MiniAC™

ANALOG/HYBRID COMPUTING SYSTEM

REFERENCE HANDBOOK

CONTENTS

PART I – FAMILIARIZATION AND GENERAL OPERATION (FOR ALL USERS)

CONTENTS (CONT)

CONTENTS (CONT)

CONTENTS (CONT)

CONTENTS (CONT)

CONTENTS (CONT)

CONTENTS (CONT)

CONTENTS (CONT)

CONTENTS (CONT)

CONTENTS (CONT)

ILLUSTRATIONS

ILLUSTRATIONS (CONT)

ILLUSTRATIONS (CONT)

TABLES

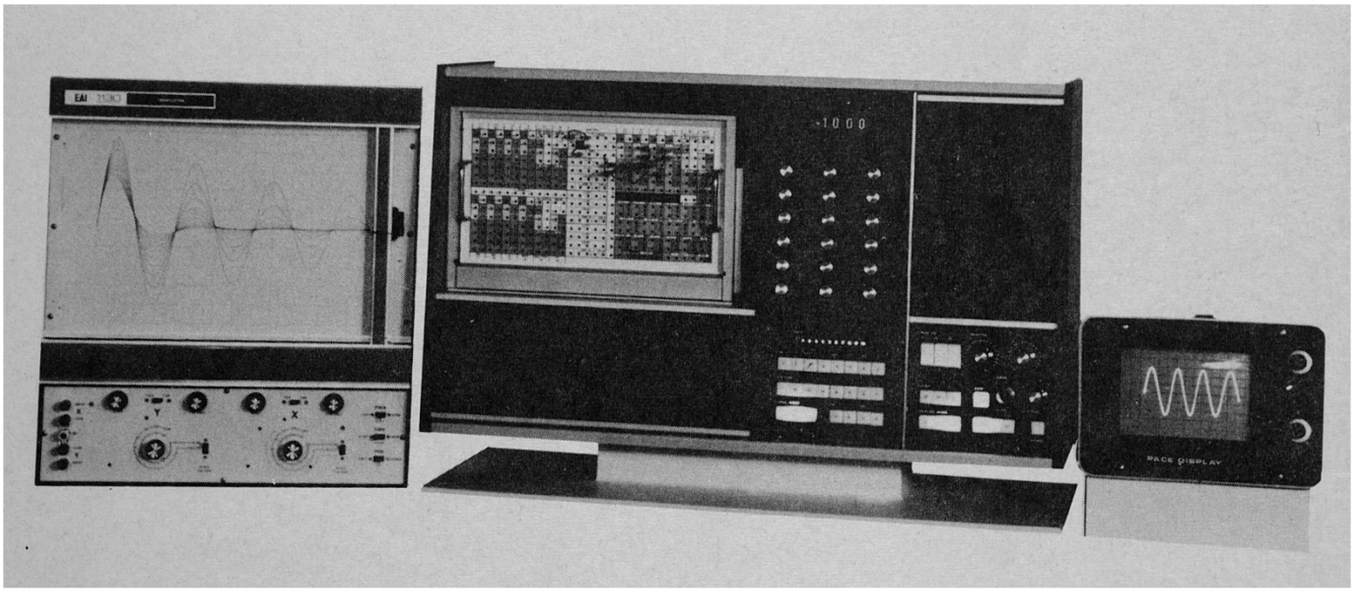


Figure 1.1 Typical MiniAC Computing System

CHAPTER 1

INTRODUCTION

1.1 THE EAI MiniAC™ (Figure 1.1)

The EAI MiniAC is an educational analog/hybrid simulation system specifically designed for technical colleges, university engineering and science departments, and industrial training groups. Peripherals that can be used with the MiniAC include an X-Y Plotter, and a display scope.

The MiniAC Console is the major system component and is a self-contained 10 volt reference analog computer. This computer contains analog and logic computational components, control logic, hybrid (analog/logic interface) components, and operating controls. A basic system can solve a non-linear second-order differential equation with an imposed limit and has provisions for an analog comparison and logical operation. A fully expanded system can solve an automatic optimization and plot two to three non-linear third order differential equations. Problems are introduced at a removable patch panel; the removable feature permits storage of programmed problems.

1.2 PURPOSE OF MANUAL

This manual is written with three specific goals in mind:

1. To familiarize *all users* with the computer organization and patching facilities, the basic functions of the various operating controls and indicators, and the general operation of the system as a whole.
2. To provide a guide for *the occasional user or trainee* that allows him to patch and run his particular problem without delving into circuit related details.
3. To provide the *relatively experienced user* those details that will permit him to utilize the MiniAC to its fullest extent.

1.3 STRUCTURE OF MANUAL

To achieve the above goals, this manual is divided into three separate and distinct parts as follows:

PART 1 – FAMILIARIZATION AND GENERAL OPERATION

PART 2 – USERS GUIDE

PART 3 – REFERENCE HANDBOOK

Each part of this manual is divided into three or more chapters. All chapters are numbered consecutively within a part, and from part-to-part.

PART 1 – FAMILIARIZATION AND GENERAL OPERATION
Is oriented toward all users and consists of Chapters 1 through 3. Describes the overall system, locates and identifies all major components, and contains general operating information for the computer as a whole. The user should be completely familiar with PART 1 before attempting to use the MiniAC or before proceeding to either PART 2 or 3.

PART 2 – USERS GUIDE
Consists of Chapters 4 through 12. This portion of the manual is directed at the infrequent user and/or the trainee. Part 2 is oriented toward problem solving and carries the user from patching and setting up the simplest mathematical function all the way up to patching and running a sample two point boundary value problem.

PART 3 – REFERENCE HANDBOOK

The reference handbook portion of this manual (Chapters 13 through 29) is directed toward the frequent or relatively experienced user. Part 3 describes in detail the workings of the MiniAC subsystem by subsystem and component by component. Sufficient information is provided to permit the experienced programmer to fully utilize all programmable features to devise unique patching schemes on a subsystem and an individual component basis.

1.4 HOW TO USE THIS MANUAL

First, read Part 1 to become thoroughly familiar with the organization of the machine, the patch panel, the basic function of the various controls and indicators, and the general operating procedures. Then, if reasonably on board with Analog Programming and analog computers in general, refer to the reference handbook portion of the manual. If this is your first introduction to an analog computer or you have seldom used one, it is advisable to proceed directly to the Users Guide. In either case (experienced or trainee), a cursory review of the Table of Contents will aid you in locating specific information required to patch and run your particular problem.

CHAPTER 2

MACHINE ORGANIZATION

2.1 INTRODUCTION

This chapter provides information to acquaint the user with the appearance, location, and basic functions of major components of the EAI MiniAC. Figure 1.1 illustrates a typical system with peripheral devices. The MiniAC is the heart of the system and contains all necessary analog programming facilities, computing elements, control features, and basic monitoring devices. All uncommitted programmable components are terminated at the patch panel and are interconnected with patch cords as required by the specific program. The peripheral devices permit the operator to interact with the master program and graphically record or document instantaneous problem solutions. The following paragraphs give a brief description of those portions of the console (Figure 2.1) that are of prime importance to the programmer/operator.

2.2 THE ANALOG COMPONENTS

A fully expanded MiniAC provides an array of analog computational devices. Table 2.1 lists the analog equipment complement provided with a fully expanded MiniAC. This table also includes specific chapter references in which each component is described. With the exception of the manual switches, coefficient potentiometers, and the variable function generator, all analog components that perform mathematical functions are located directly behind the setup switch/overload indicator panel.

Table 2.1. Equipment Complement: Analog Components

Component	Quantity	Chapter Reference	
		Users Guide	Reference Handbook
Linear (Including those associated with multipliers, function generators, etc.):			
Summer/Integrator ($\Sigma/\int$)	6	5	21
Five-Input Summer/High Gain (Σ)	3		19
Five-Input Summer/Track Store (Σ/TS)	3		20
Two-Input Summer/Inverter (MULT/DIV)	3		23
Three-Input Summer (FG)	2		24
Inverter (VFG)	2	5	25
Potentiometers	18	5,6	18
Non-Linear:			
Multiplier (MULT/DIV):	3	5	23
Function Generators:			
Log/Exponential	2	5	24
Variable (10 Segment)	2	7	25
Interval Timer:	1	9,10	29
Manual Switches:	3	5	28
Trunks:	12	–	14
Digital Voltmeter (DVM)	1	6	16

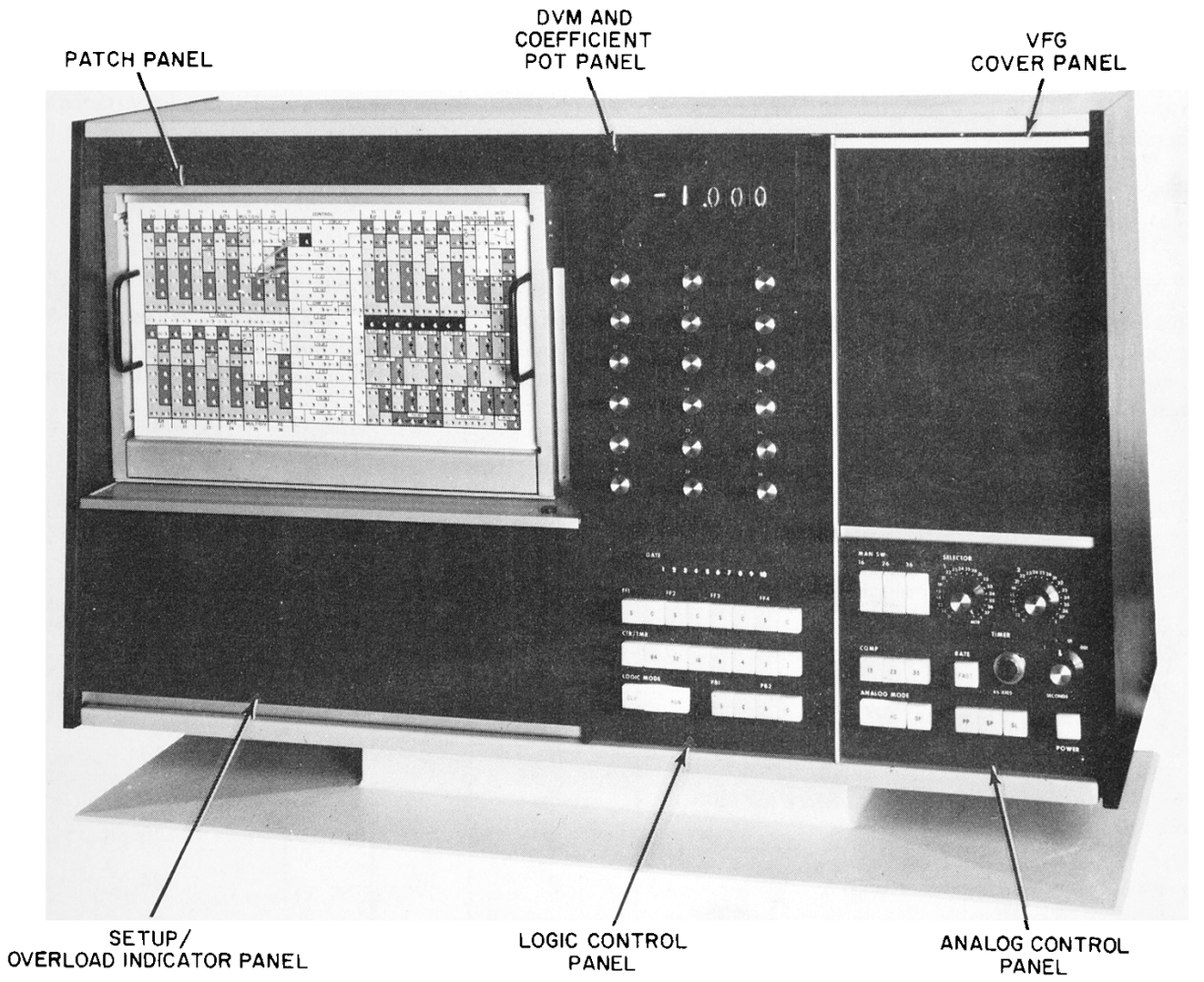


Figure 2.1. Typical MiniAC

2.3 THE LOGIC COMPONENTS

The various logic elements available for use in the MiniAC provide operational flexibility to physical system simulation. These logic elements permit simulation of descrete and continuous systems and provide a wide range of flexibility in problem control. In other words, a logical event can be used to control an analog function and vice versa. All logic elements (such as pushbutton flip-flops, general purpose flip-flops, and the counter/timer) are physically located directly behind the logic control panel (Figure 2.1). Table 2.2 lists the equipment complement provided with the logic expansion and specifies the chapter(s) in which each component is described.

Table 2.2. Equipment Complement: Logic Components

Component	Quantity	Chapter Reference	
		Users Guide	Reference Handbook
Gates:			
Two Input	6	11	26
Four Input	4		
Counter/Timer:	1	11	29
Flip-Flops:			
General Purpose	4	11	27
Pushbutton	2		
Monostable Multivibrator (One Shot):	1	11	29

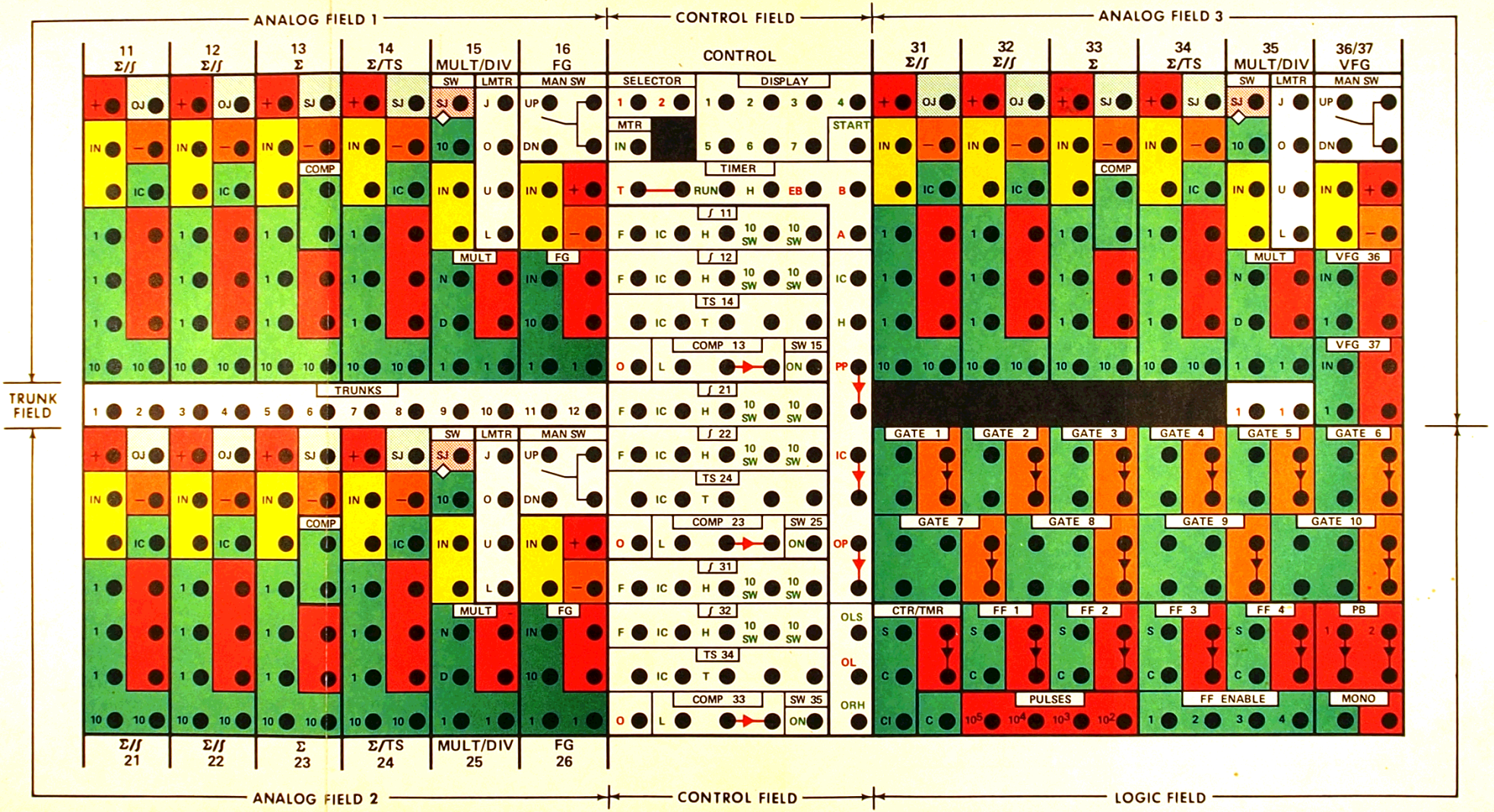
2.4 THE ANALOG/LOGIC INTERFACE COMPONENTS

Analog/logic interface components provide interface and control between the analog and logic portions of a program. These devices include analog controlled digital (A/D) comparators and digital controlled analog (D/A) switches.

The equipment complement of analog/logic interface components is listed in Table 2.3. This table also includes specific chapter references in which these devices are described. Like the analog and logic components, these hybrid devices are terminated at the patch panel.

Table 2.3. Equipment Complement: Analog/Logic Interface Components

Component	Quantity	Chapter Reference	
		Users Guide	Reference Handbook
A/D Comparators:	3	5	28
D/A Switches:			
Uncommitted (SW)	3	5	28
Summer/Integrator Committed (10SW)	12	5	21


ANALOG FIELD 1
CONTROL FIELD
ANALOG FIELD 3
TRUNK FIELD
11 Σ/∫
12 Σ/∫
13 Σ
14 Σ/TS
15 MULT/DIV
16 FG
CONTROL
31 Σ/∫
32 Σ/∫
33 Σ
34 Σ/TS
35 MULT/DIV
36/37 VFG
SELECTOR
DISPLAY
MTR
START
TIMER
COMP 13
COMP 23
COMP 33
SW 15
SW 25
SW 35
TRUNKS
GATE 1
GATE 2
GATE 3
GATE 4
GATE 5
GATE 6
GATE 7
GATE 8
GATE 9
GATE 10
CTR/TMR
FF 1
FF 2
FF 3
FF 4
PB
PULSES
FF ENABLE
MONO
Σ/∫ 21
Σ/∫ 22
Σ 23
Σ/TS 24
MULT/DIV 25
FG 26
ANALOG FIELD 2
CONTROL FIELD
LOGIC FIELD

2.5 THE CONTROL AND MONITORING SYSTEMS

The MiniAC is equipped with versatile control and monitoring systems that permit: selection of addressable analog components for value readout and display; control of analog time scale; control of analog and logic modes; and slave operations. Complete monitoring facilities are provided that permit status readout of logic elements as well as indicating amplifier overload conditions. All aspects of these systems are described in Chapters 14 through 17 of the Reference Handbook.

2.6 THE PATCH PANEL

2.6.1 INTRODUCTION

The following paragraphs describe the patch panel layout. This includes: general patch panel location of individual components and controls; the component location numbering system (address); and general color coding information. Detailed information concerning patch panel layout for individual components and controls (including patching techniques and methods) are given in subsequent chapters describing the specific component or control system.

The removable panel (Figure 2.2) provides readily accessable termination and interconnection points for assembling (patching) devices that perform mathematical and logical functions in any array required to obtain a given problem solution. The patch panel also provides patching facilities for problem control purposes and problem solution outputs.

Patching is the act of interconnecting the mathematical and logical elements so that the assembled (patched) devices correspond to the "paper" program. Patching may be performed with the panel on the computer, or "off-line". If more than one patch panel is available, a problem may be prepatched (remotely from the machine) while another problem is being run. In other words, one user can produce a problem solution while another is preparing a problem. The availability of multiple patch panels permits retaining (storing) a patched problem or simulation that may be used frequently for demonstration or other purposes.

