEATEX$;"
42930 LINE (329,74)-(349,74)
43000 '** Draw 2nd Flow Meter **
43010 DRAW "BM349,56"
43020 DRAW "XFLOWM$;"
43030 LINE (375,74)-(421,74)
43040 '
43050 '** Draw 2nd Thermocouple **
43060 DRAW "BM391,62"
43070 DRAW "XTHERMOC$;"
43080 '
43090 '** Draw upside T and Pressure Transducer **
43100 DRAW "BM421,67"
43110 DRAW "XUPSIDET$;"
43120 LINE (434,67)-(434,64)
43130 DRAW "BM434,60"
43140 DRAW "XPRESST$;"
43200 '** Draw NormalT and Sample Port Apparatus **
43210 LINE (447,74)-(462,74)
43220 DRAW "BM462,71"
43230 DRAW "XNORMALT$;"
43240 LINE (475,81)-(475,89)
43250 DRAW "BM470,89"
43260 DRAW "XPORTAPP$;"
43270 DRAW "BM475,111"
43280 DRAW "XLCIRCLE$;"
43300 '** Draw Metering Valve **
43310 LINE (488,74)-(503,74)
43320 DRAW "BM503,69"
43330 DRAW "XSOLENOID$;"
43340 LINE (512,74)-(512,64)
43350 LINE (507,64)-(517,64)
43400 '** Draw Solenoid Isolation Valve (P) **
43410 LINE (521,74)-(536,74)
43420 DRAW "BM536,69"
43430 DRAW "XSOLENOID$;"
43500 '** Hook up and draw Vacuum Tank **
43510 LINE (554,74)-(564,74)
43520 LINE (564,74)-(564,164)
43530 LINE (564,164)-(578,164)
43540 ' draw the cylinder of the tank
43550 DRAW "BM599,172"
43560 DRAW "XVACUUM$;"
43570 ' draw the pressure transducer inside the tank
43580 DRAW "BM599,167"
43590 DRAW "XPRESST$;"
43600 ' draw the feet of the tank
```

```
43610 DRAW "BM585,182"
43620 DRAW "XLRECT$;"
43630 DRAW "BM605,182"
43640 DRAW "XLRECT$;"
43650 ' draw the belt mount
43660 DRAW "BM589,156"
43670 DRAW "XBELTMOUNT$;"
43680 ' draw the left and right circles for belt
43690 DRAW "BM591,150"
43700 DRAW "XMCIRCLE$;"
43710 DRAW "BM608,153"
43720 DRAW "XLCIRCLE$;"
43730 ' draw the belt between the two circles
43740 LINE (593,144)-(609,150)
43750 LINE (593,156)-(607,156)
43760 LINE (609,150)-(619,150)
43770 DRAW "BM619,148"
43780 DRAW "XARROWHEAD$;"
43800 'start drawing bottom of diagram
43802 'starting with link to oven cell
43810 LINE (261,181)-(261,209)
43815 'draw the vertical heat exchanger
43820 DRAW "BM256,209"
43830 DRAW "XVHEATEX$;"
43840 'draw the rest of the line -
43842 'until the upside down flow meter
43850 LINE (261,229)-(261,264)  'line down
43860 LINE (261,264)-(275,264)  'line across
43870 'draw upside down flow meter
43880 DRAW "BM275,259"
43890 DRAW "XUFLOWM$;"
43900 'draw upside down thermocouple
43910 LINE (301,264)-(347,264)
43920 DRAW "BM317,264"
43930 DRAW "XUTHERMOC$;"
43950 'draw upside down pressure transducer
43960 DRAW "BM347,261"
43970 DRAW "XNORMALT$;"
43980 LINE (360,271)-(360,274)
43990 DRAW "BM360,278"
43995 DRAW "XPRESST$;"
44000 'draw sample port apparatus
44010 LINE (373,264)-(393,264)
44020 DRAW "BM393,257"
44030 DRAW "XUPSIDET$;"
44040 LINE (406,257)-(406,249)
44050 DRAW "BM401,231"
44060 DRAW "XPORTAPP$;"
44070 DRAW "BM406,227"
44080 DRAW "XLCIRCLE$;"
44100 'draw rectangle with two solenoids
44110 LINE (419,264)-(439,264)   'line to the box
44120 LINE (439,264)-(439,254)   'left upper side of the box
44130 LINE (439,254)-(465,254)   'line to the upper solenoid
```

```
44140 DRAW "BM465,249"          'draw upper solenoid
44150 DRAW "XSOLENOID$;"
44159 'line from upper solenoid to little rectangle
44160 LINE (474,254)-(474,244)
44170 DRAW "BM470,236"         'draw the little rectangle
44180 DRAW "XLRECT$;"
44190 LINE (483,254)-(511,254)'line from upper solenoid
44195 LINE (511,254)-(511,264)'right upper side of the box
44200 LINE (439,264)-(439,274)'left lower side of box
44210 LINE (439,274)-(465,274)'line to the lower solenoid
44220 DRAW "BM465,269"         'draw the lower solenoid
44230 DRAW "XSOLENOID$;"
44239 'line from solenoid to the small line
44240 LINE (474,274)-(474,284)
44250 LINE (469,284)-(479,284)'small line beklow solenoid
44260 LINE (483,274)-(511,274)'line from the lower solenoid
44270 LINE (511,274)-(511,264)'right lower side of the box
44300 'draw the solenoid isolation valve (P) on the bottom
44310 LINE (511,264)-(531,264)
44320 DRAW "BM531,259"
44330 DRAW "XSOLENOID$;"
44350 'connect bottom of diagram to the tank
44360 LINE (549,264)-(564,264)
44370 LINE (564,264)-(564,180)
44380 LINE (564,180)-(578,180)
44390 LOCATE 23,1 : PRINT "Symptom:";
44391 LOCATE 25,1 : PRINT "Press <SPACEBAR> TO EXIT";
44395 EMP38$="                                      "
44396 EMP34$="                                  "
44397 EMP32$="                                "
44398 EMP15$="               "
```

```
44400 LOCATE 1,1:PRINT "*",DATE$;"  ";TIME$;
44410 GOSUB 20000
44500 LOCATE FM1ROW,FM1COL : PRINT EMP38$;
44501 LOCATE FM1ROW,FM1COL : PRINT "F1=";A1;"V=";F1;"gm/hr";
44502 LOCATE FM2ROW,FM2COL : PRINT EMP38$;
44503 LOCATE FM2ROW,FM2COL : PRINT "F2=";A2;"V=";F2;"gm/hr";
44504 LOCATE FM3ROW,FM3COL : PRINT EMP38$;
44505 LOCATE FM3ROW,FM3COL : PRINT "F3=";A3;"V=";F3;"gm/hr";
44506 LOCATE TC1ROW,TC1COL : PRINT EMP15$;
44507 LOCATE TC1ROW,TC1COL : PRINT "Ta=";T0;"C";
44508 LOCATE TC2ROW,TC2COL : PRINT EMP15$;
44509 LOCATE TC2ROW,TC2COL : PRINT "T2=";
44510 LOCATE TC3ROW,TC3COL : PRINT EMP15$;
44511 LOCATE TC3ROW,TC3COL : PRINT "T3=";
44512 LOCATE PT1ROW,PT1COL : PRINT EMP38$;
44513 LOCATE PT1ROW,PT1COL : PRINT "P1=";A5;"V=";P1;"mb";
44514 LOCATE PT2ROW,PT2COL : PRINT EMP38$;
44515 LOCATE PT2ROW,PT2COL : PRINT "P2=";A7;"V=";P2;"mb";
44516 LOCATE PT3ROW,PT3COL : PRINT EMP38$;
44517 LOCATE PT3ROW,PT3COL : PRINT "P3=";A6;"V=";P3;"mb";
44518 LOCATE BTAROW,BTACOL : PRINT EMP34$;
44519 LOCATE BTAROW,BTACOL : PRINT "Pp=";L0;"mV=";PP;"psi";
44520 LOCATE BTBROW,BTBCOL : PRINT EMP34$;
44521 LOCATE BTBROW,BTBCOL : PRINT "Pr=";L1;"mV=";PR;"psi";
44524 LOCATE OTCROW,OTCCOL : PRINT EMP15$;
44525 LOCATE OTCROW,OTCCOL : PRINT "Tc=";TC;"K";
44526 LOCATE CEVROW,CEVCOL : PRINT EMP32$;
44527 LOCATE CEVROW,CEVCOL : PRINT "Vz=";VZ;"V";
44528 LOCATE CEIROW,CEICOL : PRINT EMP32$;
44529 LOCATE CEIROW,CEICOL : PRINT "Iz=";H0;"V=";IZ;"mA";
44532 LOCATE TANROW,TANCOL : PRINT EMP38$;
44533 LOCATE TANROW,TANCOL : PRINT "Pt=";A0;"V=";PT;"mb";
44539 LOCATE PTCROW,PTCCOL : PRINT EMP38$;
44540 LOCATE PTCROW,PTCCOL : PRINT "Pc=";A4;"V=";PC;"mb";
44541 LOCATE PTZROW,PTZCOL : PRINT EMP38$;
44542 LOCATE PTZROW,PTZCOL : PRINT "Pz=";A8;"V=";PZ;"mb";
44543 IF EXPERT=1 THEN GOSUB 51000
44544 DONE=DONE+1
44545 IF DONE<TH THEN 44558 '****************************
44546 IF P$="Y" OR P$="y" THEN 44547 ELSE 44551
44547 LPRINT DATE$,TIME$
44548 LPRINT "Iz=";IZ;"Tc=";TC;"F1=";F1;"F2=";F2;"F3=";F3
44549 LPRINT "Vz=";VZ;"Pp=";PP;"Pr=";PR;"Pc=";PC;"Pz=";PZ
44550 LPRINT "P1=";P1;"P2=";P2;"P3=";P3;"Pt=";PT
44551 IF L$="Y" OR L$="y" THEN 44552 ELSE 44557
44552 OPEN "A:MOD.LOG" FOR APPEND AS #1
44553 WRITE #1,DATE$,TIME$
44554 WRITE #1,TC,VZ,F1,F2,F3,IZ,PP,PR,PC,PZ,P1,P2,P3,PT
44556 CLOSE #1
44557 DONE=0 '*******************************************
44558 LOCATE 1,1:PRINT " "
44570 FOR T=1 TO SP:NEXT
45000 KB$ = INKEY$
45020 IF KB$ = "" THEN 44400 ELSE 10500
```