2.6.2 PATCH PANEL LAYOUT

2.6.2.1 General

For ease in programming and locating components for patching, the MiniAC patch panel is divided into symmetrical fields. Basically, there are three nearly identical analog fields, an analog trunk field, a logic field, and a control field. Each field is distinct and readily identified. Figure 2.3 is a simplified diagram of the patch panel and shows the general patching outlines and areas associated with each field. As an aid in identifying the various patching areas, compare this illustration to the patch panel details shown in Figure 2.2.

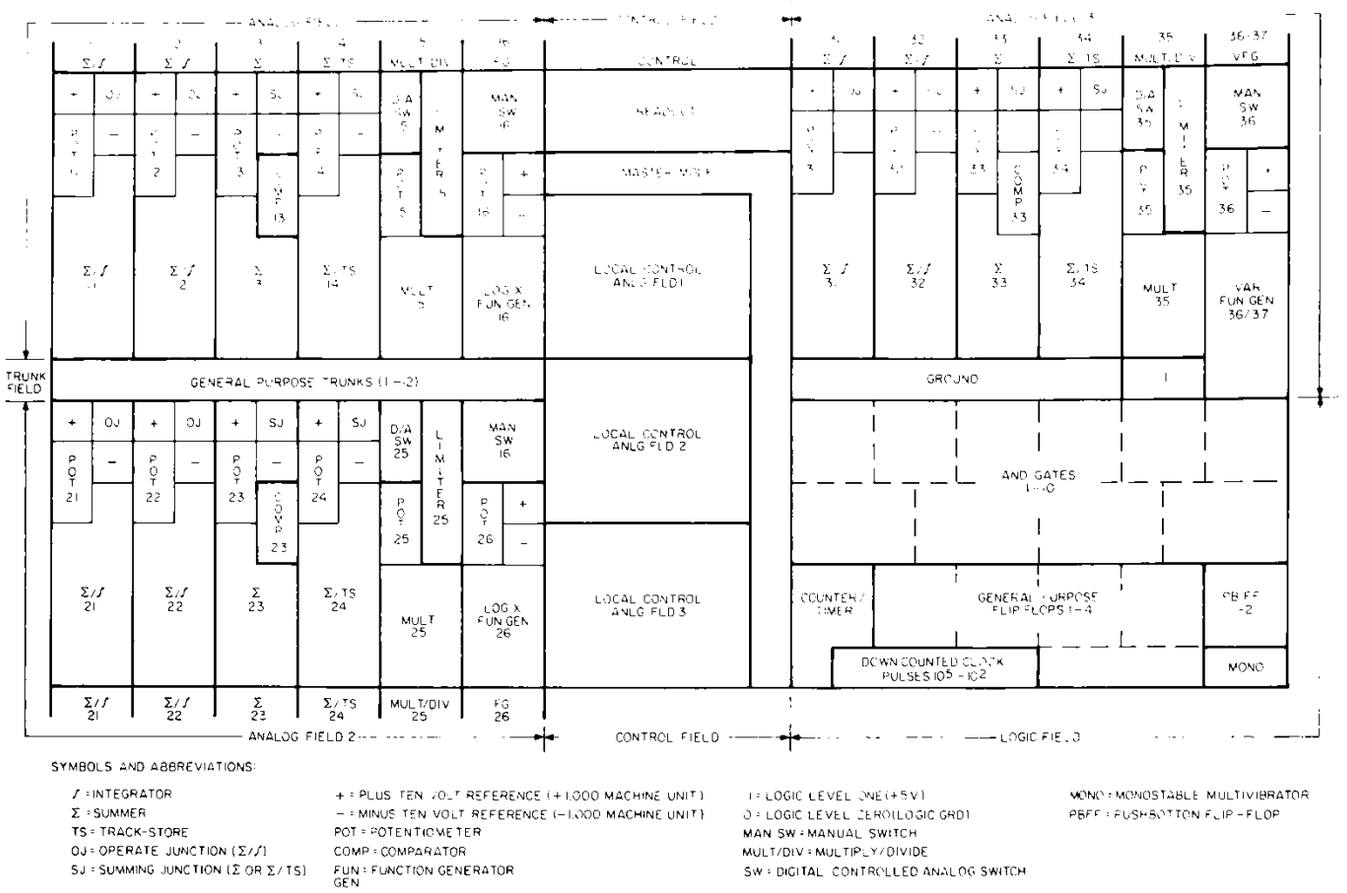


Figure 2.3. General Patching Areas

2.6.2.2 The Analog Fields

Basically, each analog field is divided into six rectangular patching areas, or modules. Each module provides terminations for at least two analog components. For patching convenience, each module includes terminations for a potentiometer, and computer reference (±1.0 machine units) is distributed throughout each field.

A two digit number embossed on the patch panel identifies the patching module and the field. The first digit (1-3) is the field designation. The second digit is the assigned module number (1-6). For example, the fifth patching module in field one is assigned the number 15 (field 1, module 5). Therefore, analog components terminated in this field/module are identified by the number 15.

In addition to the field/module numbering system, modules are further identified by a symbol or mnemonic designation that indicates the major computing component terminated in that module. For example, modules 1 and 2 in each field are marked with the Σ/∫ symbol (Figures 2.2 and 2.3). This indicates that the major computing

component terminated in these modules is the summer/integrator. Table 2.4 lists each component and facility terminated in the patch panel analog field, identifies each by symbol or mnemonic designation and module number, and lists the quantity per field.

Table 2.4. Patch Panel Field Assignments for Analog Components

Analog Component or Facility	Symbol or Mnemonic Panel Marking	Module Number	Quantity Per Field			Totals
			1	2	3	
Summer/Integrator	$\Sigma/\int$	1,2	2	2	2	6
Summer/High Gain	Σ	3	1	1	1	3
Comparator	COMP	3	1	1	1	3
Summer/Track Store	Σ/TS	4	1	1	1	3
Multiplier	MULT/DIV	5	1	1	1	3
Digital Controlled Analog Switch	SW	5	1	1	1	3
Limiter	LMTR	5	1	1	1	3
Log X Function Generator	FG	6	1	1	0	2
Manual Switch	MAN SW	6	1	1	1	3
Variable Function Generator	VFG	6/7	0	0	2	2
Potentiometer	None (Colored coded yellow)	1-6	6	6	6	18
Positive Reference	+	1-4,6	5	5	5	15
Negative Reference	−	1-4,6	5	5	5	15
Analog Ground	None (Color coded black)	N/A	0	0	8	8

To simplify the process of patching a problem, the analog fields use a readily identifiable color code. The principle colors used are green, red, orange, yellow, white and black.

Green = Analog component inputs.

Red = Analog component outputs and positive reference.

Orange = Minus reference

Yellow = Potentiometers

White = Passive element manual switches and limiters

2.6.2.3 The Trunk Field

The trunk field is located between analog fields one and two and is identified by the designation TRUNKS. These terminations provide 12 general purpose trunk lines for exchanging analog data with another computer or for use with external analog devices.

2.6.2.4 The Logic Field

All logic computing elements are terminated in the patching area directly below Analog Field 3 (Figures 2.2 and 2.3). Logic component terminations are contained in a separate field to minimize cross-talk (noise pick-up) and to reduce the possibility of inadvertently patching an analog signal to a logic component, or vice versa.

The logic field has provisions for program patching of: AND gates, a counter/timer; general purpose and pushbutton flip-flops, a monostable multivibrator, and down counted clock pulses. With the exception of AND gates, all logic outputs on the patch panel are synchronized with the system clock. Each logic component is identified by panel markings as indicated in the following list:

Logic Component	Panel Designation	Quantity
Gates:		
Two-Input	GATE 1 to GATE 6	6
Four-Input	GATE 7 to GATE 10	4
Counter/Timer	CTR/TMR	1
General Purpose Flip-Flops	FF1 to FF4	4
Pushbutton Flip-Flops	PB1 and PB2	2
Down Counted Clock	PULSES (10^5, 10^4, 10^3, 10^2)	4
Monostable Multivibrator (One Shot)	MONO	1

To simplify the process of patching a problem, the logic field has a color code similar to that of the analog fields. The colors used are green, orange, and red. These indicate the following:

Green = Logic inputs (1 = +5V; 0 = GRD)

Orange = Asynchronous (unclocked) logic outputs.

Red = Synchronous (clocked) logic outputs.

Red on White = Logic level output.

2.6.2.5 The Control Field

The control field (Figures 2.2 and 2.3) is the white patching area separating analog fields 1 and 2 from analog field 3 and the logic field. Primarily, the control field provides patching facilities to permit logic signals to be used for individual component and computer mode control. The control field is functionally divided into three major areas as follows:

1. Readout Selection and Control
2. Master mode Control
3. Local Control

The readout area is located in the uppermost portion of the control field and permits patching the readout selector to external devices, as well as logic start/stop control of a remote display device (recorder or oscilloscope).

The inverted L shaped area below the readout termination group is devoted to logic control of computer modes. That is, if the patch panel (PP) is selected as master, the IC, H and OP modes can be selected by patching of logic signals. An interval timer is terminated in this area to permit automatic repetitive operations.

The patch panel provides terminations that permit individual logic control of analog components terminated in analog fields 1 through 3. Controlling individual analog components with logic levels patched at the control field is referred to as local control. Local control functions that can be performed by patching include: integrator time scale and mode; track-store/summer mode; D/A switch operation; and latch control of comparators. These logic control terminations are divided into three identical groups (one for each analog field) and are identified using the field/module numbering system described in Paragraph 2.6.2.2. To avoid confusion and for simplicity of patching, all local controls are marked with the corresponding field/module number and are further identified by the applicable symbol or mnemonic designation. As in the logic and analog fields, the control field is also color coded. However, in this case the panel symbol or lettering is color coded, not the background. The color codes used are green, red, and black. These indicate the following:

Green = Logic input in all areas. DISPLAY 1-7 and MTR-IN may also be used as analog inputs.

Red = Synchronous logic output or fixed logic level (0 = logic zero output).

Black = Analog ground.

2.7 DVM AND COEFFICIENT POT PANEL

The DVM and coefficient pot panel (Figure 2.4) contains the digital voltmeter (DVM) and the controls required to set constant coefficients into an analog program. The DVM displays the numeric value (in machine units) of any component selected for readout. Refer to Chapter 6 in the User's Guide or Chapter 16 in the Reference Handbook portion of this manual.

Problem coefficients are normally provided by the preset potentiometers mounted on this panel. These pots are arranged in three vertical columns with six in a column. Each pot is terminated at the patch panel and is identified by the numerical panel marking that corresponds to the patch panel address described in Paragraph 2.6. The individual pot knobs are used to preset the desired coefficient value. The pushbutton switches immediately to the right of each pot are used to display the pot output on the DVM. Pot setting procedures are given in Chapters 6 and 18 of the User's Guide and Reference Handbook respectively.

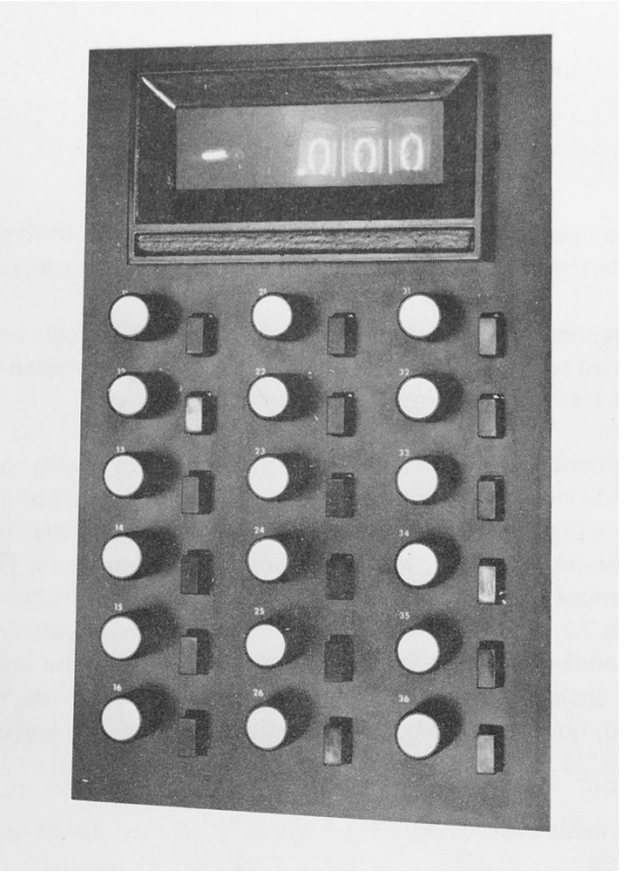


Figure 2.4. DVM and Coefficient Pot Panel

2.8 THE VFG PANEL

The variable diode function generator (VFG) is located directly behind the removable VFG panel (Figure 2.1). This panel is only removed during set-up of the VFG and should remain in place at all other times to avoid inadvertently disturbing the VFG setup. The VFG is described in Chapter 7 of the User's Guide and Chapter 25 of the Reference Handbook.

2.9 THE ANALOG CONTROL PANEL

The analog control panel (Figure 2.5) is located at the right side of the MiniAC directly below the VFG panel. This control panel contains the power on switch; the computer analog mode controls; the time scale (rate) control; the interval timer controls; the readout signal selector; manual switches and manual controls for the comparators. The basic description and function of each control and indicator is given in Table 2.5.

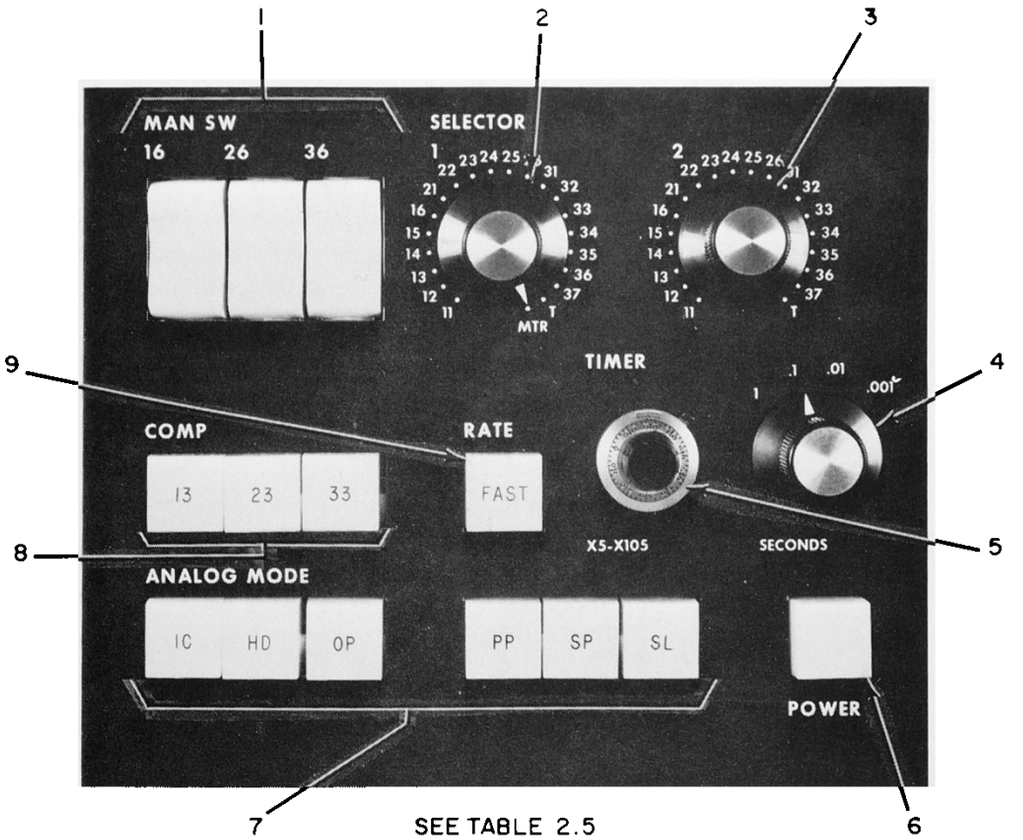


Figure 2.5. Analog Control Panel

Table 2.5. Analog Control Panel: Controls and Indicators (Ref. Figure 2.5)

Index No.	Control or Indicator			Function	Chapter Ref.	
	Type	Desig.	Position or Section		Users Guide	Ref Hdbk
1	3-Position Toggle Switch	MAN SW (16,26,36)		Used for manual control of analog functions. Numbered designations correspond to patch panel address in which these switches are terminated.	5	28
			Up	When up (top of switch flush with panel), UP termination on patch panel is connected to the switch wiper (arm).		
			Down	When down (bottom of switch flush with panel), DN termination is connected to switch wiper.		
			Centered	When centered, neither UP or DN is connected to switch wiper.		

(Cont)

Table 2.5. Analog Control Panel: Controls and Indicators (Continued)

Index No.	Control or Indicator: Type	Control or Indicator: Desig.	Control or Indicator: Position or Section	Function	Chapter Ref.: Users Guide	Chapter Ref.: Ref Hdbk
2	21-Position Rotary Switch	SELECTOR-1		Selects addressed component for readout on DVM. Connects output of selected device to SELECTOR-1 output termination on patch panel.	8	16
			11-37	Setting SELECTOR-1 at any one of these positions selects the amplifier output of the addressed device for readout as follows: 11,12,21,22,31,32=Σ/∫ 13,23,33=Σ/HG 14,24,34=Σ/TS 15,25,35=MULT/DIV 16,26=FG 36,37=VFG		
			T	Selects ramp output of timer for readout.		
			MTR	Selects any signal patched to MTR-IN termination for readout.		
3	20-Position Rotary Switch	SELECTOR-2	11-37 and T	Connects output of addressed component to SELECTOR-2 patch panel terminations. All positions identical to SELECTOR-1 except there is no MTR position.	8	16
4	4-Position Rotary Switch	TIMER-SECONDS	1 .1 .01 .001	Selects duration and range of timer intervals as follows: Pos. / A Interval (Approx.) / B Interval (Range) 1 / 0.7S / 5S-105S .1 / 0.7S / .5S-10.5S .01 / 7MS / 50MS-1.05S .001 / 7MS / 5MS-0.105S	9,10	29
5	10 Turn Calibrated Potentiometer	TIMER X5-X105	Adjustable	Selects duration of Timer B interval. Direct dial readout is multiplied by TIMER-SECONDS switch setting.	9,10	29
6	Latching Pushbutton Switch	POWER	Set Reset	When set (depressed), applies ac power to the MiniAC. When reset (released), removes power.		3

(Cont)

Table 2.5. Analog Control Panel: Controls and Indicators (Continued)

Index No.	Control or Indicator			Function	Chapter Ref.	
	Type	Desig.	Position or Section		Users Guide	Ref Hdbk
7	Pushbutton Lamp Switches (5)	ANALOG MODE		Permit selection of any one of four analog modes and source of master mode control. All except SL light to indicate selected mode.	9,10	14,17
			IC	Initial Condition: When depressed, integrators attain their initial values.		
			HD	Hold: When depressed, integrators not under local mode control enter Hold (H) mode.		
			OP	Operate: When depressed, places integrators not under local mode control in Operate (OP) mode.		
			PP	Patch Panel: When depressed selects patch panel control of the IC, OP and H modes.		
			SP	Set Pot: When depressed, selects Set Pot mode. Essentially, places feedback around all amplifiers to permit pots to be set under normal load conditions.		
	Latching Pushbutton		SL	SLAVE: When two computers are interconnected, one is usually the master and the other the slave. When depressed, this switch selects the other console as master.		
8	Pushbutton Lamp Switches	COMP (13,23,33)		Primarily used during program checkout to force the corresponding comparator to its opposite state. When lit, comparator true output is high. When extinguished, output is low.	11	17,28
9	Latching Pushbutton Lamp Switch	RATE-FAST	Set	Selects integrator FAST time scale. FAST lamp lights and the problem solution is 500 times faster than normal.	11	14,17
			Reset	When reset, the problem solution is at the normal (or real-time) rate. Lamp is extinguished.		

2.10 THE LOGIC CONTROL PANEL

The logic control panel (Figure 2.6) is located directly below the readout and coefficient pot panel. The logic control panel has pushbuttons for controlling; the logic modes, the counter/timer, and all pushbutton and general purpose flip-flops. Indicators under the pushbuttons display the status of these elements. Additionally, this panel displays the status of all AND gates. Refer to Table 2.6 for a description of the logic controls and indicators.

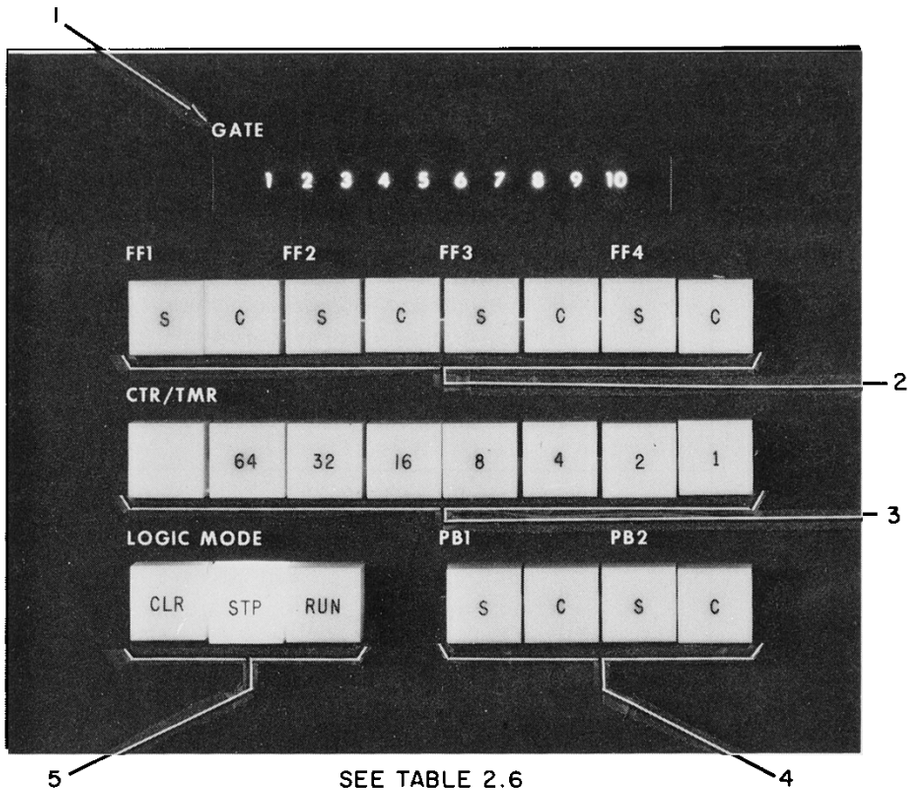


Figure 2.6. Logic Control Panel

Table 2.6. Logic Control Panel: Controls and Indicators (Ref. Figure 2.6)

Index No.	Control or Indicator			Function	Chapter Ref.	
	Type	Desig.	Position or Section		Users Guide	Ref Hdbk
1	Indicator Lamps	GATE (1-10)		Light to indicate that corresponding GATE output is true. Note that indicators for unpatched gates remain lit.	11	17,26
2	Pushbutton Lamp Switches (4 pair)	FF1-FF4	S-C	The S pushbuttons display status of corresponding numbered flip-flop. When S is lit flip-flop is set. When extinguished flip-flop is reset or cleared. Depressing S sets flip-flop. Depressing C clears flip-flop. Used for initializing and program checkout.	11	17,27

(Cont)

Table 2.6. Logic Control Panel: Controls and Indicators (Continued)

Index No.	Control or Indicator			Function	Chapter Ref.	
	Type	Desig.	Position or Section		Users Guide	Ref Hdbk
3	7 Pushbutton switches, 1 momentary	CTR/ TMR	64-1 and Blank	Numbered pushbuttons used for pre-selecting desired count. These pushbuttons light as count is attained. The undesignated (blank) switch inverts counter/timer output for program checkout. This pushbutton also lights whenever counter/timer output is true.	11	17,29
4	Pushbutton Lamp Switches (2 pair)	PB1, PB2	S-C	Manual controls for pushbutton flip-flops. Depressing S provides logic one at true output of corresponding PB termination. Depressing C cause logic zero at that point. When PB is set, S indicator lights.	11	17,27
5	Pushbutton Lamp Switches	LOGIC MODE		Control and display logic operating mode. When any one switch is depressed, lights to indicate selected mode.	11	15,17
			CLR	Clear: When depressed clears all general purpose and pushbutton flip-flops, counter/ timer, and down counted clock (PULSES). Logic then enters STP (Stop) mode.		
			STP	Stop/Step: When initially depressed stops running clock and all logic elements hold present state. Subsequent switch operation generates one clock pulse each time switch is depressed. Used during program checkout to manually step logic.		
			RUN	When depressed, enables running clock and starts logic program.		

2.11 SETUP SWITCH/OVERLOAD INDICATOR PANEL

2.11.1 INTRODUCTION

The setup switch/overload indicator panel is located directly below the patch panel. This transluscent drop down panel conceals the card file containing the majority of the MiniAC analog devices. Figure 2.7 illustrates the card location and general arrangement of devices contained in the card file. Refer to this illustration throughout the remainder of this description.

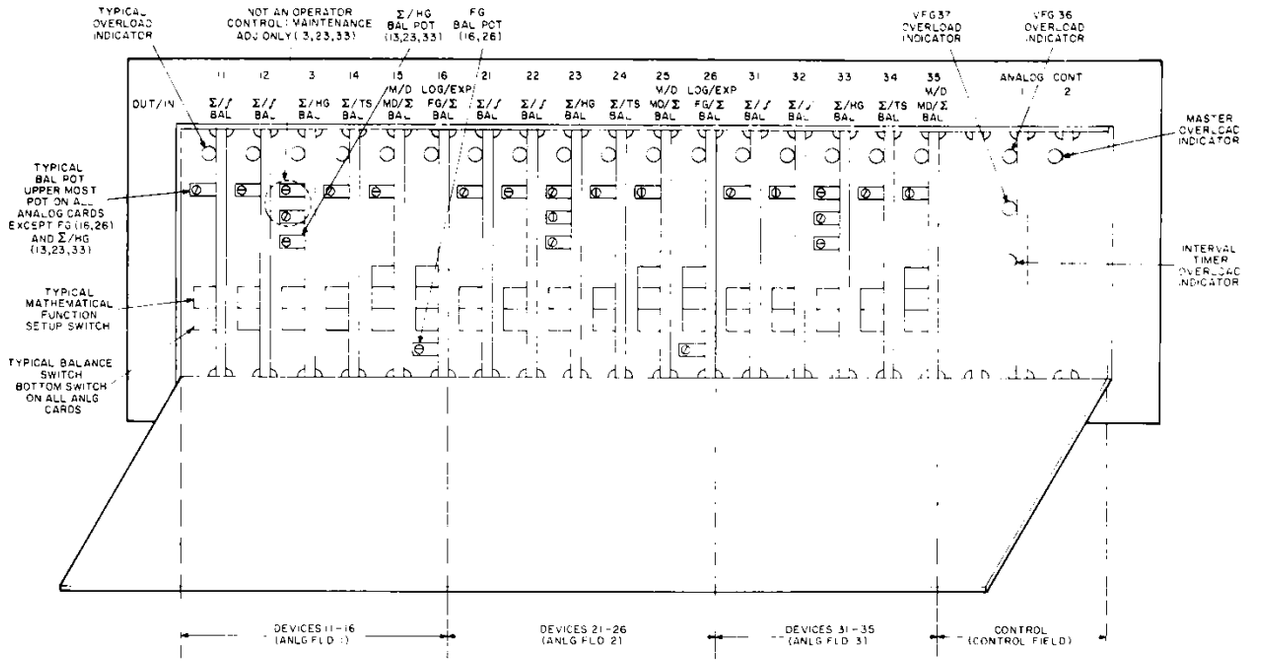


Figure 2.7. Setup Switch/Overload Indicator Panel (Open)

2.11.2 THE IDENTIFICATION STRIP

The numbered designations (11-16, 21-36, and 31-35) on the identification strip at the top of the card file correspond to the patch panel module address described in Paragraph 2.6. Note that the only analog components not installed in the card file are the VFG's (devices 36 and 37), the coefficient potentiometers, and the manual switches.

The mathematical symbols and mnemonics directly below each module number identify the functions of the various pushbutton switches mounted on each module. For example, the $\frac{\Sigma/\int}{BAL}$ designation for device 11 indicates that the upper switch $(\Sigma/\int)$ selects the device function. The designation BAL relates to the bottom switch and is used only during amplifier balancing procedures. The exact functions of the various switches are described in Paragraphs 2.11.3 and 2.11.4. The OUT/IN (black/red color code) at the extreme left of the identification strip indicates the setup switch position for selecting a device function and corresponds to the color codes for each switch symbol.

2.11.3 THE SETUP SWITCHES

The majority of the MiniAC analog devices are capable of performing two or three mathematical functions. In most analog computers having multi-purpose devices, the selection of a given function is performed by patching. To simplify

the patching process the components of the MiniAC make use of setup switches. These push-to-set, push-to-release switches are mounted directly on the component card and must be set up prior to starting a problem run. Each card has either one or two set up switches. For example, device 16 is as capable of performing three mathematical functions. It may be used as a log X function generator, an exponential (antilog) function generator, or as a summer. The desired function for device 16 is selected using the upper two switches on the card (Figure 2.7). The switch location and its function is identified by the symbols on the identification strip at the top of the card file. The LOG/EXP designation corresponds to the uppermost switch, while the FG/Σ symbol corresponds to the next lower switch. The slash mark designates the switch position. The term to the left of the slash mark (color coded black) signifies that this function is selected when the switch is released (out). The term to the right of the slash mark (color coded red) signifies the function selected when the switch is set (IN). These color codes correspond to the color coded OUT/IN designation at the left of the identification strip. For example, if both device 16 LOG/EXP and FG/Σ switches are released (OUT) the device functions as a log X function generator (LOG-FG). If the (LOG/EXP) switch is set (IN), and the FG/Σ switch released (OUT), the device functions as an exponential function generator (EXP-FG). If the FG/Σ switch is set (IN), the device functions as a simple three input summer (Σ). When using any device having two setup switches (15, 25, 35, and 16, 26), the lower switch (when IN) overrides the upper switch.

Table 2.7. lists the various devices and the setup switch function for each.

Table 2.7. Setup Switch Functions

Module Number	Type Unit	Designation (OUT/IN Black/Red)	Switch Functions
11, 12 21, 22 31, 32	Summer/Integrator	$\Sigma/\int$	When OUT, device performs summation or inversion. When IN, device performs integration with respect to time.
13, 23 33	Summer/High-Gain	Σ/HG	When OUT, device performs summation or inversion. When IN, device performs high-gain summation. When used in high-gain applications, external feedback must be patched.
14, 24, 34	Summer/Track-Store	Σ/TS	When OUT, device performs summation or inversion. When IN, device tracks the sum of the inputs and stores the output on command.
15, 25, 35	Multiply/Divide/Summer	M/D	When OUT, device perform multiplication (the product of two variables). When IN, the device performs the function of division.
		MD/Σ	When OUT, device performs multiplication or division as determined by M/D switch. When IN, device is a two input summer/inverter.
16, 26	Log/Exponent/Summer	LOG/EXP	When OUT, device is a Log X function generator. When IN, device is an exponential (antilog) function generator.
		FG/Σ	When OUT, device operates as Log X or antilog as determined by LOG/EXP switch. When IN, device is a three input summer.

2.11.4 AMPLIFIER BALANCE CONTROLS

Each amplifier in the MiniAC (Figure 2.7) is provided with controls to permit amplifier balancing. These consist of a balance potentiometer and a momentary pushbutton balance switch. The balance switch is the bottom switch on the card and is identified by the term BAL directly below the setup switch designations on the identification strip. The balance switch on each card operates independently of the setup switches to permit amplifier balancing regardless of the selected device function. Refer to Chapter 3 for amplifier balancing procedures.

2.11.5 THE OVERLOAD INDICATORS

As illustrated in Figure 2.7, each device having an amplifier is equipped with an overload indicator. When a device is forced out of its correct operating range, the corresponding indicator lights and provides easy identification of the overloaded component. Overloads are commonly caused by incorrect scaling of one or more problem variables. Refer to Reference Handbook, Chapter 17 for a more detailed description of the overload warning system and overload criteria.

2.12 EXTERNAL CONNECTIONS

There are provisions at the rear of the MiniAC for connection of ac power, and interconnecting peripheral devices and another MiniAC for slave operations. The following list gives the function of each rear mounted connector.

Connector Designation	Function
AC	Prime power connection. This connector must be connected to correct prime power source: 115V ±10% or 230V ±10%, 50 – 60 Hz. Refer to Maintenance Manual, EAI Publication Number 00 800.2072.
DISPLAY	Provides interconnection points for EAI peripheral display devices such as the Models 1110, 1130, and 1140 XY Plotter, and the Model 0.34.0035-1 Display Scope. Refer to Chapters 9 and 10 of the Users Guide.
TRUNK	Provides interconnection for exchanging analog data signals between two computers or external analog devices. The internal trunk lines are terminated in the patch panel TRUNK field. Refer to Chapter 14 of the Reference Handbook.
SLAVE	Provides interconnection with another computer during slave operations. The slaved computer will respond to all analog mode commands and time scale selection generated by the master console. Refer to Chapter 14 of the Reference Handbook.

CHAPTER 3

GENERAL SYSTEM OPERATIONS

3.1 INTRODUCTION

This chapter is devoted to overall system operations such as: applying power; installing the patch panel; typical sequence of operations when running a problem; and the methods of recording a problem solution. Unless experienced with analog computers in general the operator/programmer should not attempt to operate the MiniAC until thoroughly familiar with machine organization and the functions of all operating controls and indicators (Chapter 2), and generally familiar with all aspects of the control and monitoring systems (Chapters 14 through 17).

NOTE

If this is the initial turn-on and setup of the MiniAC, do not apply power or attempt to operate this machine until all initial checkout procedures given in the maintenance manual (EAI Publication Number 00 800.2072) have been performed.

Although amplifier balancing is generally considered as a maintenance routine, procedures are provided to permit the operator to randomly check amplifier balance in problems where precise balancing may be deemed critical.

3.2 APPLYING POWER

Operating power is applied to the MiniAC by depressing the POWER pushbutton and then manually selecting either the IC or SP analog modes and the logic CLR mode.

Apply power and observe the MiniAC for normal operation as follows:

1. Depress the POWER pushbutton. (Figure 2.5)
2. Depress the ANALOG MODE – IC or SP pushbutton, (Figure 2.5) and observe that the corresponding lamp lights. In SP, the IC lamp also lights.
3. Observe that the overload indicators (Figure 2.7) may randomly indicate overload conditions when power is first applied.
4. Depress the LOGIC MODE – CLR pushbutton and observe that the STP indicator lights and all other synchronous logic indicators are extinguished (GATE indicators depend on patching).
5. If the accuracy of the problem solution is critical, permit the MiniAC to warm-up for approximately 10 minutes prior to performing a program run.

3.3 PATCH PANEL INSTALLATION AND REMOVAL

The patch panel may be installed with or without power applied to the MiniAC. Install the patch panel as follows:

1. Ensure that the locking lever on the right side of the patch bay is at the fully extended position.
2. Grasp patch panel handles and align the upper and lower locating pins (on the left and right sides of the panel) with the corresponding slots in the patch bay.
3. Gently insert the pins in the slots until the panel is secured by the spring loaded ball retainers.
4. Raise the locking lever until it snaps in the closed position (flush with patch bay).

CAUTION

Do not force the lever to the locked position. If force is required, lower the lever, remove the patch panel, and install it correctly.

Remove the patch panel as follows:

1. Pull the locking lever to the fully extended position.
2. Grasp the patch panel handles, and pull panel straight back.

3.4 RUNNING A TYPICAL PROBLEM

3.4.1 INTRODUCTION

The following paragraphs (3.4.2 through 3.4.6) outline the general order of procedure, and provide general instructions for setting up a typical program. These instructions include preliminary procedures; methods of initializing the analog and logic components; procedures for testing the program; and the procedure for obtaining the dynamic solution. Note that these instructions are general in nature and are only provided as a guideline. As applicable, specific chapter and/or paragraph references are included to direct the user to the detailed operating procedure. Throughout the following procedures, it is assumed that the program diagram, and program sheet is completed. The program sheet is described in Paragraph 3.7.

3.4.2 PRELIMINARY PROCEDURE

1. Patch the problem in accordance with the program diagram.
2. Install the patch panel (Paragraph 3.3).
3. Select the analog SP and the logic CLR modes.

3.4.3 ANALOG INITIALIZING PROCEDURE

1. Using the Program Sheet (Paragraph 3.7), set each analog component function switch for the required method of operation. Ensure tha the VFG SETUP switches are at OPER.
2. Ensure that the computer is in SP, and using the Program Sheet, set all pots (Chapters 6 and 18) used in the problem as well as any that are to be used during problem checkout.
3. Set each MAN SW to the position required by the program (Chapters 5 and 28).
4. Depress the ANALOG MODE-IC pushbutton and set each limiter level (Chapter 22).
5. If not already setup, set the VFG for the desired function (Chapters 7 and 25).

3.4.4 LOGIC INITIALIZING PROCEDURES

1. Set each general purpose flip-flop (GPFF) and each pushbutton flip-flop (PBFF) to the desired initial state (Chapter 27).
2. Using the CTR/TMR pushbuttons, preselect the desired count (Chapter 29) as indicated on the program diagram.
3. Set the TIMER (Chapter 29) for the desired interval as indicated on the program diagram.

3.4.5 TESTING THE PROBLEM

A large portion of problem patching may be static tested in the Initial Condition (IC) mode. When testing the problem, the various parameters should be verified against the calculated parameters and should be recorded as a part of problem documentation. Refer to Chapter 8 of the Users Guide for more information concerning static test methods.

Integrators with zero output in the IC mode can be checked by applying arbitrary initial conditions (usually ± reference) to their IC inputs. This can be accomplished by the use of manual switches (that should be included in the program diagram) or by extraneous patching. In either case, all units must be returned to their original initial conditions prior to starting a program run.

Perform problem checkout as follows:

1. Select the IC mode and apply any arbitrary inputs required for checkout.
2. Using SELECTOR-1 address each mathematical element in the problem (in turn) for readout on the DVM.
3. Compare the selected amplifier output with the calculated values.
4. Using the individual pot readout pushbutton (Chapters 6 and 16), readout the value of each pot patched into the problem. Verify the values.
5. Remove any arbitrary initial conditions applied for program checkout.

3.4.6 DYNAMIC SOLUTION

The exact method of obtaining a problem solution is entirely dependent on the type of problem, the desired results, and the recording methods (if any) used. Some problems may require a single run, while others may require several runs, with parameter adjustments made between runs. Other problems may require repetitive or iterative operations with the results displayed on an oscilloscope.

To run a problem that is strictly under operator control (an unautomated program), establish the integrator initial conditions by placing the computer in IC. (Depress ANALOG MODE – IC pushbutton.) Then, select the Logic Run and Analog Operate modes (depress LOGIC MODE – RUN and ANALOG MODE – OP pushbuttons). At the end of the run, select the analog Initial Condition or Hold Mode (depress ANALOG MODE – IC or H) and the logic Stop mode (depress LOGIC MODE – STP). If subsequent runs are to be made, update problem parameters as required and repeat the above procedure.