```
50000 '*******************************************************
50014 '*                     PROGRAM    5                    *
50015 '*             Set up operating conditions            *
50016 '*******************************************************
50017 CLS
50017 EXPERT=1   :CLS
50018 STARTROW=2 : ENDROW=22 : STARTCOL=2 : ENDCOL=73
50020 GOSUB 30000          'Call the window routine
50022 LOCATE 1,25 :PRINT "Set Up Operating Conditions";
50023 LOCATE 3,20 :PRINT "PART A. Tc,Vz,F1,F2,F3,Iz,(S1)";
50026 LOCATE 5,4  :PRINT "Cell Temperature(K) Tcn=";TC;
50034 LOCATE 6,4  :PRINT "Cell Voltage(V) Vzn=";VZ;
50038 LOCATE 7,4  :PRINT "Feedgas Flowrate(gm/hr) F1n=";F1;
50042 LOCATE 8,4  :PRINT "Oxygen Flowrate(gm/hr) F2n=";F2;
50050 LOCATE 9,4  :PRINT "Wastegas Flowrate(gm/hr) F3n=";F3;
50054 LOCATE 10,4 :PRINT "Cell Current(mA) Izn=";IZ;
50056 LOCATE 11,4 :PRINT "Manifold Switch S1(P/R)=";S1$;
50070 LOCATE 20,4 :INPUT "Is everything correct (Y/N)";OC$
50072 IF OC$="Y" OR OC$="y" THEN 50100 ELSE 40000
50100 CLS
50102 GOSUB 30000
50104 LOCATE 3,20 :PRINT "PART B. (Pd),Pc,Pz,P1,P2,P3,Pt";
50106 LOCATE 5,4  :PRINT "Delivery Pressure(psi) Pdn=";PD;
50110 LOCATE 6,4  :PRINT "Cell Pressure(mb) Pcn=";PC;
50114 LOCATE 7,4  :PRINT "Oxygen Prsssure(mb) Pzn=";PZ;
50118 LOCATE 8,4  :PRINT "Pipeline #1 Pressure(mb) P1n=";P1;
50122 LOCATE 9,4  :PRINT "Pipeline #2 Pressure(mb) P2n=";P2;
50124 LOCATE 10,4 :PRINT "Pipeline #3 Pressure(mb) P3n=";P3;
50126 LOCATE 11,4 :PRINT "Tank Pressure(mb) Ptn=";PT;
50130 LOCATE 12,4:PRINT "Primary Bottle Pres.(psi) Pps=";PP;
50132 LOCATE 13,4:PRINT "Reserved Bottle Pres.(psi)Prs=";PR;
50150 LOCATE 19,4 :PRINT "Is everything correct (Y/N)"
50151 LOCATE 20,4 :INPUT "EXPERT SYSTEM starts if (Y)";OD$
50152 IF OD$="Y" OR OD$="y" THEN 50160 ELSE 40000
50160 CLS
50162 TCN=TC
50164 VZN=VZ
50166 F1N=F1
50168 F2N=F2
50170 F3N=F3
50172 IZN=IZ
50174 'S1$
50176 PDN=40!
50178 PCN=PC
50180 PZN=PZ
50182 P1N=P1
50184 P2N=P2
50186 P3N=P3
50188 PTN=PT
50190 PPS=PP
50192 PRS=PR
50195 EXPERT=1
50196 GOTO 40000
50200 '*******************************************************
```

```
50200 '*****************************************************
50202 '*      Experimental Threshold for Failure Modes     *
50204 '*****************************************************
50206 'Sensor Failure ************************************
50250 DSTCB=6.791175   'K, 5sd
50252 DSVZB=.038064    'V, 12sd
50254 DSF1B=.309896    'gm/hr, 8sd
50256 DSF2B=.003425    'gm/hr, 4sd
50258 DSF3B=.245832    'gm/hr, 8sd
50260 DSIZB=3!         'mA
50262 '(S1)             p/r
50264 'Pp               psi
50266 'Pr               psi
50268 DSPDB=3!         'psi
50270 DSPCB=5.889705   'mb, 5sd
50272 DSPZB=5.569985   'mb, 5sd
50274 DSP1B=7.80966    'mb, 6sd
50276 DSP2B=1.414806   'V, 6sd
50278 DSP3B=8.694588   'mb, 6sd
50280 DSPTB=5!         'mb
50300 PPL=100!         'psig
50302 PRL=100!         'psig
50390 '*****************************************************
50400 'Failure Mode G01 > Tc Too High (4F)*****************
50402 G01TCH=TCN+100!
50404 G01IZH=IZN+12.5191
50406 G01VZL=VZN-.1218
50408 'G01F2H=F2N
50410 'Failure Mode G02 > Tc Too Low (4F)******************
50412 G02TCL=TCN-200!
50414 G02IZL=IZN-31.828
50416 G02VZH=VZN+.315
50418 'G02F2L=F2N-0.004786
50420 'Failure Mode G03 > Vz Too High (4F)*****************
50422 G03VZH=VZN+.5
50424 G03IZH=IZN+67.22
50426 G03F2H=F2N+.0066
50428 G03PZH=PZN+6.6827
50430 'Failure Mode G04 > Vz Too Low (4F)******************
50432 G04VZL=VZN-.5
50434 G04IZL=IZN-66.18
50436 G04F2L=F2N-.0118
50438 G04PZL=PZN-13.3972
50440 'Failure Mode G05 > Vz Unstable (N/A) ***************
50450 'Failure Mode G06 > Vz Clip Oxidized or Bad Connection
50452 G06TCN=TCN
50454 G06VZH=VZN+DSVZB
50456 G06IZL=IZN-28.33
50458 G06PZL=PZN-6.19
50460 'Failure Mode G07 > Cell Degraded (4F)***************
50462 G07TCN=TCN
50464 G07VZH=VZN+DSVZB
50466 G07IZL=IZN-28.33
50468 G07PZL=PZN-6.19
```