To run an automatic rep-op or iterative problem, simply initialize the analog and logic portions of the program and then depress PP. With normal TIMER patching, (Chapter 29) this will start the program run. After the run is completed, or the desired results are attained, select IC and STP.

3.5 RECORDING PROCEDURES

3.5.1 STATIC READOUT

Static readout of component outputs and parameters is performed using the digital voltmeter (DVM) and/or an external monitoring device. Selection of components for readout is accomplished by patching and by using the signal selector system (Chapter 8). Static readout is performed during static test procedures and is an integral part of the program checkout.

3.5.2 DYNAMIC RECORDING AND MONITORING

During a problem run, the problem solution may be observed and/or permanently recorded on any one of several devices. These include the DVM, and the Models 1110, 1130 and 1140 X-Y Plotter, and the 0.34.0035-1 Display Scope. The DVM may be used in conjunction with the signal selector system, or the MTR-IN terminations. The remaining devices are external to the MiniAC and are selected by patching at the DISPLAY terminations in the patch panel control field. Their selection is an integral part of the overall program, and is entirely dependent on the readout/recording requirement. Patching and setup of peripheral devices, is given in Chapters 9 and 10 of the Users Guide.

3.6 AMPLIFIER BALANCING

Generally amplifier balancing is considered a maintenance function and should be scheduled accordingly.

In some problems where certain parameters are critical, the operator/programmer may want to ensure that specific amplifiers are balanced as accurately as possible. The following procedures are provided to permit the operator/programmer to occasionally check and balance an amplifier, or a small group of amplifiers on an arbitrary basis.

1. Open the setup switch/overload indicator panel (Figure 2.7) to gain access to the unit to be balanced.
2. Using SELECTOR-1, address the module for readout on the DVM.
3. Depress and hold the BAL switch (bottommost pushbutton on the unit) and observe the DVM for an indication ±0.000, ±10.
4. If the above requirement is not met, adjust the balance pot for the reading specified in Step 3.

CAUTION

The Σ/HG cards have more than one potentiometer accessible for adjustment. In this case, the bottommost pot is the balance pot. The other adjustments (as well as adjustments internal to any card) are strictly for maintenance purposes and must not be tampered with. Failure to observe this precaution will result in erroneous solutions and may require maintenance downtime to correct.

3.7 USING THE PROGRAM SHEET

Figure 3.1 is a sample of the Program Sheets used with the MiniAC. This sheet provides a means of documenting the various program parameters and device assignments and greatly simplifies problem setup. The Program Sheet should become a part of permanent problem documentation along with the program diagram and any recorded solutions.

MATHEMATICAL FUNCTIONS:

This section of the program sheet is devoted to the analog mathematical elements.

The numbers in the left-hand column correspond to the patch panel address system (Paragraph 2.6). The symbols in the function column define the device function and corresponding setup switch positions (Paragraph 2.11). For example, device 11 can be used as a summer (Σ) or as an integrator ($\int$). When documenting the problem, simply place a circle around the function each device is to perform. This provides a record of device assignments and is most useful during problem setup.

The NOTES column is used to enter the output variable for each element used in the problem solution. Any special operating or programming considerations that must be observed can also be entered here.

The column designated OUTPUT is reserved for entering the output value measured on the DVM during the IC mode Static Check. Output values should be compared against any calculated values entered on the program diagram. Be sure to enter the proper sign (+ or −).

COEFFICIENT:

The coefficient portion of the program sheet is similar to the Mathematic Functions section. The numbers correspond to the potentiometer address, and the notes column is used to enter the parameter represented by each pot coefficient. The pot coefficient (from the program diagram) is entered in the SETTING column. When performing IC Mode Static Check, enter the pot output measurement in the OUTPUT column. This should be compared against the calculated value.

VFG:

This section of the program sheet is tabularized for setup of the Variable Function Generators (Chapters 7 and 25). The left-hand portion of the table is related to VFG36 and the right-hand portion to VFG37. The columns designated X and Y are used to enter the pairs of values prior to setting up the VFG. The shaded areas in these columns indicate that pairs of values *are not* entered when the VFG's are used as a combined unit in the 20PT mode. As a part of problem documentation and for ease of VFG setup, always encircle the GAIN setting of the VFG.

GATES (1-10):

The box under each GATE number is used to enter the Static Check output (Logic 1 or 0) of each gate used in the problem. This provides a record of the gate assignments and indicates the status indicator(s) that should be observed during problem checkout.

FF, PB, and COMP:

Place a circle around the number that corresponds to any general purpose flip-flop (FF), pushbutton flip-flop (PB) and comparator (COMP) used in the problem.

CTR/TMR:

Enter the counter/timer setting in this block.

MAN SW:

Enter the analog mode corresponding to each position of any manual switch used in the problem.

RATE-F:

If the problem is to be operated at a fast time scale, encircle F.

TIMER:

Enter the timer control settings in this box.

EAI MiniAC™

PROGRAM _____ NAME _____

_____ DATE _____

MATHEMATICAL FUNCTION

	FUNCTION	NOTES	OUTPUT
11	Σ ∫		
12	Σ ∫		
13	Σ HG		
14	Σ TS		
15	M D Σ		
16	LOG EXP Σ		
21	Σ ∫		
22	Σ ∫		
23	Σ HG		
24	Σ TS		
25	M D Σ		
26	LOG EXP Σ		
31	Σ ∫		
32	Σ ∫		
33	Σ HG		
34	Σ TS		
35	M D Σ		
36	11 20 INV		
37	11 INV INV		

COEFFICIENT

	NOTES	SETTING	OUTPUT
11			
12			
13			
14			
15			
16			
21			
22			
23			
24			
25			
26			
31			
32			
33			
34			
35			
36			

VFG GAIN: 1 3 10 30

GAIN: 1 3 10 30

	X	Y	X	Y
0				
1				
2				
3				
4				
5				
6				
7				
8				
9				
10				

GATE: 1 2 3 4 5 6 7 8 9 10

FF: 1 2 3 4

CTR/TMR

PB: 1 2

MAN SW: 16 26 36

UP

OFF

DN

COMP: 13 23 33

RATE: F

TIMER: X

NOTES _____

M1665

PUBLICATION NO. 955.0108-0

Figure 3.1. Program Sheet

PART

PART II
USERS GUIDE
(FOR THE INFREQUENT USER OR TRAINEE)

4.1 USERS GUIDE

The following chapters (5 through 12) are oriented toward the infrequent user or trainee. This portion of the manual provides a guide to permit you to patch and run your problem on a step-by-step basis. It is assumed that you have a scaled analog computer diagram (paper program); and you are now ready to use the MiniAC. Naturally, the first step is to patch your program. Chapter 5 acts as a translator to permit you to readily relate the symbols on the diagram to components on the patch panel and their set-up switches, if applicable. This chapter is organized by function.

The next logical steps are to:

1. Set coefficient values, Chapter 6.
2. Setup arbitrary functions (if required) Chapter 7.
3. Static check the patched program, Chapter 8.
4. Connect the X-Y Plotter, Chapter 9; or oscilloscope, Chapter 10.

If logical operations are required, refer next to Chapter 11. Finally, Chapter 12 fully describes a typical iterative analog computation problem.

4.2 GLOSSARY OF TERMS AND COMPUTING ELEMENTS

As an aid to the infrequent user or trainee, this introduction to the users guide includes a glossary of general terms and a glossary of computing elements. It is recommended that the first-time user review these glossaries before continuing.

GLOSSARY OF GENERAL TERMS

1. Analog Computer	A system of electronic building blocks that perform certain mathematical functions (i.e., if the input to one such block is x, then the output is y = f(x)), and are interconnected with wires by an operator to conform to an electric circuit block diagram. The result is an electronic model or simulation of some physical system, in which each voltage on the analog computer (a computer variable) is proportional to a variable in the physical system (a physical variable).
2. Simulation	The representation of one system by another, usually employing analogy.
3. Analog Program	A diagram to interconnect the mathematical and logic elements (electronic building blocks) which includes information on *scaling, potentiometer settings,* and *static check* values.
4. Analog Diagram	The portion of the program indicating the interconnection of mathematical elements, this diagram may or may not be scaled.
5. Logic Diagram	The portion of the program indicating the interconnection of logical elements.
6. Patch Panel	A removable panel that is mounted on the operator side of the computer. The front is labeled to show the input, output and control terminals of each of the building blocks. The rear of the patch panel mates with the actual blocks. By placing wires (called "patch cords") from terminal to terminal in accord with an *analog program* the operator "patches his program". The program is executed by placing the panel on the computer and performing *mode control* operation, and is stored by physically storing the patched patch panel.

7. Scaling, Amplitude	A procedure by which a programmer adjusts the amplitude of problem variables to fit within the *reference* range of the computer.
8. Scaling, Time	A procedure by which a programmer adjusts the relation between the time for an experiment on the physical system (problem time) and the time to perform an analogous experiment on the simulator (computer time).
9. β	The ratio between computer time, τ, and physical system time, t. (See *scaling, time)* $\tau = \beta t$.
10. Analog Signal	A time-varying voltage on the computer which may be any value between *−Reference* and *+Reference*.
11. Logic Signal	A time-varying voltage which can have one of two values at any instant of time. These values are called "1" and "0".
12. Reference	The maximum ± value for a computer variable. (+Reference = +1.0 *machine unit*, −Reference = −1.0 *machine unit.*)
13. Coefficient	The combination of problem constant, parameter and scaling values which determine a *potentiometer* setting; a multiplying factor, usually less than 1.
14. Gain	A multiplying factor, usually greater than 1.
15. Potentiometer (Pot)	An electrical device for multiplying an *analog signal voltage* by a constant (see *coefficient*).
16. Static Check	A means for debugging an analog computer program including patching performed for the checking.
17. Program Check	The portion of the *static check* which compares the computer program on paper with the mathematical statement of the original problem.
18. Circuit Check	The portion of the *static check* which compares the state of the computer simulation at a single instant of time with what the state of the orginal system would be at the same instant.
19. Mode	The state of the analog computer mathematical and logic elements. Mathematical elements can be in Operate, Hold, Initial Condition or Set Pot; while logic elements can be in Run, Stop, or Clear.
20. Mode Control	The process or act of selecting which mode the computing elements shall be in.
21. Repetitive Operation (Rep-Op)	The method of operation in which the computer cycles between modes automatically, with all parameters held constant. Usually used when an oscilloscope display is desired.
22. Iterative Analog Computation	That process of operating an analog computer in which: modes are controlled by a logic program and; one or more parameters are changed each cycle, so that many different solutions are produced. Compare to *repetitive operation*).
23. Logic Clock	A timing signal that synchronizes the execution of a logic program.
24. Function Generator (FG)	A mathematical element, the output of which is a function of the input except that the function is *not* integration or summation. FG's are usually "fixed" to function such as multiplication, exponentiation, etc.

25. Variable Function Generator (VFG) — A mathematical element, the output of which is a function of the input. The function is designated by the programmer and the VFG adjusted by the operator to achieve the desired mathematical result. VFG's are usually used when empirical data are introduced into a simulation.

26. Boolean Algebra — An algebra which deals with binary-valued variables, such as logic signals.

27. Bit — A binary digit, such as 1 or 0.

GLOSSARY OF COMPUTING ELEMENTS

1. Integrator ($\int$) — A mathematic element which accepts one or more analog signals and produces an analog output signal equal to the negative of the integral of the sum of the inputs. Several of the inputs can be multiplied by a factor of 10, if desired. Further, the initial value of the output can be programmed into the integrator.

2. Summer (Σ) — A mathematical element which accepts two or more analog signals and produces an analog output signal equal to the negative of the sum of the inputs, some of which may be multiplied by 10.

3. Track-Store (TS) — A mathematical element which functions a "sample and hold" storage unit under logic control. It accepts one or more analog signal and produces an analog output equal to the negative of the sum of the inputs, some of which may be multiplied by 10. The output can be placed in Hold at any time and will store its present value. Further, the initial value of the output can be programmed into the unit.

4. Inverter — A mathematical element which accepts one analog signal and produces an analog signal equal to the negative of the input; a logic element which performs the NOT function (see *GATE*).

5. Multiplier (Mult) — A mathematical element of the function generator type which produces the product of the two analog input signals.

6. Log/Exp — A mathematical element of the function generator type which can produce an analog output proportional to either; the natural logarithm of the input, or the antilogarithm of the input (e^{-10x}).

7. Comparator (Comp) — An analog to logic element which accepts two or more analog signals and produces a logic signal whose value depends on whether the sum of the input is positive or negative.

8. Gate — A logic element which performs one of the Boolean functions, AND, OR, or NOT.

9. Flip-Flop (FF) — A logic element which is a binary memory or storage device. The FF is the basic building block for synchronized logical programs and stores one bit of data.

10. Register — A logic element which can store more than one bit of data.

11. Counter — A logic element which stores data pertaining to the number of occurrences of an event.

12. Timer — A logic element which produces one or more logic signals which are 1 for a certain length of time and which are 0 for a certain length of time. At least one of the time intervals is usually adjustable.

CHAPTER 5

PATCHING AND SETTING UP MATHEMATICAL FUNCTIONS

5.1 INTRODUCTION

This chapter is devoted to presenting the various mathematical functions that each analog component in the MiniAC is capable of performing. The following paragraphs describe how each component and facility may be used on a functional basis starting with the mathematical function of inversion and proceeding through such functions as summation, integration, multiplication, etc. Included with each basic mathematical function is an illustration that shows the basic program symbol, patching, and setup requirements. The program symbols and patching requirements for each function are shown with unity scaling (machine units) rather than voltage values. A machine unit (MU=±1.0) is equal to computer reference (±10 volts). Therefore, 0.1 MU is equal to 1.0 volt. Computer reference and machine units are described in Paragraph 5.10.

5.2 INVERSION

There are six analog devices in the MiniAC that can perform the mathematical function of inversion (multiplication by −1). Because a multipurpose device can only be used for a single function in any given problem, some care must be used when choosing a device for use as an inverter. Careless or hap-hazard choice may result in an inadequate number of devices to perform some other mathematical function. To avoid waste of computational power, the paper program should be studied thoroughly before assigning the devices required in the problem. Figure 5.1 illustrates the program symbol, the basic patching, and gives the set-up procedure for each device that may be used to perform inversion.

5.3 SUMMATION

The algebraic sum of two or more variables is achieved by means of summers. There are up to five devices that may be used to obtain the function of summation. In fact, all devices that can be used for inversion (Paragraph 5.2) except the VFG inverter, can also sum. The program symbol, the basic patching, and the set-up for each device that may be used to perform summation is illustrated and described in Figure 5.2.

NOTE

The summer/high-gain, summer/track-store, and the summer/integrator, all have the same symbol and patching arrangement, and are therefore illustrated as one.

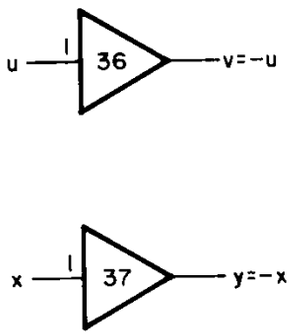
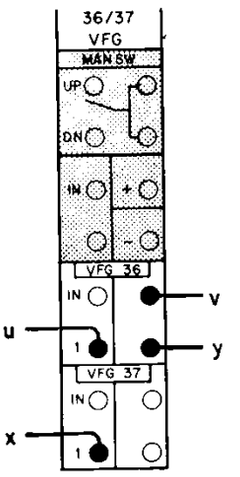
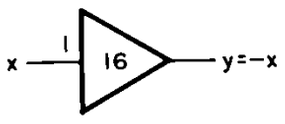
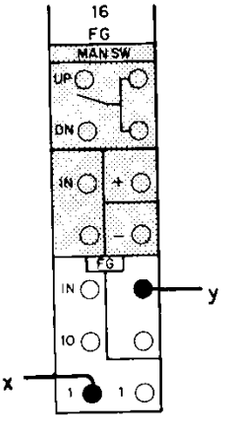
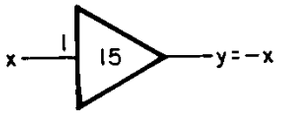
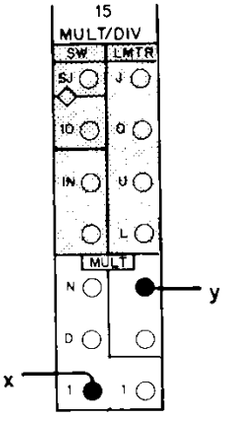
DEVICE ADDRESS	SYMBOL	PATCHING	SET – UP
VFG 36,37			OPERATION CONTROLLED BY MODE SELECTOR SWITCH BEHIND VFG PANEL. WHEN AT INV BOTH 36 AND 37 ARE AVAILABLE AS INVERTERS. IF VFG 20 PT MODE IS SELECTED, 37 IS AVAILABLE AS AN INVERTER
FG 16,26			FG/Σ SWITCH SET. WHEN PATCHING, EITHER TERMINATION DESIGNATED I MAY BE USED AS THE x INPUT.
MULT 15,25,35			MD/Σ SWITCH SET. WHEN PATCHING, EITHER TERMINATION DESIGNATED I MAY BE USED AS THE x INPUT.

Figure 5.1. Inversion: Patching and Setup

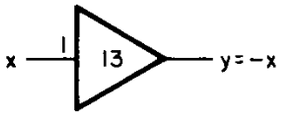
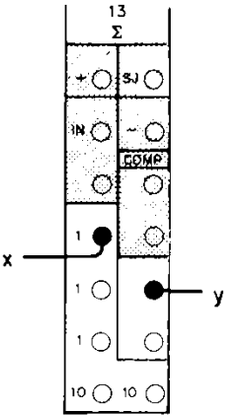
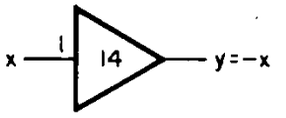
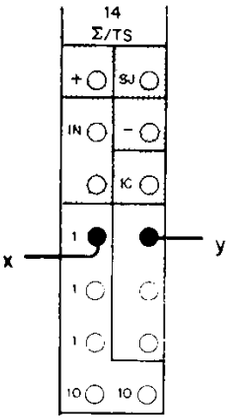
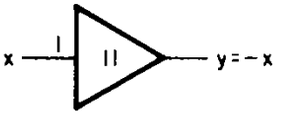
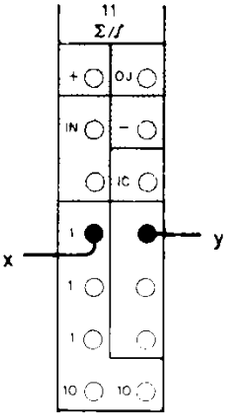
DEVICE AND ADDRESS	SYMBOL	PATCHING	SET–UP
Σ 13,23,33			Σ/HG SWITCH RELEASED. WHEN PATCHING, ANY ONE TERMINATION DESIGNATED I MAY BE USED AS THE x INPUT.
Σ/TS 14,24,34			Σ/TS SWITCH RELEASED. WHEN PATCHING, ANY ONE TERMINATION DESIGNATED I MAY BE USED AS THE x INPUT.
Σ/∫ 11,12 21,22 31,32			Σ/∫ SWITCH RELEASED. WHEN PATCHING, ANY ONE TERMINATION DESIGNATED I MAY BE USED AS THE x INPUT.

Figure 5.1. Inversion: Patching and Setup (Cont)

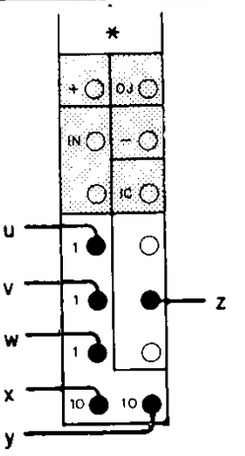
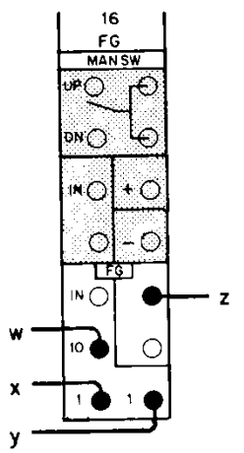
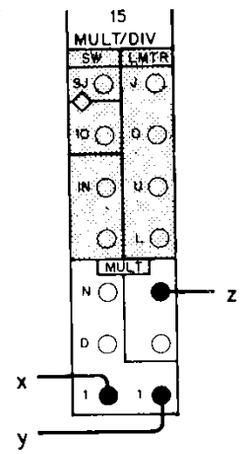
DEVICE ADDRESS	SYMBOL	PATCHING	SET–UP
Σ 13,23,33 Σ/TS 14,24,34 Σ/∫ 11,12 21,22 31,32	u 1 v 1 w 1 x 10 y 10 N DEVICE ADDRESS $z=-(u+v+w+10x+10y)$ NOTE: THESE SUMMERS HAVE FIVE INPUTS. TWO WITH. FACTORS OF 10.	 * + OJ IN − IC u 1 v 1 w 1 x 10 10 y z * = Σ, Σ/TS, OR Σ/∫	CORRESPONDING Σ/HG, Σ/TS, OR Σ/∫ SWITCH RELEACED.
FG 16,26	w 10 x 1 y 1 16 $z=-(10w+x+y)$ NOTE: THIS SUMMER HAS THREE INPUTS WITH ONLY ONE FACTOR OF 10.	 16 FG MAN SW UP DN IN + − FG IN w 10 x 1 1 y z	FG/Σ SWITCH SET.
MULT/DIV 15,25,35	x 1 y 1 15 $z=-(x+y)$ NOTE: THIS SUMMER IS MORE LIMITED THEN THOSE ABOVE. IT HAS ONLY TWO INPUTS AND NO FACTORS OF 10.	 15 MULT/DIV SW LMTR SJ J 10 0 IN U L MULT N D x 1 1 y z	MD/Σ SWITCH SET.

Figure 5.2. Summation: Patching and Setup

5.4 INTEGRATION

The major device that distinguishes the analog computer from other computers is its ability to perform integration with respect to time. The MiniAC has one device, the $\Sigma/\int$ (Summer/Integrator) that performs this function. The $\Sigma/\int$ (devices 11, 12, 21, 22, 31, 32) may be used to produce the integral of the sum of several input variables. In addition to performing this function, the $\Sigma/\int$ can store (or hold) any value attained while operating, and can be started at an arbitrary initial condition. Refer to Paragraph 5.10.

Figure 5.3 illustrates and describes the program symbol, patching, and setup requirements for performing the function of integration.

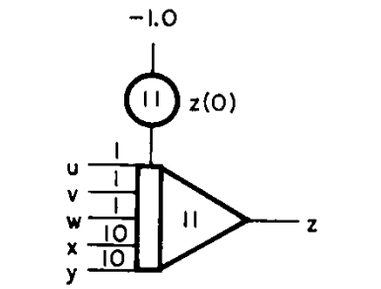
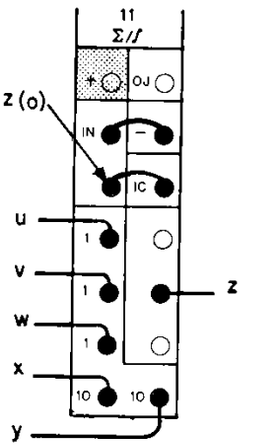
DEVICE ADDRESS	SYMBOL	PATCHING	SET-UP
$\Sigma/\int$ 11,12 21,22 31,32	 $z = -\int_0^t (u+v+w+10x+10y)dt + z(0)$ NOTE: POT USED TO PROVIDE INTERGRATOR WITH INITIAL CONDITION (IC) SEE PARAGRAPH 5.10.		Σ/∫ SWITCH SET. SET POT 11, z(0) TO DESIRED IC NOTE: REFER TO CHAPTER 14 FOR DETAILES CONCERNING MODE AND TIME SCALE CONTROL.

Figure 5.3. Integration (With Initial Condition): Patching and Setup

5.5 MULTIPLICATION AND DIVISION

5.5.1 INTRODUCTION

The process of multiplication (and division) on an analog computer is handled in one of two ways, depending on whether it is desired to multiply two variables (XY) or a variable and a constant (kz). The following paragraphs describe these methods of multiplication, and the mathematical function of division.

5.5.2 PRODUCT OF VARIABLE AND CONSTANT

The MiniAC has eighteen coefficient potentiometers (pots). Six pots are terminated in each analog field. Each pot takes on the address of the patching module in which it is terminated and is characterized by a single input and a single output. The program symbol and patching requirements for pot 12 (P12) are illustrated in Figure 5.4. The coefficient k is restricted to the range $0 < k < 1$; or k must be greater than zero and less than 1.0 MU. Setting the pot coefficient value is described in Chapter 6.

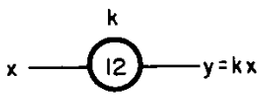
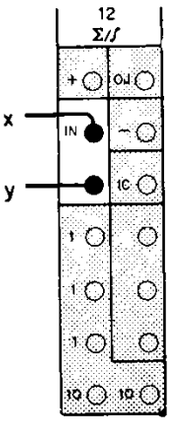
DEVICE ADDRESS	SYMBOL	PATCHING	SET–UP
P 11–16 21–26 31–36			PLACE COMPUTER, IN SP DEPRESS. SWITCH ADJACENT TO POT AND SET POT SO THAT DVM INDICATES VALUE OF k. THE VALUE OF k DEPENDS ON POT LOADING. THEREFORE, y MUST BE PATCHED BEFORE SETTING POT. REFER TO CHAPTER 6 FOR DETAILED PROCEDURE.

Figure 5.4. Multiplication of Variable and Constant: Patching and Setup

5.5.3 PRODUCT OF TWO VARIABLES

The product or ratio of two variables requires the devices designated MULT/DIV (addresses 15, 25, 35). Each MULT/DIV unit has two setup switches that must be released when using the device as a multiplier. Figure 5.5 illustrates the program symbol, the basic patching, and the setup requirements for multiplication of two variables.

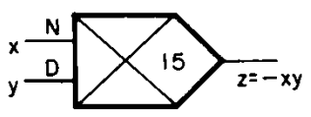
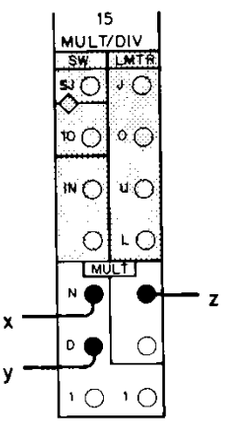
DEVICE ADDRESS	SYMBOL	PATCHING	SET–UP
MULT/DIV 15,25,35	 NOTE: x AND y MUST NOT BE PATCHED FROM THE OUTPUT OF A POT N AND D (x AND y) ARE INTERCHANGEABLE.		M/D SWITCH RELEASED MD/Σ SWITCH RELEASED

Figure 5.5. Multiplication of Two Variables: Patching and Setup

5.5.4 *DIVISION*

The MULT/DIV device (15, 25, 35) is used for division by setting the M/D switch and releasing the MD/Σ switch. Refer to Figure 5.6.

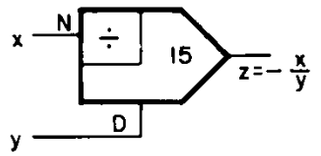
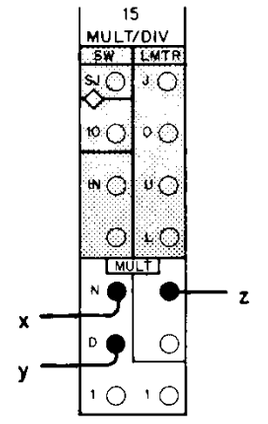
DEVICE ADDRESS	SYMBOL	PATCHING	SET-UP
MULT/DIV 15,25,35	 NOTES: 1. N=NUMERATOR D=DENOMERATOR 2. x MAY BE A POT OUTPUT. 3. y MUST NOT BE A POT OUTPUT. 4. ~~x~~ |X| ≤ Y 5. y MUST BE > 0		M/D SWITCH SET MD/Σ SWITCH RELEASED

Figure 5.6. Division: Patching and Setup

5.6 SQUARING AND SQUARE ROOT EXTRACTION

In addition to multiplication and division, the MULT/DIV devices (15, 25, 35) can be used to perform functions of squaring and square root extraction. Since $X^2 = X \cdot X$, it should be apparent that the square of a variable is produced by multiplying X by itself (Paragraph 4.5.3). To obtain X^2, use the same setup given in Figure 5.4, but patch X to both the N and D inputs.

Obtaining the square root of X is somewhat more complicated as it requires the use of a summer as well as the MULT/DIV device. The requirements for square root extraction are shown in Figure 5.7.

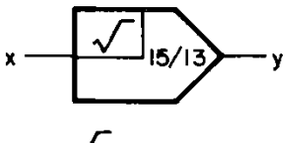
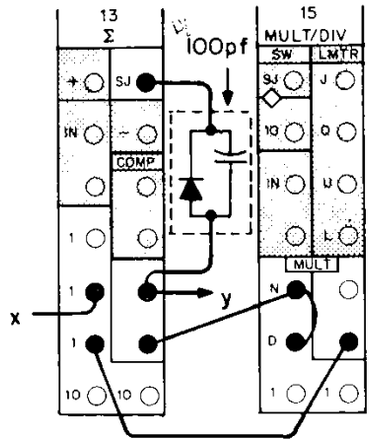
DEVICE ADDRESS	SYMBOL AND CONSIDERATIONS	PATCHING	SET-UP
Σ 13 MULT/DIV 15	 $y = -\sqrt{x}$ $x \geq 0$		Σ/HG SWITCH SET M/D SWITCH RELEASED MD/Σ SWITCH RELEASED NOTE: SPECIAL DIODE CAPACITOR NETWORK CAN BE MANUFACTURED LOCALLY USING A STANDARD DIODE AND A 100pf CAPACITOR. MODIFY PATCH CORD AND INSTALL NETWORK AS SHOWN.

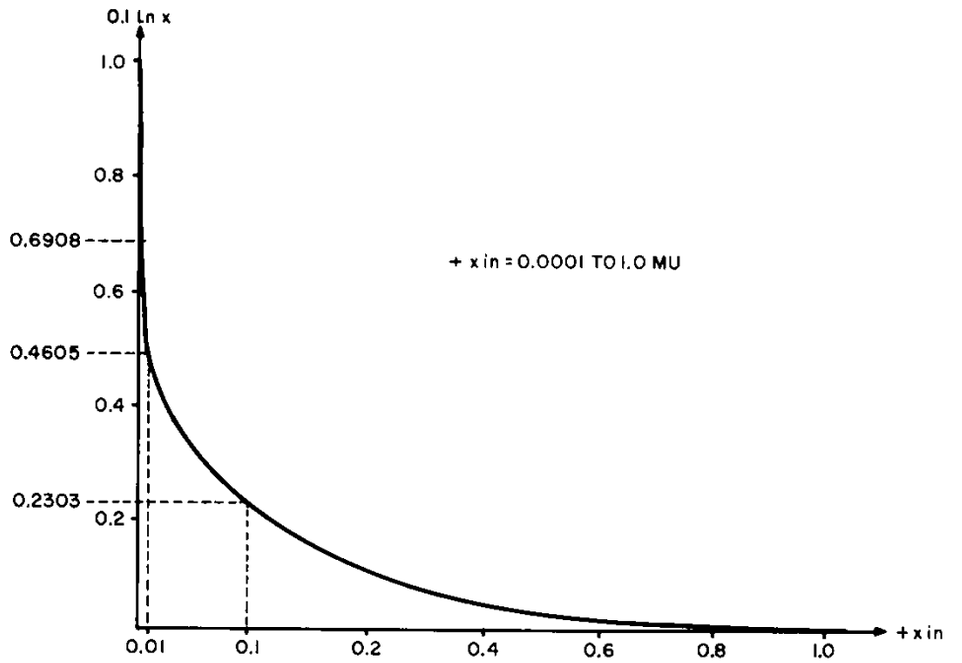
Figure 5.7. Square Root: Patching and Setup

The curve indicates that a fixed *scale-factor*, 0.1, has been applied to the device so that it can be used over 4 decades of x (realizing that 0.1 ln 0.0001 = 0.9211). Therefore, some care must be taken in scaling when lnx and not 0.1 lnx is required in a program. Patching and setup is straight forward as illustrated.

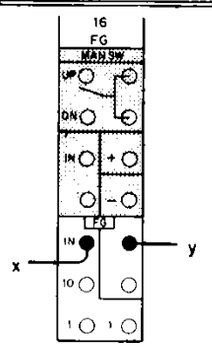
5.7 LOGARITHMS/EXPONENTIAL FUNCTIONS (FG Devices 16, 26)

5.7.1 NATURAL LOGARITHM (lnx)

The natural log function, lnx, can be generated on the MINIAC subject to certain restrictions. One of these restrictions has to do with amplitude scaling, in that the input/output relation of the device is constrained to be that described in Figure 5.8.



a. 0.1 Ln x CHARACTERISTICS

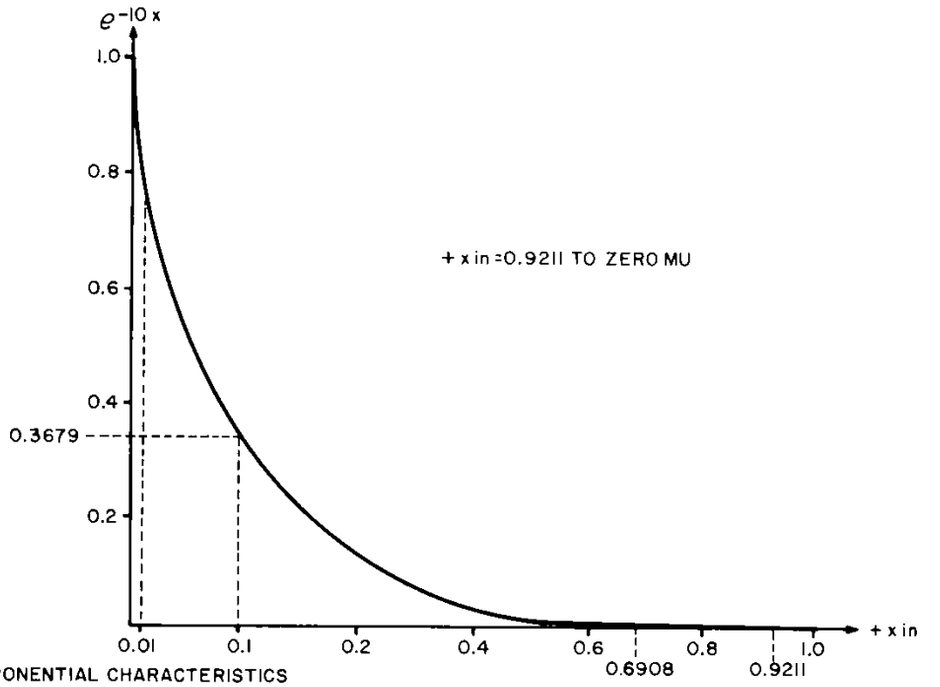
DEVICE ADDRESS	SYMBOL AND CONSIDERATIONS	PATCHING	SET-UP
FG 16,26	x → Ln 16 → y=0.1 Lnx +0.0001 ≤ x ≤ +1.000		LOG/EXP SWITCH RELEASED FG/Σ SWITCH RELEASED

b. PATCHING AND SETUP

Figure 5.8. Natural Logarithm

5.7.2 EXPONENTIAL (e^{-10x})

The exponential function often found in physical problems as e^{-10x} or exp (−10x), is easily generated using the FG device found in locations 16 and 26. A fixed scale factor is also applied to this device as indicated in Figure 5.9. As in the case of generating the natural logarithm, patching and setup is straight forward.



a. EXPONENTIAL CHARACTERISTICS

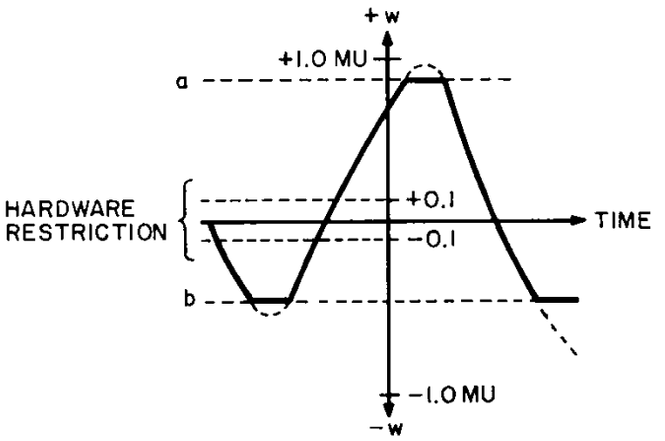
DEVICE ADDRESS	SYMBOL AND CONSIDERATIONS	PATCHING	SET-UP
FG 16,26	x → EXP 16 → y = e^{-10x}	16 FG MAN SW UP DN IN + − FG IN 10 1 x y	LOG/EXP SWITCH SET FG/Σ SWITCH RELEASED

b. PATCHING AND SETUP

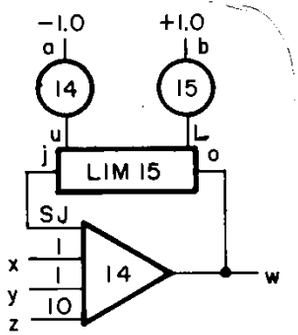
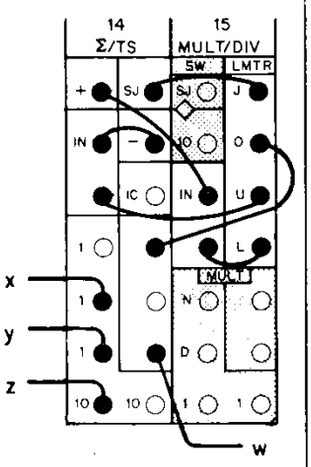
Figure 5.9. Exponential Function

5.8 CONSTRAINTS (LIMITS)

Many functions that occur in engineering and scientific problems must be constrained or limited to some maximum value. Suppose a variable (w) is constrained in a problem to be in the range $b \leq w \leq a$; where $b < -0.1$ MU and $a > +0.1$ MU. The variable w may be the output of a summer or an integrator. As can be seen in Figure 5.10, the output variable w is limited to the values of a and b as determined by the coefficient pot settings. This ± limiting is especially useful with integrators to prevent overloading. An amplifier or integrator can be limited to any value from ±0.1 to 1.0 MU. Due to certain electrical restrictions, limits below ±0.1 machine units are not reliable on all units.



a. CONSTRAINT DIAGRAM (± LIMITING)

DEVICE ADDRESS	SYMBOL AND CONSIDERATIONS	PATCHING	SET-UP
LMTR 15,25,35	 $w = -(x+y+10z)$ u = UPPER LIMIT = a l = LOWER LIMIT = b NOTE: LIMITED DEVICE CAN BE ANY UNIT WHOSE SUMMING JUNCTION (SJ) IS AVIAL-ABLE AT THE PATCH PANEL; OR AN INTEGRATOR (OJ)		1. FORCE AMPLIFIER OUTPUT TO GO POSITIVE AND ADJUST UPPER LIMIT POT (a). 2. FORCE OUTPUT TO GO NEGATIVE AND ADJUST LOWER LIMIT POT (b). NOTE: OBSERVE POLARITIES APPLIED TO POTS. UPPER LIMIT = NEGATIVE. LOWER LIMIT = POSITIVE. REFER TO CHAPTER 22 FOR A MORE DETAILED PROCEDURE.