```
50470 'Failure Mode G08 > Vz Short Circuit (4F)*************
50472 G08TCN=TCN
50474 G08VZL=VZN-DSVZB
50476 G08IZH=IZN+50!
50478 G08F2L=F2N-DSF2B
50480 'Failure Mode G09 > Manifold Switch Failure (4F)******
50482 G09PDL=PDN-5!
50484 'S1$="P"
50486 G09PPL=PPL
50488 G09PRL=PRL
50490 'Failure Mode G10 > Pr Too Low (4F)*******************
50492 G10PDL=PDN-5!
50494 'S1$="R"
50496 G10PPL=PPL
50498 G10PRL=PRL
50500 'Failure Mode G11 > Pd Too High (2F)******************
50502 G11PPH=1000!
50504 G11PDH=PDN+DSPDB
50510 'Failure Mode G12 > Pd Too Low (2F)*******************
50512 G12PPL=100!
50514 G12PDL=PDN-DSPDB
50520 'Failure Mode G13 > F1 Too High (4F)******************
50522 G13F1H=F1N+1!
50524 G13P1H=P1N+19.4507
50526 G13P3H=P3N+8.982701
50528 G13F3H=F3N+1.0256
50530 'Failure Mode G14 > F1 Too Low (4F)*******************
50532 G14F1L=F1N-1!
50534 G14P1L=P1N-21.8718
50536 G14P3L=P3N-10.6678
50538 G14F3L=F3N-1.0152
50540 'Failure Mode G15 > Vw Set-point High (4F)************
50542 G15F1N=F1N
50544 G15P1H=P1N+8!
50546 G15PCH=130!
50548 G15F3N=F3N
50550 'Failure Mode G16 > Vw Set-point Low (4F)*************
50552 G16F1N=F1N
50554 G16P1L=P1N-8!
50556 G16PCL=PCN-9.4599
50558 G16F3N=F3N
50560 'Failure Mode G17 > Vo Set-point High (4F)************
50562 G17PZH=PZN+6.6827
50563 G17P2H=P2N+DSP2B
50564 G17F2H=F2N+.0066
50568 G17PCN=PCN
50570 'Failure Mode G18 > Vo Set-point Low (4F)*************
50572 G18PZL=PZN-6.6827
50573 G18P2L=P2N-DSP2B
50574 G18F2L=F2N-.0066
50578 G18PCN=PCN
```

```
50580 'Failure Mode G19 > Pump Degraded (4F)******************
50582 G19F1N=F1N
50584 G19PCH=PCN+DSPCB
50586 G19PZH=PZN+DSPZB
50588 G19F3N=F3N
50590 'Failure Mode G20 > PST-10 Failure ********************
50592 'Only P1,P2,P3 are out of normal range
50600 'Failure Mode G21 > PST-15 Failure ********************
50602 'Only F1,F2,F3,PC,PZ are out of normal range
50610 'Failure Mode G22 > FMs Ground Error *******************
50612 'Only F1,F2,F3 are out of normal range
50800 'Leakage Failure L1 (4F)********************************
50802 L01F1N=F1N
50804 L01P1H=P1N+30!
50806 L01P3H=P3N+11.912
50808 L01F3H=F3N+.8249
50999 GOTO 40000
```

```
51000 '*****************************************************
51001 '*                    PROGRAM   6                     *
51002 '*     Rules for F.D.I. (using Pattern Matching)     *
51004 '*****************************************************
51010 G01%=0:G02%=0:G03%=0:G04%=0:G05%=0:G06%=0:G07%=0
51011 G08%=0:G09%=0:G10%=0:G11%=0:G12%=0:G13%=0:G14%=0
51012 G15%=0:G16%=0:G17%=0:G18%=0:G19%=0:G20%=0:G21%=0
51013 G22%=0
51014 SF%=0:FM$=""
51015 EMP54$="
51016 L1%=0
51018 'Sensor Failure ***************************************
51020 IF ABS(TC-TCN)>DSTCB THEN SF%=SF%+1:FM$="T/C Failure"
51022 IF ABS(VZ-VZN)>DSVZB THEN SF%=SF%+1:FM$="VMz Failure"
51024 IF ABS(F1-F1N)>DSF1B THEN SF%=SF%+1:FM$="FM1 Failure"
51026 IF ABS(F2-F2N)>DSF2B THEN SF%=SF%+1:FM$="FM2 Failure"
51028 IF ABS(F3-F3N)>DSF3B THEN SF%=SF%+1:FM$="FM3 Failure"
51030 IF ABS(IZ-IZN)>DSIZB THEN SF%=SF%+1:FM$="IMz Failure"
51032 '(S1)
51034 IF ABS(PP-PPN)>DSPPB THEN SF%=SF%+1:FM$="PTp Failure"
51036 IF ABS(PR-PRN)>DSPRB THEN SF%=SF%+1:FM$="PTr Failure"
51038 '(Pd)
51040 IF ABS(PC-PCN)>DSPCB THEN SF%=SF%+1:FM$="PTc Failure"
51042 IF ABS(PZ-PZN)>DSPZB THEN SF%=SF%+1:FM$="PTz Failure"
51044 IF ABS(P1-P1N)>DSP1B THEN SF%=SF%+1:FM$="PT1 Failure"
51046 IF ABS(P2-P2N)>DSP2B THEN SF%=SF%+1:FM$="PT2 Failure"
51048 IF ABS(P3-P3N)>DSP3B THEN SF%=SF%+1:FM$="PT3 Failure"
51050 '(Pt)
51060 '
51070 IF SF%=0 THEN 51072 ELSE 51082
51072 CALL DIGWRITE'("D31",1.0)
51074 LOCATE 23,10 : PRINT EMP54$
51076 RETURN
51082 IF SF%=1 THEN 51084 ELSE 51100
51084 CALL DIGWRITE'("D31",0.0)
51085 LOCATE 23,10 : PRINT EMP54$
51086 LOCATE 23,10 : PRINT FM$
51088 RETURN
51099 'General Failure ************************************
51100 'Failure Mode G01 > Tc Too High (4F)*****************
51102 IF TC>G01TCH THEN G01%=G01%+1
51104 IF VZ<G01VZL THEN G01%=G01%+1
51106 IF IZ>G01IZH THEN G01%=G01%+1
51108 IF F2>G01F2H THEN G01%=G01%+1
51150 'Failure Mode G02 > Tc Too Low (4F)******************
51152 IF TC<G02TCL THEN G02%=G02%+1
51154 IF VZ>G02VZH THEN G02%=G02%+1
51156 IF IZ<G02IZL THEN G02%=G02%+1
51158 IF F2<G02F2L THEN G02%=G02%+1
51200 'Failure Mode G03 > Vz Too High (4F)*****************
51202 IF VZ>G03VZH THEN G03%=G03%+1
51204 IF IZ>G03IZH THEN G03%=G03%+1
51206 IF F2>G03F2H THEN G03%=G03%+1
51208 IF PZ>G03PZH THEN G03%=G03%+1
```

```
51250 'Failure Mode G04 > Vz Too Low (4F)*******************
51252 IF VZ<G04VZL THEN G04%=G04%+1
51254 IF IZ<G04IZL THEN G04%=G04%+1
51256 IF F2<G04F2L THEN G04%=G04%+1
51258 IF PZ<G04PZL THEN G04%=G04%+1
51300 'Failure Mode G06 > Vz Clip Oxidized or Bad Connection
51302 IF ABS(TC-TCN)<=DSTCB THEN G06%=G06%+1
51304 IF IZ<G06IZL THEN G06%=G06%+1
51306 IF VZ>G06VZH THEN G06%=G06%+1
51308 IF PZ<G06PZL THEN G06%=G06%+1
51330 'Failure Mode G07 > Cell Degraded (4F)****************
51332 'same as G06
51350 'Failure Mode G08 > Vz Short Circuit (4F)*************
51352 IF ABS(TC-TCN)<=DSTCB THEN G08%=G08%+1
51354 IF VZ<G08VZL THEN G08%=G08%+1
51356 IF IZ>G08IZH THEN G08%=G08%+1
51358 IF F2<G08F2L THEN G08%=G08%+1
51400 'Failure Mode G09 > Manifold Switch Failure (4F)******
51402 IF PD<G09PDL THEN G09%=G09%+1
51404 IF S1$="P"   THEN G09%=G09%+1
51406 IF PP<G09PPL THEN G09%=G09%+1
51408 IF PR>G09PRL THEN G09%=G09%+1
51450 'Failure Mode G10 > Pr Too Low (4F)*******************
51452 IF PD<G10PDL THEN G10%=G10%+1
51454 IF S1$="R"   THEN G10%=G10%+1
51456 IF PP<G10PPL THEN G10%=G10%+1
51458 IF PR<G10PRL THEN G10%=G10%+1
51500 'Failure Mode G11 > Pd Too High (2F)******************
51502 IF PD>G11PDH THEN G11%=G11%+1
51504 IF PP<G11PPH THEN G11%=G11%+1
51550 'Failure Mode G12 > Pd Too Low (2F)*******************
51552 IF PD<G12PDL THEN G12%=G12%+1
51554 IF PP>G12PPL THEN G12%=G12%+1
51600 'Failure Mode G13 > F1 Too High (4F)******************
51602 IF F1>G13F1H THEN G13%=G13%+1
51604 IF P1>G13P1H THEN G13%=G13%+1
51606 IF P3>G13P3H THEN G13%=G13%+1
51608 IF F3>G13F3H THEN G13%=G13%+1
51650 'Failure Mode G14 > F1 Too Low (4F)*******************
51652 IF F1<G14F1L THEN G14%=G14%+1
51654 IF P1<G14P1L THEN G14%=G14%+1
51656 IF P3<G14P3L THEN G14%=G14%+1
51658 IF F3<G14F3L THEN G14%=G14%+1
51700 'Failure Mode G15 > Vw Set-point High (4F)************
51702 IF ABS(F1-F1N)<=DSF1B THEN G15%=G15%+1
51704 IF P1>G15P1H THEN G15%=G15%+1
51706 IF PC>G15PCH THEN G15%=G15%+1
51708 IF ABS(F3-F3N)<=DSF3B THEN G15%=G15%+1
51750 'Failure Mode G16 > Vw Set-point Low (4F)*************
51752 IF ABS(F1-F1N)<=DSF1B THEN G16%=G16%+1
51754 IF P1<G16P1L THEN G16%=G16%+1
51756 IF PC<G16PCL THEN G16%=G16%+1
51758 IF ABS(F3-F3N)<=DSF3B THEN G16%=G16%+1
```