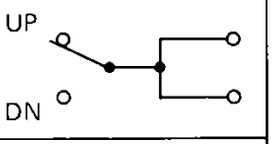
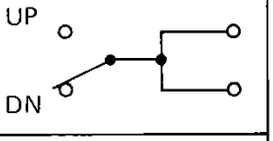
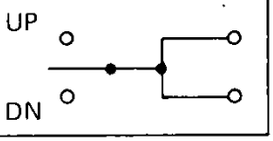
b. PATCHING AND SETUP

Figure 5.10. Constraints: Patching and Setup

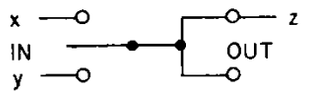
5.9 SWITCHING FUNCTIONS

5.9.1 MANUAL SWITCHING

Three manually-operated switches (16, 26, 36) are included in the fully-expanded computer, located physically in the upper left-hand corner of the analog control panel. They are labeled MAN SW. Each can be set in one of three physical positions: top fully depressed (flush with panel); bottom fully depressed (flush with panel); and centered (neither top nor bottom is flush with the panel). The switching functions corresponding to these positions are as follows:

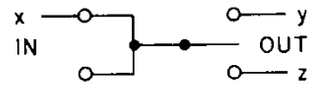
Position	Function	Diagram
Top In	UP connected to unmarked terminal.	
Bottom In	DN connected to unmarked terminal.	
Centered	Neither connected to unmarked terminal.	

A switch may be used to select between two functions:



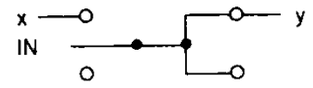
UP: z = x
CTR: z=0
DN: z = y

or to "feed" a signal to one of two destinations:



UP: y = x, z = 0
CTR: y AND z = 0
DN: z = x, y = 0

or in a simple "on-off" arrangement



UP: y = x
CTR: y = 0
DN: y = 0

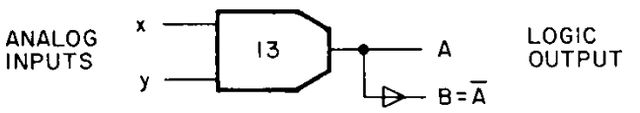
5.9.2 AUTOMATIC SWITCHING (Programmed Switch)

At times, it is not desirable or convenient for a switching function to depend on a human operator. For example, you may wish to close a switch at the instant a certain event occurs. This requires that the switching time be much less than human response time. Further, you may make several runs that require the switching time be repeatable. Under such conditions, manual switch operation is impractical: manual reactions are not fast enough and cannot achieve the necessary consistency. A fast acting *programmed switch* (operated by the computer program), is required.

5.9.2.1 General-Purpose Programmed Switch

The programmed switch is a very different device for electrical reasons and is not a 1:1 replacement for the manual switch. It normally consists of two devices: an electronic switch (SW15, 25, 35) and a comparator (COMP13, 23, 33).

The comparator, shown symbolically below,



accepts inputs from analog fields 1-3 and produces logic signals A and B. These logic signals differ from analog signals in a very fundamental manner: while an analog signal may be anything from –Reference to +Reference, a logic signal can only be one of two possible values. Much as we simplified things by calling Reference = 1.000, we call the two logic values (or states) ONE (1) and ZERO (0). The B signal from the comparator is the logical *complement* of A, i.e., $B = \overline{A}$ (B equals *NOT* A). The following "truth table" shows all possible combinations of A and B.

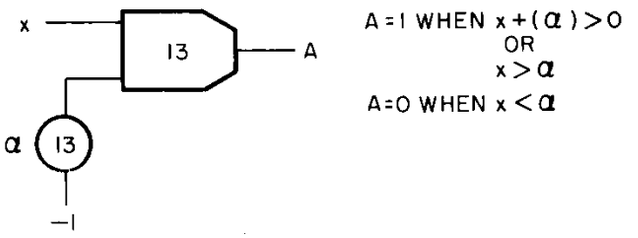
A	$B = \overline{A}$
0	1
1	0

What makes A = 0 or A = 1? The answer is, the analog inputs:

if $x + y > 0$, A = 1

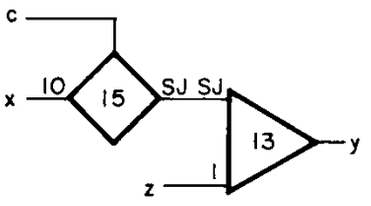
if $x + y < 0$, A = 0

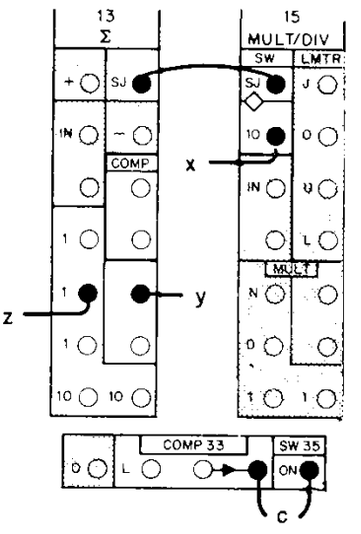
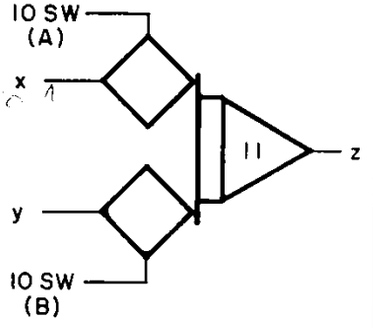
The case x + y = 0 is not considered because of the electrical limitations of the comparator. In the usual application, one of the inputs (say, y) is a constant or *bias*, and acts as the reference for switching.

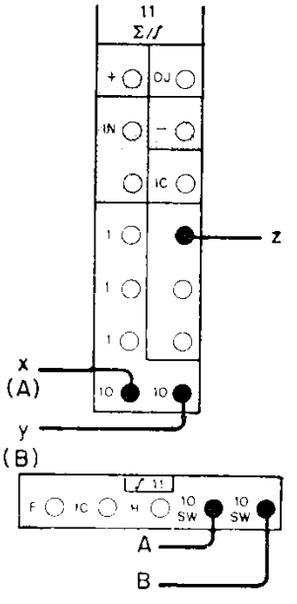


The electronic switch (SW) accepts a logic signal (ON) that closes the switch when ON = 1 and opens it when ON = 0. It must be noted here that SW is always used with one of the Σ devices (11-14, 21-24, and 31-34), because it is not a simple switch (as is MAN SW), but actually a *switched input* for a Σ device. Thus, the "10" is the input terminal of the switch. However, the SJ termination is not an output in the sense that an analog signal can be measured there, only in the sense that you must connect the SW-SJ termination to the SJ termination of a Σ/HG or Σ/TS (SJ to SJ); or in some rare instances to the 0J of a Σ/∫ (SJ to 0J). Figure 5.11 shows Σ 13 with a normal input (z) and a switched input (x). The output (y) depends on the state of the logic control signal (C). Note that the ON control termination for SW-15 is located in the CONTROL field, while the input (10) and pseudo-output (SJ) are in the first analog field. Also note that the output terminals of COMP-13 are adjacent to SW-15 ON which makes using COMP-13/SW-15 as a *programmed switch* convenient.

<table>
<tr><th>DEVICE ADDRESS</th><th>SYMBOL</th><th>PATCHING</th><th>SET-UP PROCEDURE</th></tr>
<tr>
<td>COMP
13,23,33

SW
15,25,35</td>
<td> 

C IS NORMALLY THE LOGIC OUTPUT OF A COMP. CAN BE ANY LOGIC SOURCE.
WHEN c=0, y=−z
WHEN c=1, y=−(10x+z)</td>
<td> 
</td>
<td>Σ/HG SWITCH RELEASED</td>
</tr>
<tr>
<td>ANALOG FIELDS
Σ/∫ AND CONTROL FIELD
∫−10 SW
11,12
21,22
31,32</td>
<td> 

10 SW(A) AND 10 SW(B)=LOGIC OUTPUT OF COMP OR ANY LOGIC SOURCE.
TRUTH TABLE
<table>
<tr><th>A</th><th>B</th><th>z</th></tr>
<tr><td>0</td><td>0</td><td>0</td></tr>
<tr><td>0</td><td>1</td><td>$-\int y\,dt$</td></tr>
<tr><td>1</td><td>0</td><td>$-\int x\,dt$</td></tr>
<tr><td>1</td><td>1</td><td>$-\int (x+y)\,dt$</td></tr>
</table></td>
<td> 
</td>
<td>Σ// SWITCH SET

NOTE:
THESE SWITCHES MAY ALSO BE USED WHEN THE UNIT IS OPERATED AS A SUMMER(Σ// SWITCH RELEASED).</td>
</tr>
</table>

Figure 5.11. Programmed Switches: Patching and Setup

5.9.2.2 The Σ/∫ Switches

It was mentioned in the preceding Paragraph that an SW is rarely used with a Σ/∫. And for good reason – each Σ/∫ already has an assigned pair of programmable X10 inputs. The X10 inputs on each Σ/∫ can be controlled by logic signals in the same manner as the general purpose switches, when the integrator is in the OPERATE mode or when it is used as a summer. The Σ/∫ switches are automatically disabled during the H and IC computer modes.

Figure 5.11 also shows the control area for an integrator and the integrator element itself to illustrate the controls for the two X10 switchable inputs. As shown, the left-hand 10-SW termination controls the left-hand X10 input, while the right-hand 10-SW controls the right-hand X10 input.

The absence of patching at 10-SW causes the switch to be ON, allowing the integrator to follow normal mode selection.

5.9.2.3 Using SW with $\Sigma/\int$

In the case when it is necessary to have more than two switched inputs on a $\Sigma/\int$ or a gain 100, one or more SW devices can be added by patching the SW-SJ to the 0J terminal of the $\Sigma/\int$. When patching SW-SJ to $\Sigma/\int$-0J there is an important consideration: the X10 input of the SW behaves as if it were a *X100* input.

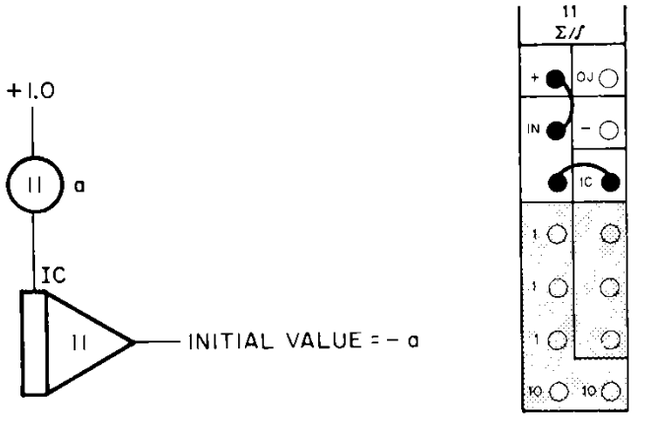
5.10 CONSTANT VALUES

In many problems, constant values are required. Constants are obtained from a source of constant signal level, known as Reference. The reference source is terminated at various locations on the patch panel as either positive 1.000 MU or negative 1.000 MU. Each patching module with an address ending in 1, 2, 3, 4 and 6 has both +Reference and −Reference terminations. See Figure 2.2. +Reference is color-coded *red* and −Reference, *orange*. A constant value of zero (or ground) is terminated in the black horizontal strip between analog field 3 and the logic field.

The assignment of the value ±1.000 to reference is entirely arbitrary; (±10 volts in the MiniAC), though quite sensible. Reference is actually a voltage that depends on the electrical design of the system. The MiniAC system operates within specifications when all signals lie between +Reference (+10.00 volts) and −Reference (−10.00 volts). Since this is not an operating concern, we can *normalize* everything by dividing the voltage value of Reference by 10. This implies that Reference is +1.000 machine units (unity) and that all signals are in the range −1 to +1. The effect of this is to make amplitude scaling easier and *independent* of the actual voltage value of Reference.

A constant other than reference (and less than unity) can be obtained by patching the output of a pot to the device requiring the constant and patching reference to the pot IN termination. After all patching is completed, the pot is then set to the desired coefficient value.

A sample constant of less than 1.0 MU to provide an initial condition for an integrator is achieved as shown below.



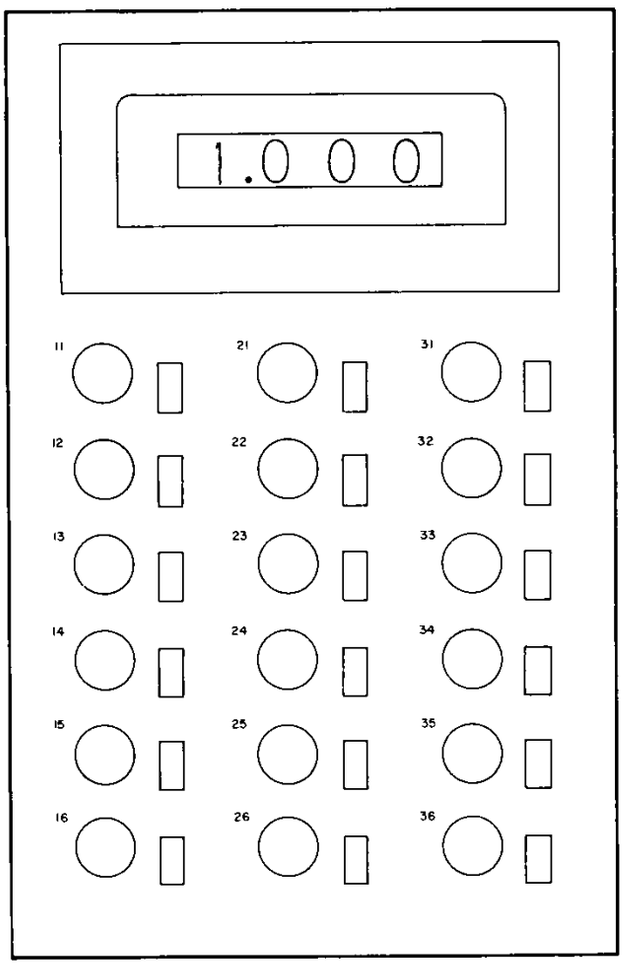


Figure 6.1. DVM and Coefficient Pot Controls

SETTING AND READING COEFFICIENT VALUES

6.1 INTRODUCTION

When a pot is used to introduce a parameter or a constant into a program, that pot must be set to the value required by that program. This chapter describes how this is done. We must first note that *all patching involving pots must be completed* before attempting to set the first pot. This point will be repeated to stress its importance.

6.2 THE DIGITAL VOLTMETER AND POT CONTROLS

The digital voltmeter (DVM) is the primary measuring device for constant values. Since a pot coefficient is a constant, we will use the DVM in our pot-setting procedure.

The DVM is located just to the right of the patch panel (Figure 6.1), and displays values to three significant figures to the right of the decimal point. Since the maximum reading of the DVM is ±1.000 and the maximum setting of a pot is 1.000, the DVM will provide direct readout of the coefficient value.

Beneath the DVM are the eighteen pots with their controls. Each pot has a rotary knob used for adjusting the coefficient and a momentary pushbutton switch used in the setting procedure which is detailed below.

6.3 PROCEDURE FOR SETTING A POT

The following table lists all of the steps for setting a pot to its required value, k, where $0 \leqslant k < 1$.

NOTE

It is not possible to set a pot to 1.000 because of certain electrical design requirements. If your program appears to require a pot with a setting of 1.000, replace that pot with a patch cord.

Step	Action	Note
1	Analog mode: SP	This is the Set Pot mode.
2	Patch panel engaged with *all* patching completed and all setup switches properly set.	*Important!*
3	Press pushbutton to the right of desired pot and hold.	This switches certain internal circuits so that the coefficient value is displayed on the DVM.
4	Rotate pot knob until desired coefficient is indicated by DVM.	
5	Release pushbutton	

6.4 PROCEDURE FOR READING POT VALUES

By "reading pot values", we can mean one of two things:

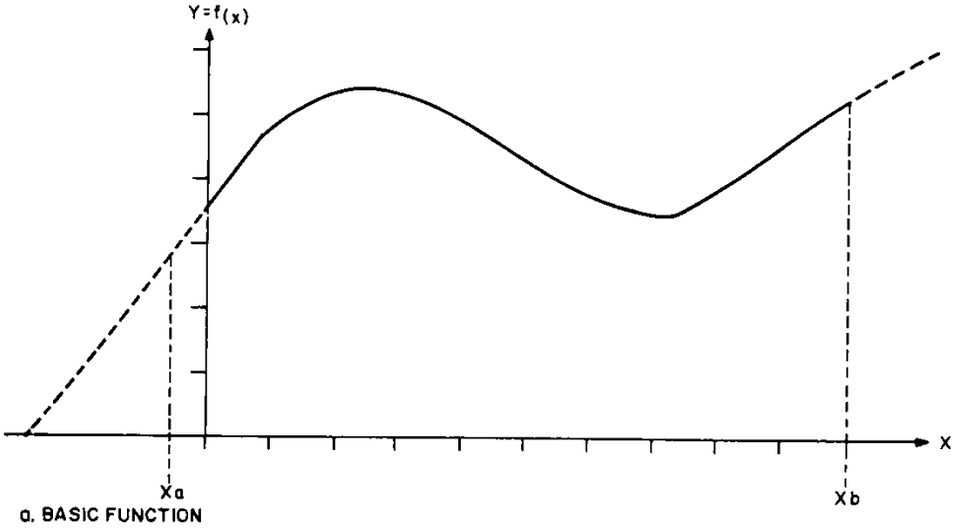
1. What is the coefficient setting?
2. What is the output of the pot?

The first question would arise in a simulation in which you had to vary a parameter until some desired response (of the program) was reached, at which point the pot setting must be known in order to calculate the parameter value. The procedure is simple:

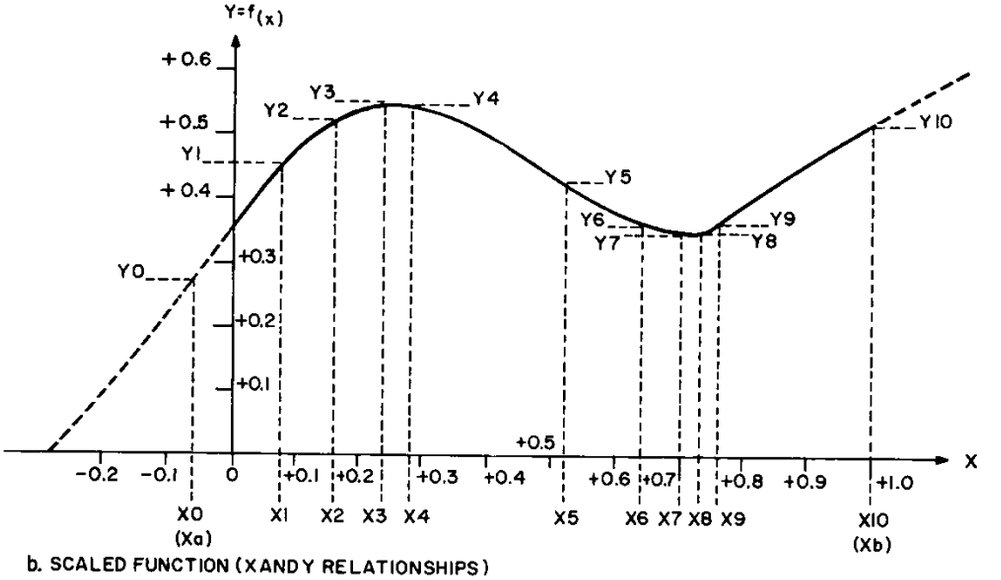
1. Analog mode: SP
2. Press pot pushbutton and read coefficient value on DVM.