```
51800 'Failure Mode G17 > Vo Set-point High (4F)************
51802 IF F2<G17F2H THEN G17%=G17%+1
51806 IF ABS(PC-PCN)<=DSPCB THEN G17%=G17%+1
51808 IF PZ>G17PZH THEN G17%=G17%+1
51810 IF P2>G17P2H THEN G17%=G17%+1
51850 'Failure Mode G18 > Vo Set-point Low (4F)*************
51852 IF F2>G18F2L THEN G18%=G18%+1
51856 IF ABS(PC-PCN)<=DSPCB THEN G18%=G18%+1
51858 IF PZ<G18PZL THEN G18%=G18%+1
51860 IF P2<G18P2L THEN G18%=G18%+1
51900 'Failure Mode G19 > Pump Degraded (4F)***************
51902 IF ABS(F1-F1N)<=DSF1B THEN G19%=G19%+1
51904 IF ABS(F3-F3N)<=DSF3B THEN G19%=G19%+1
51906 IF PC>G19PCH THEN G19%=G19%+1
51908 IF PZ>G19PZH THEN G19%=G19%+1
51950 'Failure Mode G20 > PST-10 Failed (6F)***************
51952 IF ABS(P1-P1N)>DSP1B THEN G20%=G20%+1
51954 IF ABS(P2-P2N)>DSP2B THEN G20%=G20%+1
51956 IF ABS(P3-P3N)>DSP3B THEN G20%=G20%+1
51958 IF ABS(F1-F1N)<=DSF1B THEN G20%=G20%+1
51960 IF ABS(F3-F3N)<=DSF3B THEN G20%=G20%+1
51962 IF ABS(PC-PCN)<=DSPCB THEN G20%=G20%+1
52000 'Failure Mode G21 > PST-B15 Failured (3F)*************
52002 IF (F1<.1) OR (F1>5!) THEN G21%=G21%+1
52004 IF (F2<.1) OR (F2>5!) THEN G21%=G21%+1
52006 IF (F3<.1) OR (F3>5!) THEN G21%=G21%+1
52050 'Failure Mode G22 > FMs Ground Error ****************
52052 'same as G21
52500 'Leakage Failure L1 (4F)*****************************
52502 IF ABS(F1-F1N)<=DSF1B THEN L01%=L01%+1
52504 IF P1>L01P1H THEN L01%=L01%+1
52506 IF P3>L01P3H THEN L01%=L01%+1
52508 IF F3>L01F3H THEN L01%=L01%+1
53000 '****************************************************
53010 IF G01%=4 THEN 53012 ELSE 53020
53012 FM$="Tc Too High" : GOTO 54000
53020 IF G02%=4 THEN 53022 ELSE 53030
53022 FM$="Tc Too Low" : GOTO 54000
53030 IF G03%=4 THEN 53032 ELSE 53040
53032 FM$="Vz Too High" : GOTO 54000
53040 IF G04%=4 THEN 53042 ELSE 53060
53042 FM$="Vz Too Low" : GOTO 54000
53060 IF G06%=4 THEN 53062 ELSE 53080
53062 FM$="Vz Clip Oxidized, Bad Conn. OR Cell Degraded"
53064 GOTO 54000
53080 IF G08%=4 THEN 53082 ELSE 53090
53082 FM$="Vz Short Circuit" : GOTO 54000
53090 IF G09%=4 THEN 53092 ELSE 53100
53092 FM$="Manifold Switch Failure" : GOTO 54000
53100 IF G10%=4 THEN 53102 ELSE 53110
53102 FM$="Pr Too Low" : GOTO 54000
53110 IF G11%=2 THEN 53112 ELSE 53120
53112 FM$="Pd Too High" : GOTO 54000
53120 IF G12%=2 THEN 53122 ELSE 53130
```

```
53122 FM$="Pd Too Low" : GOTO 54000
53130 IF G13%=4 THEN 53132 ELSE 53140
53132 FM$="F1 Too High" : GOTO 54000
53140 IF G14%=4 THEN 53142 ELSE 53150
53142 FM$="F1 Too Low" : GOTO 54000
53150 IF G15%=4 THEN 53152 ELSE 53160
53152 FM$="Vw Set-point High" : GOTO 54000
53160 IF G16%=4 THEN 53162 ELSE 53170
53162 FM$="Vw Set-point Low" : GOTO 54000
53170 IF G17%=4 THEN 53172 ELSE 53180
53172 FM$="Vo Set-point High" : GOTO 54000
53180 IF G18%=4 THEN 53182 ELSE 53190
53182 FM$="Vo Set-point Low" : GOTO 54000
53190 IF G19%=4 THEN 53192 ELSE 53200
53192 FM$="Pump Degraded" : GOTO 54000
53200 IF G20%=6 THEN 53202 ELSE 53210
53202 FM$="PST-10 Failed" : GOTO 54000
53210 IF G21%=3 THEN 53212 ELSE 53500
53212 FM$="PST-15B Failed OR FMs Ground Error"
53214 GOTO 54000
53500 IF L01%=4 THEN 53502 ELSE 53900
53502 FM$="Leakage At Location #1" : GOTO 54000
53900 FM$="Unrecognized Failure Mode" : GOTO 54000
54000 '*************************************************
54002 CALL DIGWRITE'("D31",0.0)
54004 LOCATE 23,10 : PRINT EMP54$
54006 LOCATE 23,10 : PRINT FM$
54010 '*************************************************
58000 IF Z>TH THEN 59000
58002 ATC(Z)=TC
58004 AVZ(Z)=VZ
58006 AF1(Z)=F1
58008 AF2(Z)=F2
58010 AF3(Z)=F3
58012 AIZ(Z)=IZ
58014 'AS1(Z)=S1
58016 APP(Z)=PP
58018 APR(Z)=PR
58020 'APD(Z)=PD
58022 APC(Z)=PC
58024 APZ(Z)=PZ
58026 AP1(Z)=P1
58028 AP2(Z)=P2
58030 AP3(Z)=P3
58032 'APT(Z)=PT
58040 AFM$(Z)=FM$
58100 Z=Z+1
58200 GOTO 59200
59000 FOR V=2 TO TH
59002 U=V-1
59004 ATC(U)=ATC(V)
59006 AVZ(U)=AVZ(V)
59008 AF1(U)=AF1(V)
59010 AF2(U)=AF2(V)
```