The second question often arises during the check-out phase of programming (Chapter 8): The actual output of a pot (i.e., output = output times coefficient) is required for further calculations. The procedure is:

1. Analog mode: IC
2. Press pot pushbutton
3. The DVM now displays the sign and magnitude of the pot output.



a. BASIC FUNCTION



b. SCALED FUNCTION (X AND Y RELATIONSHIPS)

TABLE OF VALUES
(FROM PROGRAM SHEET)
VFG 36
GAIN

① 3	10	30

X	Y	
-0.060	+0.280	0
+0.080	+0.460	1
+0.160	+0.520	2
+0.240	+0.545	3
+0.280	+0.540	4
+0.520	+0.420	5
+0.640	+0.380	6
+0.700	+0.360	7
+0.730	+0.350	8
+0.760	+0.370	9
+1.000	+0.520	10

Figure 7.1. Sample Arbitrary Function

ARBITRARY FUNCTIONS FROM EMPIRICAL DATA

7.1 INTRODUCTION

In addition to the mathematical functions previously described, the engineering problem solver is often confronted with functions arising from empirical data. For example, an ecological study may include tabular data of pollution values measured at 50 points along a river. Since these data are to be used in the river simulation, a device that can be programmed with these data is needed. This device is the variable function generator (VFG).

7.2 VFG THEORY

The VFG operates on the assumption that a single-valued function can be approximated by a series of straight lines. This is one of the fundamental assumptions in nearly all non-linear analysis, and is used in essentially all computer methods for non-linear problem solving. Consider the function shown in Figure 7.1a. Suppose we are interested in the function between x_a and x_b. The question is, how do we best approximate the curve with straight-line segments, and how many segments are available? With a ten-segment VFG we can "eyeball" a straight-line function by increasing the density of segment ends (breakpoints) at places where curvature is great and reducing breakpoints where curvature is minimal.

The result of this procedure is a table of pairs of values (y_i, x_i) where y_i is the value of y when $x = x_i$. The MiniAC VFG accepts these value pairs to produce a straight line segment approximation to the curve.

If you already have tabular data (say, pollution versus distance from the plant), you may be tempted to use these as pairs of values for the VFG. *Don't!* Instead, plot your data, draw a smooth curve through the points, and then divide the curve into straight line segments (Figure 7.1b).

7.3 THE VFG (DEVICES 36 and 37)

The VFG is actually a pair of 10-segment devices that can be operated as independent 10 segment units, or combined to provide a greater number of segments. Combining the VFG's is useful in functions having several regions of curvature. As illustrated in Figure 7.2, the dual VFG control panel is divided into three distinct and separate areas. The segment and setup controls for VFG 36 are to the left of the panel, and those for VFG37 are to the right. The area at the bottom of the panel contains controls that are common to both VFG sections.

X0-X10 POTS:

The vertical row of potentiometers (designed X0 to X10) at the outer edge of each VFG section are used to set the values of X required to setup the function. The X0 and X10 pots select the initial and final values of X. The controls designated X1 through X9 select the breakpoints.

Y0-Y10 POTS:

The vertical row of potentiometers designed Y0 through Y10 on each VFG section are used to set the Y output or f_x. The Y0 and Y10 pots correspond to the initial value of f_x and the final values, respectively. The Y1 through Y9 pots select the value of f_x for the corresponding breakpoint or value of x.

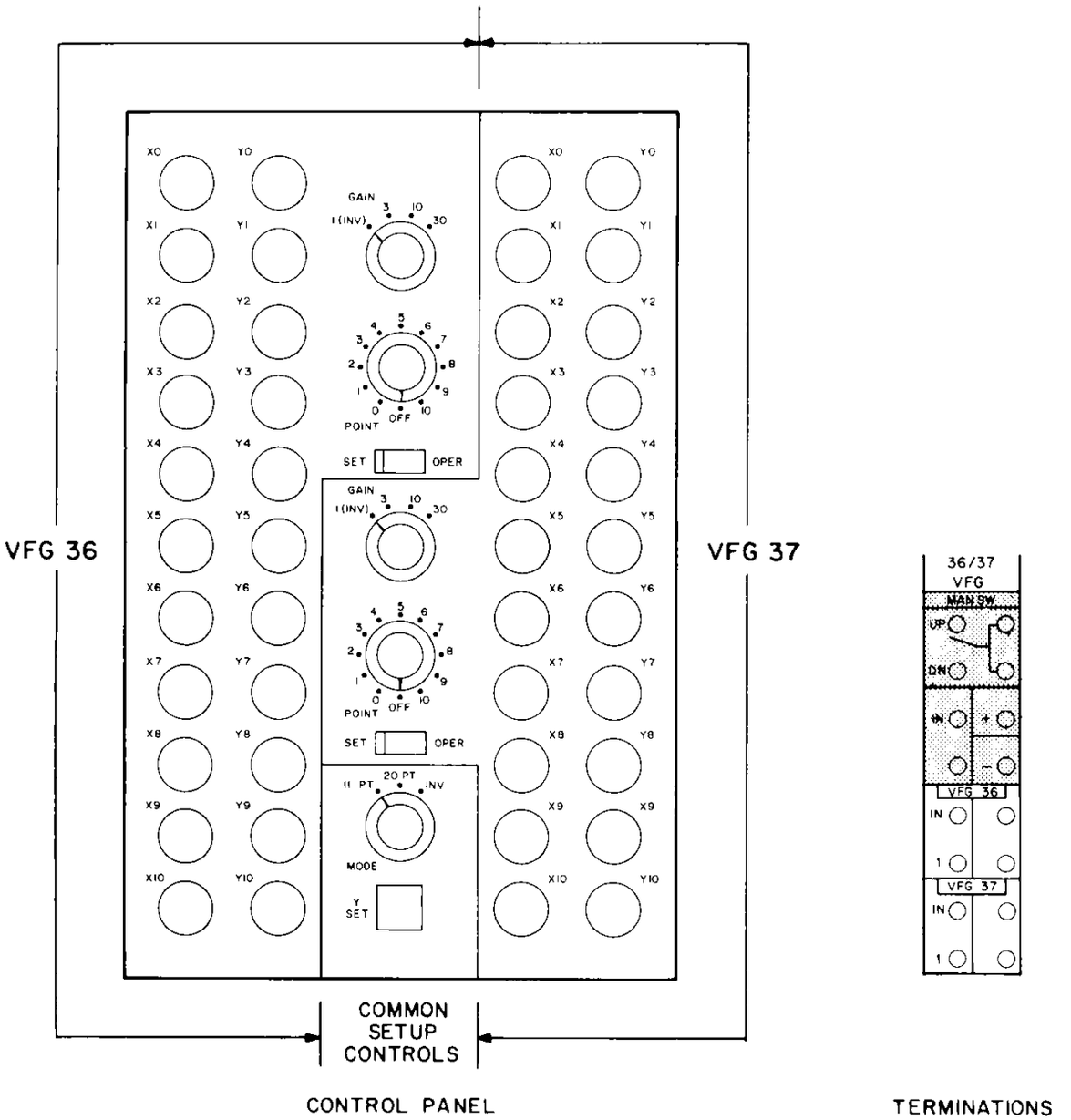


Figure 7.2. The VFG Controls

GAIN SELECTOR:

The GAIN selector (uppermost switch) on each VFG section controls the change in slope (gain) of the output amplifier. There are four gains (1, 3, 10 and 30) available. As larger values than 1 are selected steeper curves can be set. Since the GAIN control increases the sensitivity of the Y controls, large gains make accurate settings progressively more difficult. If the slope is changing more than 1.4 volts/volt, a gain greater than 1 is required. Always use the least possible gain setting. GAIN-1 must be selected when using the VFG in the INV (inverter) mode.

POINT SELECTOR:

The POINT selector (middle switch) in each VFG section is used during VFG setup to select the point being set. Positions 0-10 correspond to the X0-X10 and Y0-Y10 controls. The POINT switch must be placed in the OFF position after the function is setup and prior to placing the VFG in service.

SET/OPER SWITCH:

The SET/OPER switch (bottommost switch) in each VFG section is placed at SET during the VFG setup procedure. The VFG cannot be placed in service until the switch is set at the OPER position.

MODE SELECTOR:

The MODE selector is located in the central area between the VFG sections and is common to both. When set at 11PT each VFG section can be used as independent 11 point (10 segment) VFG's. When at the 20PT position, the VFG sections are combined to provide a single 20 point VFG. The combined output is taken from VFG36 and the output amplifier of VFG37 becomes a free inverter. When the MODE selector is set at INV, both output amplifiers become free inverters.

Y-SET PUSHBUTTON:

This momentary pushbutton switch is used during VFG setup while setting the Y0-Y10 potentiometers. Depressing this switch displays the value of Y (at the selected point) on the DVM.

NOTE

When the VFG is unused or is operated in the INV Mode, false overloads may be indicated. If a valid function is stored in the VFG, the possibility of false overloads may be eliminated by patching the IN terminations to ground. If an invalid function is stored, false overloads may be indicated even with the IN terminations grounded. If this occurs, set the X0 and Y0 pots at mid position and turn all other pots fully clockwise.

7.4 SET-UP PROCEDURE

7.4.1 11PT (10 SEGMENT) MODE

1. Prepare a table of values for each X and Y setting.
2. Select the computer IC Mode.
3. Using SELECTOR-1 (on Analog Control Panel) address the VFG (36 or 37) that is to be set up.
4. Remove the VFG cover panel and place SET-OPER switch at SET.
5. Set GAIN switch at 1 (unless you know that a larger gain is needed), and set the POINT switch at 0.

6. Rotate all X pots (X0 through X10) fully clockwise.
7. Adjust X0 until the desired value is indicated on the DVM.
8. Press the Y-SET pushbutton and adjust Y0 until the desired value is displayed on the DVM; then release Y-SET.
9. Repeat Steps 7 and 8 for POINT positions 1, 2, 3, . . . 10.

NOTE

If the Y value for any point cannot be attained, increase the GAIN setting to the next higher position. Then turn all X pots fully clockwise, and reset the entire function starting with Step 7.

10. After setting the last X,Y point it may be necessary to "touch up" the functions by repeating Steps 7 through 9.
11. When the final Y value is set, set POINT switch at OFF, place SET-OPER switch at OPER, and replace VFG cover panel.
12. If desired, check the function using the XY Plotter. Simply setup the program shown in Figure 7.3, and execute the plotting instructions given in Chapter 9.

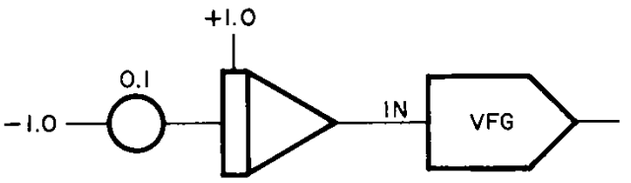


Figure 7.3. Program for Plotting VFG Output

7.4.2 20PT (19 SEGMENT) MODE

This mode provides increased accuracy through the use of more segments. Basically the function is set in the same manner as functions having ten segments or less, except as follows: the MODE switch is set at 20PT; the X10 control on VFG36 is not used; and the X0, Y0, Y1 and GAIN controls on VFG37 are not used. The GAIN-37 switch must be set at 1 if VFG37 is used as an inverter.

1. Prepare a table of values for each X and Y setting.
2. Select the computer IC Mode.
3. Using SELECTOR-1 (on Analog Control Panel) address VFG36.
4. Remove VFG cover panel and place both SET-OPER switches at SET.
5. Set both GAIN switches at 1 (unless you know a larger gain is required).
6. Set VFG36 POINT switch at 0 and VFG37 POINT switch at OFF.
7. Rotate X pots (X0 through X10) on both VFG's fully clockwise.
8. Set X0-VFG36 for desired value, as displayed on DVM.
9. Press Y-SET and adjust Y0 until the desired value is displayed on the DVM. Then release Y-SET.

10. Repeat Steps 8 and 9 for POINT-36 positions 1, 2, 3, . . . 9.
11. After completing the Y9 setting, set POINT-36 at OFF and POINT-37 at 1.
12. Set value of X for tenth breakpoint using X1-37.
13. Depress Y-SET and adjust Y10-36 for the desired Y value.
14. Continue to alternately set X and Y for the remaining POINT-37 positions (2-10) using the corresponding X2, X3 . . . X10 pots on VFG37.
15. After setting the last X,Y point it may be necessary to ',touch up'' the function by repeating Steps 8 through 14.
16. After setting Y10-37 set POINT-37 at OFF, and place both SET-OPER switches at OPER.
17. If desired, check the function using the XY plotter. See Figure 7.3

7.5 VFG PATCHING

Basic VFG patching the program symbol for each VFG mode of operation is illustrated in Figure 7.4.

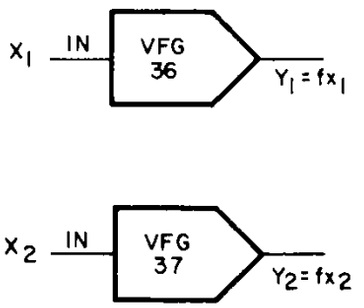
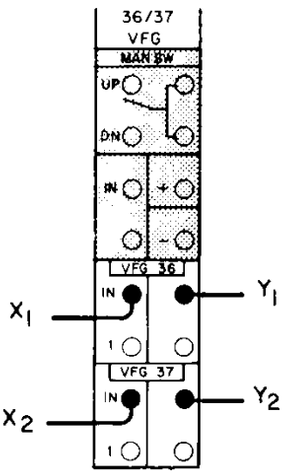
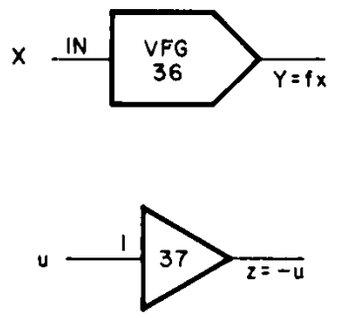
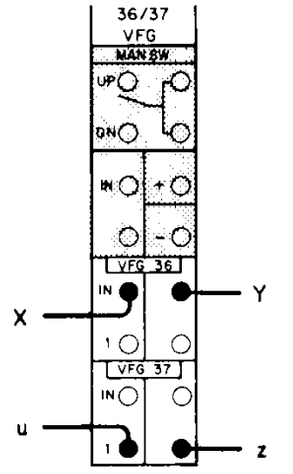
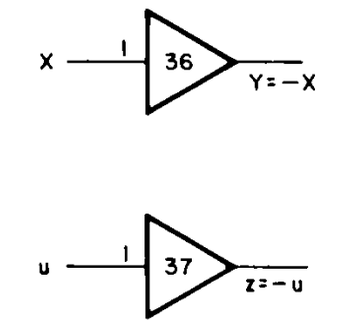
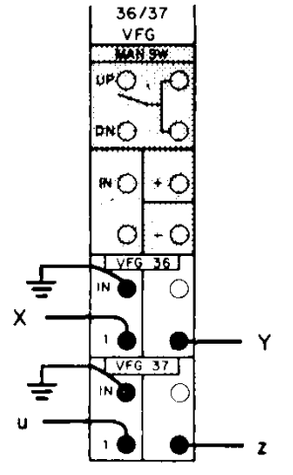
MODE	PROGRAM SYMBOL	PATCHING	OPERATIONAL SWITCH SETTINGS
11–PT			MODE: 11 PT SET–OPER (BOTH): OPER. POINT (BOTH): OFF GAIN (BOTH): A/R
20–PT			MODE: 20 PT SET–OPER (BOTH): OPER. POINT (BOTH): OFF GAIN: VFG 36 – A/R VFG 37 – 1 IF USED AS INVERTER
INV			MODE: INV SET–OPER (BOTH): OPER. POINT (BOTH): OFF GAIN (BOTH): 1 NOTE: VFG 36,37 IN PATCHED TO GROUND TO AVOID FALSE OVERLOAD

Figure 7.4. VFG Patching

8.1 INTRODUCTION

At this point, you should have an analog program patched on the MiniAC patch panel; all setup switches in the proper position; all pots are set to the proper values; and the VFG's set up. However, you are not ready to enter the operate (OP) mode, as the *static check* has not yet been performed. The following paragraphs describe the methods of static checking and documenting the problem.

8.2 STATIC CHECK THEORY

The purpose of the static check is to ensure that the patched program represents the mathematical statement of the system to be simulated. This implies correct patching, correct scaling, and correct operation of the computer electronic elements which produce the mathematical functions (Figure 8.1).

The process of checking is two-fold; there is a portion done on paper (program check) and a portion done on the computer (circuit check). The program check requires that an initial set of data (parameter values and initial conditions) be used to calculate the values of variables whose initial conditions are not specified, and that the calculations be made in two ways; using the original equations, and using the computer diagram. To make the program check, you must know all initial conditions and all pot settings. When the two methods yield equivalent answers, the paper program represents the original mathematical equations.

To make the circuit check, you must *measure* the output values of the computational devices that perform mathematical operations, and match these against the values calculated from the computer diagram and entered on the Program Sheet. If they agree, all is well *statically.*

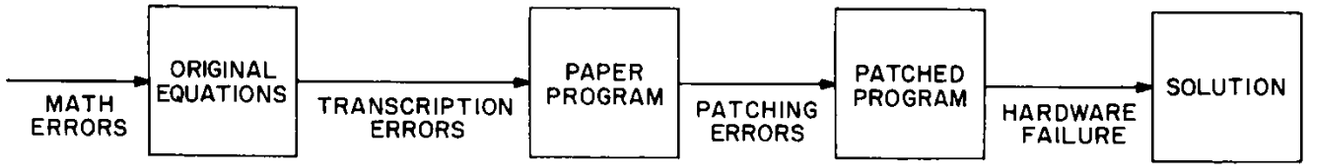


Figure 8.1. The Programming Process and Possible Errors

8.3 PROGRAM SHEET

Figure 8.2 shows the Program Sheet used when static checking and running of a problem. Use of the Program Sheet is best shown with the sample program given in Figure 8.3. Figure 8.2 shows how the devices are set up to perform the desired mathematical operations, what the variable represented by each output is, and what the value of that output (under static check conditions) should be. This figure also shows the parameter represented by each coefficient, the pot setting, and the pot *output* under static check conditions. The latter value will help you track down many of the problems that may occur during patching.

8.4 STATIC CHECK PROCEDURE

The DVM and SELECTOR Control (Figure 8.4) are used to static check a problem patched on the computer. When the SELECTOR-1 indicates any number, 11-37, then the output of the mathematical element with that address is displayed on the DVM. Further, that output will also appear at the SELECTOR-1 patch panel terminal. For example, if SELECTOR-1 points to 11 (which in Figure 8.3 is an integrator) then the output of $\int 11$ is displayed on the DVM and also appears at the SELECTOR-1 terminal. SELECTOR-2 connects the selected output to the SELECTOR-2 patch panel terminal.