```
59012 AF3(U)=AF3(V)
59014 AIZ(U)=AIZ(V)
59016 'AS1(U)=AS1(V)
59018 APP(U)=APP(V)
59020 APR(U)=APR(V)
59022 'APD(U)=APD(V)
59024 APC(U)=APC(V)
59026 APZ(U)=APZ(V)
59028 AP1(U)=AP1(V)
59030 AP2(U)=AP2(V)
59032 AP3(U)=AP3(V)
59034 'APT(U)=APT(V)
59036 AFM$(U)=AFM$(V)
59038 NEXT V
59040 ATC(TH)=TC
59042 AVZ(TH)=VZ
59044 AF1(TH)=F1
59046 AF2(TH)=F2
59048 AF3(TH)=F3
59050 AIZ(TH)=IZ
59052 'AS1(TH)=S1
59054 APP(TH)=PP
59056 APR(TH)=PR
59058 'APD(TH)=PD
59060 APC(TH)=PC
59062 APZ(TH)=PZ
59064 AP1(TH)=P1
59066 AP2(TH)=P2
59068 AP3(TH)=P3
59070 'APT(TH)=PT
59072 AFM$(TH)=FM$
59100 Z=TH+1
59200 RETURN
59202 'Return to EXPERT SUB.
60000 '*****************************************************
60002 '*        Analyze Individual Input Readings          *
60004 '*****************************************************
60010 CLS
60012 STARTROW=2 : ENDROW=22 : STARTCOL=2 : ENDCOL=73
60014 GOSUB 30000
60016 Z1=Z-1
60018 IF Z1=0 THEN 60236
60020 OPEN "O",#2,"A:MODTC.DAT"
60021 FOR I=1 TO Z1
60022 WRITE #2,ATC(I)
60023 NEXT I
60024 CLOSE #2
60030 OPEN "O",#2,"A:MODVZ.DAT"
60031 FOR I=1 TO Z1
60032 WRITE #2,AVZ(I)
60033 NEXT I
60034 CLOSE #2
60040 OPEN "O",#2,"A:MODF1.DAT"
60041 FOR I=1 TO Z1
```

```
60042 WRITE #2,AF1(I)
60043 NEXT I
60044 CLOSE #2
60050 OPEN "O",#2,"A:MODF2.DAT"
60051 FOR I=1 TO Z1
60052 WRITE #2,AF2(I)
60053 NEXT I
60054 CLOSE #2
60060 OPEN "O",#2,"A:MODF3.DAT"
60061 FOR I=1 TO Z1
60062 WRITE #2,AF3(I)
60063 NEXT I
60064 CLOSE #2
60070 OPEN "O",#2,"A:MODIZ.DAT"
60071 FOR I=1 TO Z1
60072 WRITE #2,AIZ(I)
60073 NEXT I
60074 CLOSE #2
60080 'S1
60090 OPEN "O",#2,"A:MODPP.DAT"
60091 FOR I=1 TO Z1
60092 WRITE #2,APP(I)
60093 NEXT I
60094 CLOSE #2
60100 OPEN "O",#2,"A:MODPR.DAT"
60101 FOR I=1 TO Z1
60102 WRITE #2,APR(I)
60103 NEXT I
60104 CLOSE #2
60110 'PD
60120 OPEN "O",#2,"A:MODPC.DAT"
60121 FOR I=1 TO Z1
60122 WRITE #2,APC(I)
60123 NEXT I
60124 CLOSE #2
60130 OPEN "O",#2,"A:MODPZ.DAT"
60131 FOR I=1 TO Z1
60132 WRITE #2,APZ(I)
60133 NEXT I
60134 CLOSE #2
60140 OPEN "O",#2,"A:MODP1.DAT"
60141 FOR I=1 TO Z1
60142 WRITE #2,AP1(I)
60143 NEXT I
60144 CLOSE #2
60150 OPEN "O",#2,"A:MODP2.DAT"
60151 FOR I=1 TO Z1
60152 WRITE #2,AP2(I)
60153 NEXT I
60154 CLOSE #2
60160 OPEN "O",#2,"A:MODP3.DAT"
60161 FOR I=1 TO Z1
60162 WRITE #2,AP3(I)
60163 NEXT I
```

```
60164 CLOSE #2
60170 'PT
60180 OPEN "O",#2,"A:MODFM.DAT"
60181 FOR I=1 TO Z1
60182 WRITE #2,AFM$(I)
60183 NEXT I
60184 CLOSE #2
60230 '*****************************************************
60236 LOCATE 1,22 : PRINT "Analyze Individual Input Reading"
60238 LOCATE 23,2 : PRINT "F10 Return To Main Menu"
60240 LOCATE 4,10 : PRINT "Analyze data in a:MOD.LOG and"
60242 LOCATE 5,10 : PRINT "a:MOD**.DAT by using Lotus 1-2-3"
60250 FOR I=1 TO 10
60252   KEY(I) OFF
60254 NEXT I
60256 KB$=INKEY$
60258 IF LEN(KB$)<>2 THEN GOTO 60284
60260 KB$=RIGHT$(KB$,1)
60280 IF KB$=F10KEY$ THEN GOTO 10500
60282 '
60284 GOTO 60256
62000 '*****************************************************
62002 '*           General System Information              *
62004 '*****************************************************
62010 CLS
62012 STARTROW=2 : ENDROW=22 : STARTCOL=2 : ENDCOL=73
62014 GOSUB 30000
62016 LOCATE 1,22 : PRINT "General System Information"
62018 LOCATE 23,2 : PRINT "F10 Return To Main Menu"
62050 FOR I=1 TO 10
62052   KEY(I) OFF
62054 NEXT I
62056 KB$=INKEY$
62058 IF LEN(KB$)<>2 THEN GOTO 62084
62060 KB$=RIGHT$(KB$,1)
62080 IF KB$=F10KEY$ THEN GOTO 10500
62082 '
62084 GOTO 62056
65000 '*****************************************************
65002 CLS
65010 SCREEN 0,0
65020 KEY ON
65100 END
```

APPENDIX B. SUBPROGRAMS OF COMPUTER PROGRAM "MODFDI3.1SC"

(MODFDI3.1SC is similar to MODFDI2.BAS for the most part except the subprograms to acquire data and send data to 1st-Class. The subprograms are listed in this appendix. Refer to Section 3.4 for description.)

```
20000 '*******************************************************
20002 '* Start checking the data from the Keith Lee system *
20004 '*******************************************************
20010 'CHANNEL H1 **********
20012 POKE &H8C,1
20013 FOR C=1 TO 50 : NEXT C
20015 POKE &H81,7
20016 POKE &H9A,0
20018 POKE &H9B,255
20020 READY=PEEK(&H9B)
20022 IF READY=127 THEN 20024 ELSE 20020
20024 DH1L=PEEK(&H80)
20026 DH1H=PEEK(&H81)
20028 DH1H=(DH1H-240)*256
20030 DH1R=DH1L+DH1H
20032 Q1=DH1R*(20/4095)-10
20040 'CHANNEL A1 **********
20042 POKE &H86,1
20044 POKE &H81,4
20046 POKE &H9A,0
20048 POKE &H9B,255
20050 DLOW=PEEK(&H80)
20052 DHIGH=PEEK(&H81)
20054 DHIGH=(DHIGH-240)*256
20056 DRES=DLOW+DHIGH
20058 A1=DRES*(20/4095)-10
20070 'CHANNEL A2 **********
20072 POKE &H86,2
20074 POKE &H81,4
20076 POKE &H9A,0
20078 POKE &H9B,255
20080 DLOW=PEEK(&H80)
20082 DHIGH=PEEK(&H81)
20084 DHIGH=(DHIGH-240)*256
20086 DRES=DLOW+DHIGH
20088 A2=DRES*(20/4095)-10
20100 'CHANNEL A3 **********
20102 POKE &H86,3
20104 POKE &H81,4
20106 POKE &H9A,0
20108 POKE &H9B,255
20110 DLOW=PEEK(&H80)
20112 DHIGH=PEEK(&H81)
20114 DHIGH=(DHIGH-240)*256
20116 DRES=DLOW+DHIGH
20118 A3=DRES*(20/4095)-10
20130 'CHANNEL A4 **********
20132 POKE &H86,4
20134 POKE &H81,4
20136 POKE &H9A,0
20138 POKE &H9B,255
20140 DLOW=PEEK(&H80)
20142 DHIGH=PEEK(&H81)
20144 DHIGH=(DHIGH-240)*256
```

```
20146 DRES=DLOW+DHIGH
20148 A4=DRES*(20/4095)-10
20160 'CHANNEL A5 **********
20162 POKE &H86,5
20164 POKE &H81,4
20166 POKE &H9A,0
20168 POKE &H9B,255
20170 DLOW=PEEK(&H80)
20172 DHIGH=PEEK(&H81)
20174 DHIGH=(DHIGH-240)*256
20176 DRES=DLOW+DHIGH
20178 A5=DRES*(20/4095)-10
20190 'CHANNEL A6 **********
20192 POKE &H86,6
20194 POKE &H81,4
20196 POKE &H9A,0
20198 POKE &H9B,255
20200 DLOW=PEEK(&H80)
20202 DHIGH=PEEK(&H81)
20204 DHIGH=(DHIGH-240)*256
20206 DRES=DLOW+DHIGH
20208 A6=DRES*(20/4095)-10
20220 'CHANNEL A7 **********
20222 POKE &H86,7
20224 POKE &H81,4
20226 POKE &H9A,0
20228 POKE &H9B,255
20230 DLOW=PEEK(&H80)
20232 DHIGH=PEEK(&H81)
20234 DHIGH=(DHIGH-240)*256
20236 DRES=DLOW+DHIGH
20238 A7=DRES*(20/4095)-10
20250 'CHANNEL A8 **********
20252 POKE &H86,8
20254 POKE &H81,4
20256 POKE &H9A,0
20258 POKE &H9B,255
20260 DLOW=PEEK(&H80)
20262 DHIGH=PEEK(&H81)
20264 DHIGH=(DHIGH-240)*256
20266 DRES=DLOW+DHIGH
20268 A8=DRES*(20/4095)-10
20280 'CHANNEL A0 **********
20282 POKE &H86,0
20284 POKE &H81,4
20286 POKE &H9A,0
20288 POKE &H9B,255
20290 READY=PEEK(&H9B)
20292 IF READY=127 THEN 20294 ELSE 20290
20294 DA0L=PEEK(&H80)
20296 DA0H=PEEK(&H81)
20298 DA0H=(DA0H-240)*256
20300 DA0R=DA0L+DA0H
20302 A0=DA0R*(20/4095)-10
```

```
20310 'CHANNEL H0 **********
20312 POKE &H8C,0
20313 FOR C=1 TO 50: NEXT C
20314 POKE &H81,7
20316 POKE &H9A,0
20318 POKE &H9B,255
20320 READY=PEEK(&H9B)
20322 IF READY=127 THEN 20324 ELSE 20320
20324 DH0L=PEEK(&H80)
20326 DH0H=PEEK(&H81)
20327 DH0H=(DH0H-240)*256
20328 DH0R=DH0L+DH0H
20330 Q0=DH0R*(20/4095)-10
20340 'CHANNEL L0 **********
20342 POKE &H8A,0
20343 FOR C=1 TO 50 : NEXT C
20344 POKE &H81,6
20346 POKE &H9A,0
20348 POKE &H9B,255
20350 DLOW=PEEK(&H80)
20352 DHIGH=PEEK(&H81)
20354 DHIGH=(DHIGH-240)*256
20356 DRES=DLOW+DHIGH
20358 L0=DRES*(20/4095)-10
20360 L0=(L0/100)*1000
20370 'CHANNEL L1 **********
20372 POKE &H8A,1
20373 FOR C=1 TO 50 : NEXT C
20374 POKE &H81,6
20376 POKE &H9A,0
20378 POKE &H9B,255
20380 DLOW=PEEK(&H80)
20382 DHIGH=PEEK(&H81)
20384 DHIGH=(DHIGH-240)*256
20386 DRES=DLOW+DHIGH
20388 L1=DRES*(20/4095)-10
20390 L1=(L1/100)*1000
20400 'CHANNEL CJ **********
20402 POKE &H82,32
20403 FOR C=1 TO 50 : NEXT C
20404 POKE &H81,2
20406 POKE &H9A,0
20408 POKE &H9B,255
20410 DLOW=PEEK(&H80)
20412 DHIGH=PEEK(&H81)
20414 DHIGH=(DHIGH-240)*256
20416 DRES=DLOW+DHIGH
20418 CJ=DRES*(20/4095)-10
20420 CJ=CJ*10
20430 'CHANNEL C0 **********
20432 POKE &H82,0
20433 FOR C= 1 TO 50 :NEXT C
20434 POKE &H81,2
20436 POKE &H9A,1
```

```
20438 POKE &H9B,255
20440 DLOW=PEEK(&H80)
20442 DHIGH=PEEK(&H81)
20444 DHIGH=(DHIGH-240)*256
20446 DRES=DLOW+DHIGH
20448 C0=DRES*(20/4095)-10
20450 C0=(C0/200!)
20460 TC=K0+C0*(K1+C0*(K2+C0*(K3+C0*(K4+C0*(K5+C0*(K6+C0*
20462 TC=TC+CJ                                    (K7+C0*K8)))
20490 'CHANNEL C1 **********
20492 POKE &H82,1
20493 FOR C= 1 TO 50 :NEXT C
20494 POKE &H81,2
20496 POKE &H9A,1
20498 POKE &H9B,255
20500 DLOW=PEEK(&H80)
20502 DHIGH=PEEK(&H81)
20504 DHIGH=(DHIGH-240)*256
20506 DRES=DLOW+DHIGH
20508 C1=DRES*(20/4095)-10
20510 C1=(C1/200!)
20520 TA=J0+C1*(J1+C1*(J2+C1*(J3+C1*(J4+C1*J5))))
20522 TA=TA+CJ
20550 'CHANNEL C2 **********
20999 '*************************************************
21000 TC=TC+273
21001 VZ=Q1
21002 F1=A1*8.057
21004 F2=A2*.16
21006 F3=A3*8.057
21007 IZ=Q0
21008 S1$="P"
21010 PP=(L0/30!)*1000!
21012 PR=(L1/30!)*1000!
21014 PD=40!
21016 PC=(A4*10)*1.3332
21018 PZ=(A8*10)*1.3332
21020 P1=(A5*(-163.1278)+186.2344)+PA
21022 P2=(A7*(-170.1535)+230.4117)+PA
21024 P3=(A6*(-165.0467)+203.1309)+PA
21025 PT=(A0*(0))
21026 TA=TA
21030 '
21100 GOSUB 22000
21999 RETURN
22000 'CHANNEL V0 **********
22002 POKE &H9D,64
22004 D=INT(V0/.004882)+2048!
22006 DHIGH=INT(D/256!)
22008 DLOW=D-(DHIGH*256!)
22016 POKE &H88,0:POKE &H89,DLOW
22018 POKE &H88,1:POKE &H89,DHIGH
22020 POKE &H9D,1
22030 RETURN
```

```
22060 'CHANNEL D31 ON ******
22062 POKE &H8E,3
22064 POKE &H8F,128
22075 RETURN
22080 'CHANNEL D31 OFF *****
22082 POKE &H8E,3
22084 POKE &H8F,0
22095 RETURN
22100 '*******************************************************
```

```
51000 '****************************************************
51002 '*         Send data to "1st CLASS"                 *
51004 '****************************************************
51100 '****************************************************
51102 IF ABS(TC-TCN)<=DSTCB THEN RTC$="N" : GOTO 51112
51104 IF (TC-TCN)>DSTCB THEN RTC$="H"     : GOTO 51112
51106 IF (TCN-TC)>DSTCB THEN RTC$="L"     : GOTO 51112
51110 '****************************************************
51112 IF ABS(VZ-VZN)<=DSVZB THEN RVZ$="N" : GOTO 51122
51114 IF (VZ-VZN)>DSVZB THEN RVZ$="H"     : GOTO 51122
51116 IF (VZN-VZ)>DSVZB THEN RVZ$="L"     : GOTO 51122
51120 '****************************************************
51122 IF ABS(F1-F1N)<=DSF1B THEN RF1$="N" : GOTO 51132
51124 IF (F1-F1N)>DSF1B THEN RF1$="H"     : GOTO 51132
51126 IF (F1N-F1)>DSF1B THEN RF1$="L"     : GOTO 51132
51130 '****************************************************
51132 IF ABS(F2-F2N)<=DSF2B THEN RF2$="N" : GOTO 51142
51134 IF (F2-F2N)>DSF2B THEN RF2$="H"     : GOTO 51142
51136 IF (F2N-F2)>DSF2B THEN RF2$="L"     : GOTO 51142
51140 '****************************************************
51142 IF ABS(F3-F3N)<=DSF3B THEN RF3$="N" : GOTO 51152
51144 IF (F3-F3N)>DSF3B THEN RF3$="H"     : GOTO 51152
51146 IF (F3N-F3)>DSF3B THEN RF3$="L"     : GOTO 51152
51150 '****************************************************
51152 IF ABS(IZ-IZN)<=DSIZB THEN RIZ$="N" : GOTO 51162
51154 IF (IZ-IZN)>DSIZB THEN RIZ$="H"     : GOTO 51162
51156 IF (IZN-IZ)>DSIZB THEN RIZ$="L"     : GOTO 51162
51160 '****************************************************
51162 'S1$
51170 '****************************************************
51172 IF PP>1000 THEN RPP$="H"                          : GOTO 51182
51174 IF (PP<1000) AND (PP>100) THEN RPP$="N"  : GOTO 51182
51176 IF PP<100 THEN RPP$="L"                            : GOTO 51182
51180 '****************************************************
51182 IF PR>1000 THEN RPR$="H"                          : GOTO 51192
51184 IF (PR<1000) AND (PR>100) THEN RPR$="N"  : GOTO 51192
51186 IF PR<100 THEN RPR$="L"                            : GOTO 51192
51190 '****************************************************
51192 RPD$="N"
51200 '****************************************************
51202 IF ABS(PC-PCN)<=DSPCB THEN RPC$="N" : GOTO 51212
51204 IF (PC-PCN)>DSPCB THEN RPC$="H"     : GOTO 51212
51206 IF (PCN-PC)>DSPCB THEN RPC$="L"     : GOTO 51212
51210 '****************************************************
51212 IF ABS(PZ-PZN)<=DSPZB THEN RPZ$="N" : GOTO 51222
51214 IF (PZ-PZN)>DSPZB THEN RPZ$="H"     : GOTO 51222
51216 IF (PZN-PZ)>DSPZB THEN RPZ$="L"     : GOTO 51222
51220 '****************************************************
51222 IF ABS(P1-P1N)<=DSP1B THEN RP1$="N" : GOTO 51232
51224 IF (P1-P1N)>DSP1B THEN RP1$="H"     : GOTO 51232
51226 IF (P1N-P1)>DSP1B THEN RP1$="L"     : GOTO 51232
51230 '****************************************************
51232 IF ABS(P2-P2N)<=DSP2B THEN RP2$="N" : GOTO 51242
51234 IF (P2-P2N)>DSP2B THEN RP2$="H"     : GOTO 51242
```

```
51000 '**************************************************
51002 '*          Send data to "1st CLASS"              *
51004 '**************************************************
51100 '**************************************************
51102 IF ABS(TC-TCN)<=DSTCB THEN RTC$="N" : GOTO 51112
51104 IF (TC-TCN)>DSTCB THEN RTC$="H"     : GOTO 51112
51106 IF (TCN-TC)>DSTCB THEN RTC$="L"     : GOTO 51112
51110 '**************************************************
51112 IF ABS(VZ-VZN)<=DSVZB THEN RVZ$="N" : GOTO 51122
51114 IF (VZ-VZN)>DSVZB THEN RVZ$="H"     : GOTO 51122
51116 IF (VZN-VZ)>DSVZB THEN RVZ$="L"     : GOTO 51122
51120 '**************************************************
51122 IF ABS(F1-F1N)<=DSF1B THEN RF1$="N" : GOTO 51132
51124 IF (F1-F1N)>DSF1B THEN RF1$="H"     : GOTO 51132
51126 IF (F1N-F1)>DSF1B THEN RF1$="L"     : GOTO 51132
51130 '**************************************************
51132 IF ABS(F2-F2N)<=DSF2B THEN RF2$="N" : GOTO 51142
51134 IF (F2-F2N)>DSF2B THEN RF2$="H"     : GOTO 51142
51136 IF (F2N-F2)>DSF2B THEN RF2$="L"     : GOTO 51142
51140 '**************************************************
51142 IF ABS(F3-F3N)<=DSF3B THEN RF3$="N" : GOTO 51152
51144 IF (F3-F3N)>DSF3B THEN RF3$="H"     : GOTO 51152
51146 IF (F3N-F3)>DSF3B THEN RF3$="L"     : GOTO 51152
51150 '**************************************************
51152 IF ABS(IZ-IZN)<=DSIZB THEN RIZ$="N" : GOTO 51162
51154 IF (IZ-IZN)>DSIZB THEN RIZ$="H"     : GOTO 51162
51156 IF (IZN-IZ)>DSIZB THEN RIZ$="L"     : GOTO 51162
51160 '**************************************************
51162 'S1$
51170 '**************************************************
51172 IF PP>1000 THEN RPP$="H"                     : GOTO 51182
51174 IF (PP<1000) AND (PP>100) THEN RPP$="N" : GOTO 51182
51176 IF PP<100 THEN RPP$="L"                      : GOTO 51182
51180 '**************************************************
51182 IF PR>1000 THEN RPR$="H"                     : GOTO 51192
51184 IF (PR<1000) AND (PR>100) THEN RPR$="N" : GOTO 51192
51186 IF PR<100 THEN RPR$="L"                      : GOTO 51192
51190 '**************************************************
51192 RPD$="N"
51200 '**************************************************
51202 IF ABS(PC-PCN)<=DSPCB THEN RPC$="N" : GOTO 51212
51204 IF (PC-PCN)>DSPCB THEN RPC$="H"     : GOTO 51212
51206 IF (PCN-PC)>DSPCB THEN RPC$="L"     : GOTO 51212
51210 '**************************************************
51212 IF ABS(PZ-PZN)<=DSPZB THEN RPZ$="N" : GOTO 51222
51214 IF (PZ-PZN)>DSPZB THEN RPZ$="H"     : GOTO 51222
51216 IF (PZN-PZ)>DSPZB THEN RPZ$="L"     : GOTO 51222
51220 '**************************************************
51222 IF ABS(P1-P1N)<=DSP1B THEN RP1$="N" : GOTO 51232
51224 IF (P1-P1N)>DSP1B THEN RP1$="H"     : GOTO 51232
51226 IF (P1N-P1)>DSP1B THEN RP1$="L"     : GOTO 51232
51230 '**************************************************
51232 IF ABS(P2-P2N)<=DSP2B THEN RP2$="N" : GOTO 51242
51234 IF (P2-P2N)>DSP2B THEN RP2$="H"     : GOTO 51242
```

```
51236 IF (P2N-P2)>DSP2B THEN RP2$="L"      : GOTO 51242
51240 '*****************************************************
51242 IF ABS(P3-P3N)<=DSP3B THEN RP3$="N" : GOTO 51252
51244 IF (P3-P3N)>DSP3B THEN RP3$="H"      : GOTO 51252
51246 IF (P3N-P3)>DSP3B THEN RP3$="L"      : GOTO 51252
51250 '*****************************************************
51252 RPT$="N"
51260 '*****************************************************
51810 OPEN "O",#2,"MOD1SC3.ANS"
51815 WRITE #2,RTC$
51820 WRITE #2,RVZ$
51825 WRITE #2,RF1$
51830 WRITE #2,RF2$
51835 WRITE #2,RF3$
51840 WRITE #2,RIZ$
51845 WRITE #2,S1$
51850 WRITE #2,RPP$
51855 WRITE #2,RPR$
51860 WRITE #2,RPD$
51865 WRITE #2,RPC$
51870 WRITE #2,RPZ$
51875 WRITE #2,RP1$
51880 WRITE #2,RP2$
51885 WRITE #2,RP3$
51890 WRITE #2,RPT$
51900 CLOSE #2
51905 SHELL "ADVISOR.EXE MOD1SC3/IMRX"
51908 LOCATE 23,10 : PRINT EMP38$;
51910 OPEN "I",#3,"MOD1SC3.RPT"
51912 FOR R=1 TO 10
51913 INPUT #3,E$(R)
51914 NEXT R
51915 INPUT #3,Z$
51920 CLOSE #3
51925 LOCATE 23,10 : PRINT Z$;
51930 GOTO 44544
51940 '*****************************************************
```

APPENDIX C. KNOWLEDGE BASE FILE "MOD1SC3.KBM"
(This is the file that the 1st-Class Advisor program needs in order to diagnose failures. Refer to Section 3.4 for description.)

```
new_Example,  Replicate,  Change,  Activate,  Move,  Delete
            Files  Definitions  Examples  Methods  Rule  Advisor
[F1=Help]       22 Examples in MOD1SC3          [F9=Definitions] [F10=Methods]
        (inactive)                                                weights--:
```

		MEMO	TC	VZ	F1	F2	F3
>	1:	NORMAL	N	N	N	N	N
	2:	Tc HIGH	H	L	N	*	N
	3:	Tc LOW	L	H	N	*	N
	4:	Vz HIGH	N	H	N	H	N
	5:	Vz LOW	N	L	N	L	N
	6:	G06,G07	N	H	N	L	N
	7:	Short Circu	N	L	N	L	N
	8:	M-SWITCH F	N	N	N	N	N
	9:	Pr LOW	N	N	N	N	N
	10:	Pd HIGH	N	N	N	N	N
	11:	Pd LOW	N	N	N	N	N
	12:	F1 HIGH	N	*	H	*	H
	13:	F1 LOW	N	*	L	*	L
	14:	Vw HIGH	N	N	N	*	N
	15:	Vw LOW	N	N	N	*	N
	16:	Vo HIGH	N	N	N	*	N
	17:	Vo LOW	N	N	N	*	N
	18:	Pump Degrad	N	*	N	*	N
	19:	PST10 F(H)	N	N	N	N	N
	20:	PST10 F(L)	N	N	N	N	N
	21:	Ground Erro	N	*	O	O	O
	22:	Leakage #1	N	*	N	H	H

```
new_Example,  Replicate,  Change,  Activate,  Move,  Delete
           Files  Definitions  Examples  Methods  Rule  Advisor
[F1=Help]       22 Examples in MOD1SC3        [F9=Definitions] [F10=Methods]
                                                                  weights-->
          IZ          S1          PP          PR          PD          PC
>   1:    N           P           N           N           N           N
    2:    H           P           N           N           N           N
    3:    L           P           N           N           N           N
    4:    H           P           N           N           N           N
    5:    L           P           N           N           N           N
    6:    L           P           N           N           N           N
    7:    H           P           N           N           N           N
    8:    N           P           L           N           L           N
    9:    N           P           N           L           N           N
   10:    N           P           N           N           H           N
   11:    N           P           N           N           L           N
   12:    *           P           N           N           N           H
   13:    *           P           N           N           N           L
   14:    *           P           N           N           N           H
   15:    *           P           N           N           N           L
   16:    *           P           N           N           N           N
   17:    *           P           N           N           N           N
   18:    *           P           N           N           N           H
   19:    N           P           *           *           N           N
   20:    N           P           *           *           N           N
   21:    *           P           N           N           N           *
   22:    H           P           N           N           N           H
```

new_Example, Replicate, Change, Activate, Move, Delete
Files Definitions Examples Methods Rule Advisor
[F1=Help] 22 Examples in MOD1SC3 [F9=Definitions] [F10=Methods]
weights-->

	PZ	P1	P2	P3	PT	RESULT
> 1:	N	N	N	N	N	NORMAL
2:	*	N	*	N	N	G01
3:	*	N	*	N	N	G02
4:	H	N	H	N	N	G03
5:	L	N	L	N	N	G04
6:	L	N	*	N	N	G67
7:	L	N	*	N	N	G08
8:	N	N	N	N	N	G09
9:	N	N	N	N	N	G10
10:	N	N	N	N	N	G11
11:	N	N	N	N	N	G12
12:	*	H	*	H	N	G13
13:	*	L	*	L	N	G14
14:	*	H	*	H	N	G15
15:	*	L	*	L	N	G16
16:	H	N	H	N	N	G17
17:	L	N	L	N	N	G18
18:	H	H	H	H	H	G19
19:	N	H	H	H	*	G20H
20:	N	L	L	L	*	G20L
21:	*	*	*	*	*	G21
22:	H	H	H	H	N	L01

APPENDIX D. COMPUTER PROGRAM "MODFDI4.PRO"

(Refer to Section 3.5 for description.)

```
/*      MODFDI4.PRO
        expert system for failure control
        in Mars oxygen production system */

nowarnings
domains

    n = symbol

database

    current(n, n) /* to store input data */

predicates

    readdatal /* to read part of input data */
    readdata2 /* to read part of input data */
    input     /* call readdatal & readdata2 */
    work      /* fire suitable rules        */
    rule(n, n, n, n, n, n, n, n, n,
         n, n, n, n, n, n, n)
    start     /* internal defined goal */

goal

    clearwindow, /* clear dialog window */
    start.       /* initial execution */

clauses

    start:-
        retract(current(X,Y)), fail.
        /* kill previous data */

    start:-
        input, work.
        /* call input data & fire suitable rules */

    input:-
        write("Input Operating Conditions"),
        nl,
        write("Part A. tc, vz, fl, f2, f3, iz, sl, pp, pr"),
        nl,
        readdatal,
        write("Part B. (pd), pc, pz, pl, p2, p3, pt"),
        nl,
        readdata2.


    readdatal:-
        write("Cell Temperature (K) Tc="),
        readln(Vl),
        assertz(current("tc", Vl)),
        write("Cell Voltage (V) Vz="),
        readln(V2),
        assertz(current("vz", V2)),
        write("Feedgas Flowrate (gm/hr) Fl="),
        readln(V3),
```

```
    assertz(current("f1", V3)),
    write("Oxygen Flowrate (gm/hr) F2="),
    readln(V4),
    assertz(current("f2", V4)),
    write("Wastegas Flowrate (gm/hr) F3="),
    readln(V5),
    assertz(current("f3", V5)),
    write("Cell Current (mA) Iz="),
    readln(V6),
    assertz(current("iz", V6)),
    write("Manifold Switch S1="),
    readln(V7),
    assertz(current("s1", V7)),
    write("Primary Bottle Presure Pp="),
    readln(V8),
    assertz(current("pp", V8)),
    write("Reserved Bottle Presure Pr="),
    readln(V9),
    assertz(current("pr", V9)).

readdata2:-
    write("Delivery Pressure (psi) Pd="),
    readln(V1),
    assertz(current("pd", V1)),
    write("Cell pressure (mb) Pc="),
    readln(V2),
    assertz(current("pc", V2)),
    write("Oxygen Pressure (mb) Pz="),
    readln(V3),
    assertz(current("pz", V3)),
    write("Pipeline #1 Pressure (mb) P1="),
    readln(V4),
    assertz(current("p1", V4)),
    write("Pipeline #2 Pressure (mb) P2="),
    readln(V5),
    assertz(current("p2", V5)),
    write("Pipeline #3 Pressure (mb) P3="),
    readln(V6),
    assertz(current("p3", V6)),
    write("Tank Pressure (mb) Pt="),
    readln(V7),
    assertz(current("pt", V7)).

work:-
    current("tc", Tc),
    current("vz", Vz),
    current("f1", F1),
    current("f2", F2),
    current("f3", F3),
    current("iz", Iz),
    current("s1", S1),
    current("pp", Pp),
    current("pr", Pr),
    current("pd", Pd),
    current("pc", Pc),
    current("pz", Pz),
    current("p1", P1),
    current("p2", P2),
```

```
    current("p3", P3),
    current("pt", Pt),
    rule(Tc, Vz, F1, F2, F3, Iz, S1, Pp,
         Pr, Pd, Pc, Pz, P1, P2, P3, Pt),
    fail.
    /* enforced backtrack for fire rules */

rule(_, h, _, h, _, h, _, _, _,
    _, _, h, _, _, _, _):-
    write("Vz Too High"), nl.

rule(_, l, _, l, _, l, _, _, _,
    _, _, l, _, _, _, _):-
    write("Vz Too Low"), nl.

rule(h, l, _, h, _, h, _, _, _,
    _, _, _, _, _, _, _):-
    write("Tc Too High"), nl.

rule(l, h, _, l, _, l, _, _, _,
    _, _, _, _, _, _, _):-
    write("Tc Too Low"), nl.

rule(n, h, _, _, _, l, _, _, _,
    _, _, l, _, _, _, _):-
    write("Vz Clip Oxidized"), nl.

rule(n, l, _, l, _, h, _, _, _,
    _, _, _, _, _, _, _):-
    write("Vz Short Circuit"), nl.

rule(_, _, _, _, _, _, p, l, h,
    l, _, _, _, _, _, _):-
    write("Manifold Switch Failure"), nl.

rule(_, _, _, _, _, _, r, l, l,
    l, _, _, _, _, _, _):-
    write("Pr Too Low"), nl.

rule(_, _, _, _, _, _, _, l, _,
    h, _, _, _, _, _, _):-
    write("Pd Too High"), nl.

rule(_, _, _, _, _, _, _, h, _,
    l, _, _, _, _, _, _):-
    write("Pd Too Low"), nl.

rule(n, _, _, _, n, _, _, _, _,
    _, h, _, h, _, _, _):-
    write("Vw Set-Point High"), nl.

rule(n, _, _, _, n, _, _, _, _,
    _, l, _, l, _, _, _):-
    write("Vw Set-point Low"), nl.

rule(_, _, _, l, _, _, _, _, _,
    _, n, h, _, h, _, _):-
    write("Vo Set-point High"), nl.
```

```
rule(_, _, _, h, _, _, _,_, _,
    _, n, l, _, l, _, _):-
    write("Vo Set-point Low"), nl.

rule(_, _, n, _, n, _, _,_, _,
    _, h, h, _, _, _, _):-
    write("Pump Degraded"), nl.

rule(_, _, n, _, n, _, _,_, _,
    _, n, _, n, n, n, _):-
    write("PST-10 Failed"), nl.
rule(_, _, h, _, h, _, _,_, _,
    _, _, _, h, _, h, _):-
    write("Fl Too High"), nl.

rule(_, _, l, _, l, _, _,_, _,
    _, _, _, l, _, l, _):-
    write("Fl Too Low"), nl.
```