EAI MiniAC™

PROGRAM _____ NAME _____

_____ DATE _____

MATHEMATICAL FUNCTION

	FUNCTION	NOTES	OUTPUT
11	Σ (∫)	$-2\dot{c}(0)$	+.800
12	Σ (∫)	c	+1.000
13	Σ HG		
14	Σ TS		
15	M (D) Σ	$\frac{(2\dot{c})^2}{c}$	+.640
16	LOG EXP Σ		
21	Σ ∫		
22	Σ ∫		
23	Σ HG		
24	Σ TS		
25	(M) D Σ	$-(2\dot{c})^2$	+.640
26	LOG EXP Σ		
31	Σ ∫		
32	Σ ∫		
33	Σ HG		
34	Σ TS		
35	M D Σ		
36	II 20 INV		
37	II INV INV		

COEFFICIENT

	NOTES	SETTING	OUTPUT
11	$2\dot{c}(0)$	.800	-.800
12	$c(0)$	1.000	-1.000
13			
14			
15			
16			
21	Constant	.500	+.400
22	$\frac{\rho}{20}$	.150	+.096
23			
24			
25			
26			
31			
32			
33			
34			
35			
36			

VFG GAIN: 1 3 10 30

GAIN: 1 3 10 30

	X	Y	X	Y
0				
1				
2				
3				
4				
5				
6				
7				
8				
9				
10				

GATE: 1 2 3 4 5 6 7 8 9 10

FF: 1 2 3 4

CTR/TMR:

PB: 1 2

MAN SW: 16 26 36

UP

OFF

DN

COMP: 13 23 33

RATE: F

TIMER: X

NOTES _____

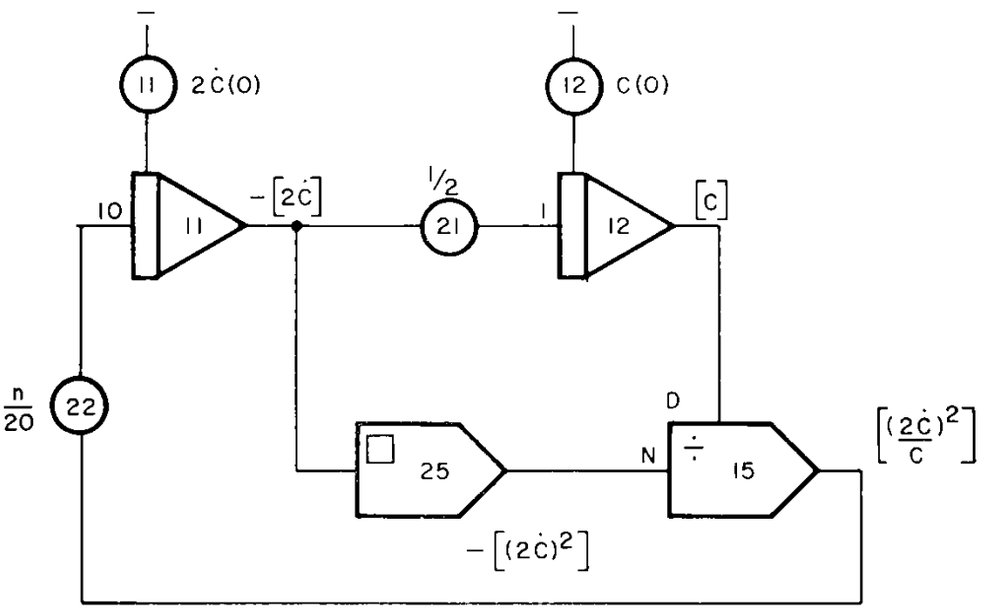
PUBLICATION NO. 955.0108-0

M1665

Figure 8.2. Sample Program Sheet

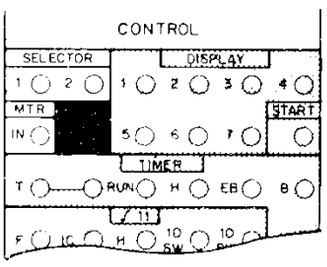
$$\frac{d}{dt}\left[2\dot{C}\right] = 10\left(\frac{n}{20}\right)\left[\frac{(2\dot{C})^2}{C}\right]$$

a. EQUATION

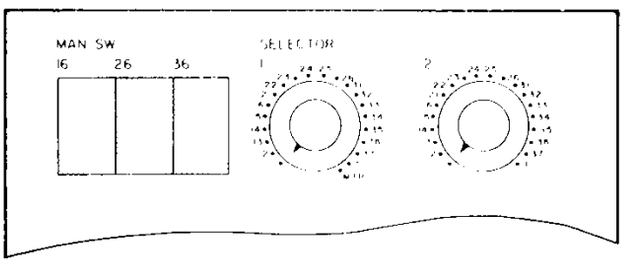


b. PROGRAM DIAGRAM

Figure 8.3. Sample Problem



PATCHING AREA



CONTROLS

Figure 8.4. Selector: Patching Area and Controls

The following is a step-by-step static check procedure for the sample problem illustrated in Figure 8.3 and documented on the Program Sheet (Figure 8.2). It is assumed that you have performed all analog initializing procedures (Chapter 3).

1. Select ANALOG MODE-IC.
2. Readout all mathematical elements elements in order using SELECTOR-1.
   a. SELECTOR-1: 11; does DVM = +0.800?
   b. SELECTOR-1: 12; does DVM = +1.000?
   c. SELECTOR-1: 15; does DVM = +0.640?
   d. SELECTOR-1: 25; does DVM = +0.640?
3. If one of the integrators reads incorrectly, stop right there and check the IC pot setting (depress corresponding pot pushbutton and read DVM).

   If the integrators are OK, but one (or more) other device is not, check the patching from other elements to this one, and the outputs of any pots (Step 6) connected to its inputs.
4. Readout each pot in turn (depress the associated pot pushbutton and read the DVM). Record the output of each pot on the program sheet.

At this point, it is well to review just what has been accomplished and, perhaps more importantly, what has not been accomplished. The measurements just described cover many checkpoints, but are not a complete check. Since the output of an integrator in the IC mode does not depend on its input(s), the X10 input of $\int$ 11 and X1 input of $\int$ 12 have not been checked. Notice that the P12-$\int$ 12 patching is checked because the output of $\int$ 12 depends on the output of P12, but *not* on P21. Remember static checking is performed on the IC mode! How can you be sure that these "uncheckable" connections are correct?

To reduce the possibility of patching errors, follow exactly the patching shown on your diagram, and check the patching against the diagram. Lightly tracing over the interconnecting lines on the diagram while patching the problem is a good method of ensuring that the patching is complete and correct. The use of a yellow pencil is recommended, as it is easily visible on other diagrams, but does not mar the drawing.

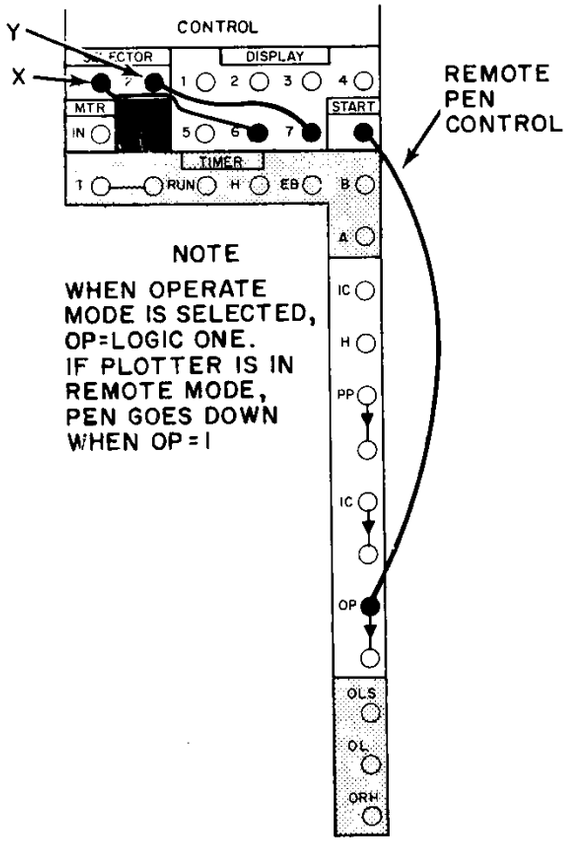


Figure 9.1. Typical Patching for X Versus Y Plot

CHAPTER 9

OBTAINING RESULTS USING AN X-Y PLOTTER

9.1 INTRODUCTION

The X-Y Plotter is the traditional device used for recording "hard copy" of analog computer solutions. The typical general purpose plotter has the following characteristics:

1. Movable pen and arm (Y and X) which follow input signals.
2. Controls for locating the initial pen and arm positions (Y0 and X0).
3. Adjustable scale factor controls for pen and arm, calibrated in volts/inch or volts/centimeter.
4. Plotting paper hold-down system, usually a vacuum system.
5. Remote or local control of writing (pen control).

The following paragraphs describe: connecting the plotter to the MiniAC; calibrating the plotter; selecting the signals to be plotted; and operating the computer to produce a result.

If you have an EAI Plotter compatible with the MiniAC, refer to Paragraph 9.2. Otherwise, read Paragraph 9.3.

9.2 MiniAC TO EAI PLOTTER

9.2.1 CONNECTOR LOCATIONS

At the rear of the MiniAC, there is a connector labeled DISPLAY. The plotter data cable must be plugged in here. With the cable in place, signals are routed from the MiniAC to the plotter via the DISPLAY terminations in the upper part of the patch panel CONTROL field. The DISPLAY inputs (Figure 9.1) are allocated as follows:

1 – Y1	SCOPE
2 – Y2	
3 – Y3	
4 – Y4	
5 – X	
6 – X	PLOTTER
7 – Y	
START – PEN UP/DN	

Three terminals are used: 6,7, and START. Terminal 6 is the X input to the plotter, terminal 7 is the Y input, and START allows a *logic* signal to operate the pen (position it *up* or *down*) remotely.

9.2.2 X VERSUS Y PLOTTING

One typical connection is shown in Figure 9.1. Here we are plotting the output two devices using SELECTOR 1 and 2. Each selector can be set to any active mathematical element. The pen is controlled by the Operate (OP) mode signal.

With the plotter calibrated (Paragraph 9.4) and paper in position, a solution can be recorded. Start with the computer in the IC mode and then press the OP button. The pen will drop to the writing surface and the run will start. When a sufficient record has been made, press the IC button to re-initialize the computer.

9.2.3 Y VERSUS T PLOTTING

A Y versus T (time) plot can be made in exactly the same manner as X versus Y described above. If device selected by SELECTOR 1 is an integrator as shown below the integrator will provide a "time-Base" T seconds long (it takes T seconds for the output of the integrator to go from 0 to + 1 machine unit.

The TIMER can also be used to provide a time base. This eliminates the need of using one of the six MiniAC integrators. A time base of a desired length can be set up using the TIMER – 1- .1- .01- .001 SECONDS and the X5-X105 controls. For example, a 20 second time base is set by selecting 1 SECOND and dialing 20 with the X5-X105 control. The patching for this is shown in Figure 9.2. Note that the MiniAC must be in PP for the timer to operate.

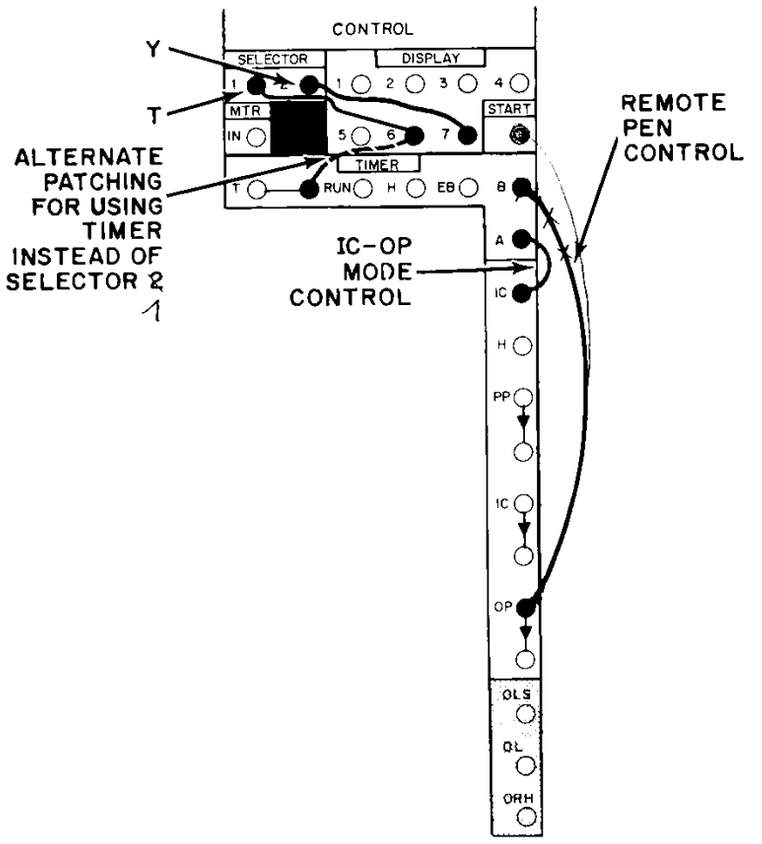


Figure 9.2. Typical Patching for Y Versus T Plot

The procedure is as follows:

1. Select IC Mode.
2. SELECTOR-1: T
3. SELECTOR-2: Device whose output is to be plotted.

4. Set TIMER duration using SECONDS and X5-X105.
5. Press PP button: This transfers mode control to the TIMER which will place the computer in OP for the duration set in Step 4, and in IC for 0.7 second. *The same plot will be repeated,* unless you press the IC button (removing control from the TIMER), prior to another OP period.

9.3. MiniAC TO NON-EAI PLOTTER

If you have a non-EAI Plotter, and do not have a compatible data cable, the DISPLAY terminations cannot be used to supply signals from MiniAC to that plotter. Instead, the SELECTOR terminations must be patched directly to the X and Y terminals of the plotter. Furthermore, you may have to raise and lower the pen manually. The pen must be raised before the end of the OP period or a retrace will occur.

The plotter manufacturer may have provided a means of using an external logic signal to lift the pen. Consult his manuals; if a level of +5 volts is required, use the true output of OP (as used to drop the pen on the EAI Plotter).

9.4 CALIBRATION

The simplest way to calibrate an X-Y plotter involves the following procedure:

1. Set all plotter scale factors to their lease sensitive positions (*highest* volts/inch).
2. Input zero for both X and Y. This can be done by grounding the X and Y input terminals.
3. Position the pen and arm to Y0 and X0. The Y0/X0 position will usually be either in the center of the paper for a four-quadrant plot, at the lower left for a first-quadrant plot, or at the left center for a first-and fourth-quadrant plot.

At this point assume that we wish to plot $\theta/200$ (θ is measured in degrees) against time. Also assume that the time scale factor is such that each second of computer running time (T) represents 2 hours of problem time and the real time of the problem is 60 hours. The computer run time is then 30 seconds (60/2). We are using 11 x 17 inch graph paper (a 10 x 15 inch plotting area) and expect that θ may be both positive and negative. Therefore, the calibrated graph should appear as shown in Figure 9.3.

From out knowledge of amplitude scaling, we know that $|\theta/200| \leqslant 1$. Therefore, Step 4 will allow us to properly label the vertical axis.

4. With Y0/X0 set at the point shown, unground the Y input and patch it to +1.0 machine unit The pen should move slightly. Increase the scale factor sensitivity until the pen points exactly to the adjustment point shown in Figure 9.3. Remove the +1.0 from Y.
5. Unground the X input and repeat Step 4 using the X-scale factors. This allows us to properly label the X-axis.
6. Restore the X and Y connections to the SELECTOR system.

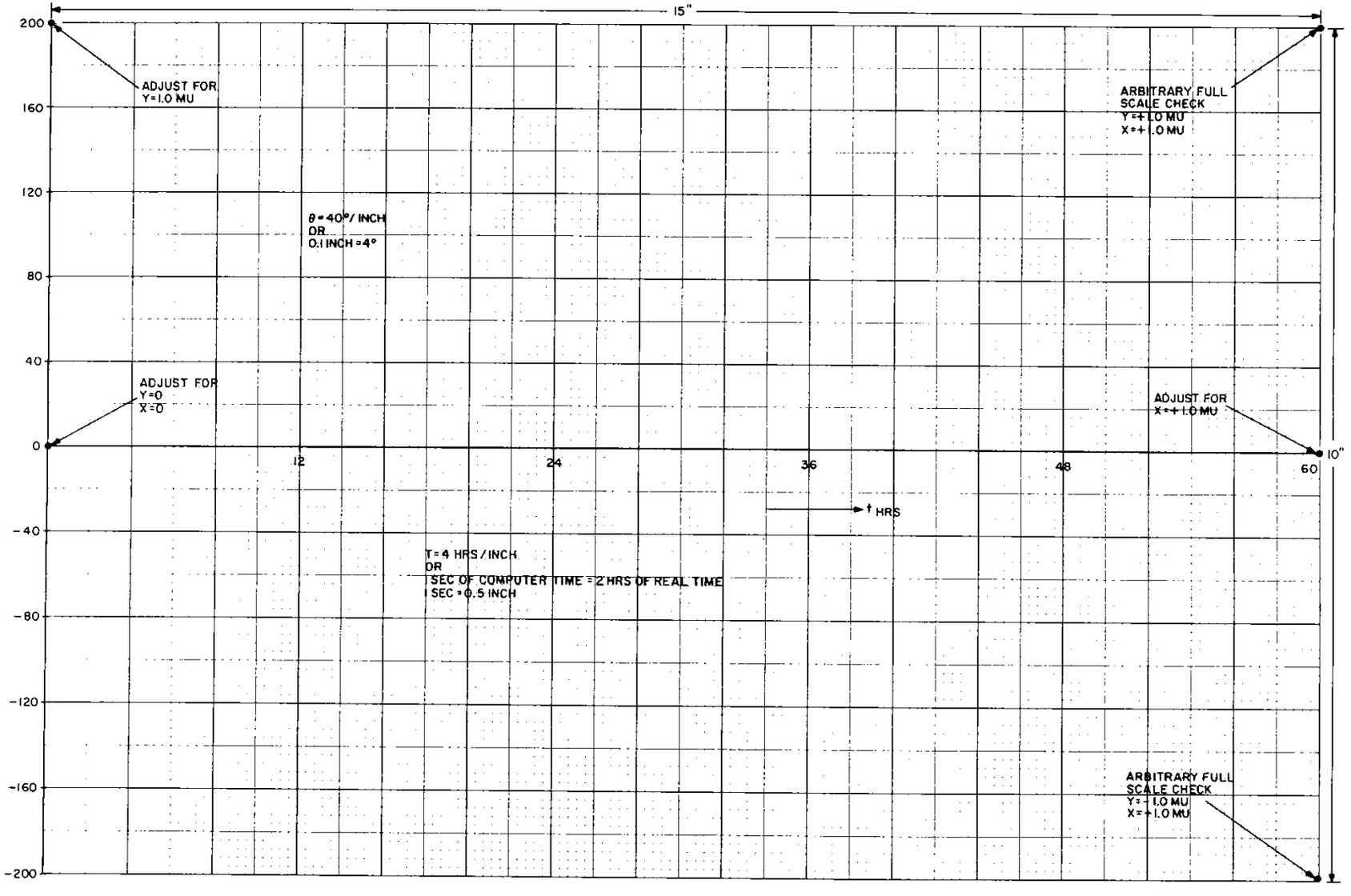


Figure 9.3. Sample Plotter Calibration

OBTAINING RESULTS USING AN OSCILLOSCOPE AND HIGH-SPEED REPETITIVE OPERATION

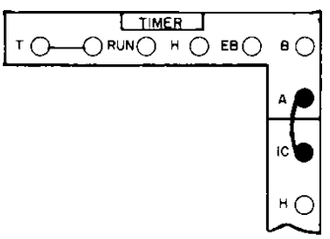
10.1 INTRODUCTION

While the X-Y plotter (Chapter 9) is the basic tool for recording hard copy output of an analog computer, it is an electro-mechanical device and is relatively slow in operation. Additionally, it is not always necessary to record every problem solution.

Many simulations involve multiple runs with one or more parameter values changed between runs. For example, plotting the results of each of 100 runs with 10 values each for two parameters would take 1 to 2 hours. In such case, repetitive operation (Rep-Op) at a high rate of repetition and an oscilloscope (for observing the solution curves) should be used. Then, when the desired parameter values are obtained (by adjusting pots, etc.) we can slow down the solution and plot the final results in slow time.

10.2 REP-OP

Repetitive operation (Rep-Op) permits automatic switching between the computer OP and IC modes. High speed Rep-Op can produce up to 150 solutions per second. When using oscilloscope to display the Rep-Op problem solution you can observe how system behavior changes as a pot (parameter) is adjusted. Patch the TIMER as shown below, and then proceed as follows:



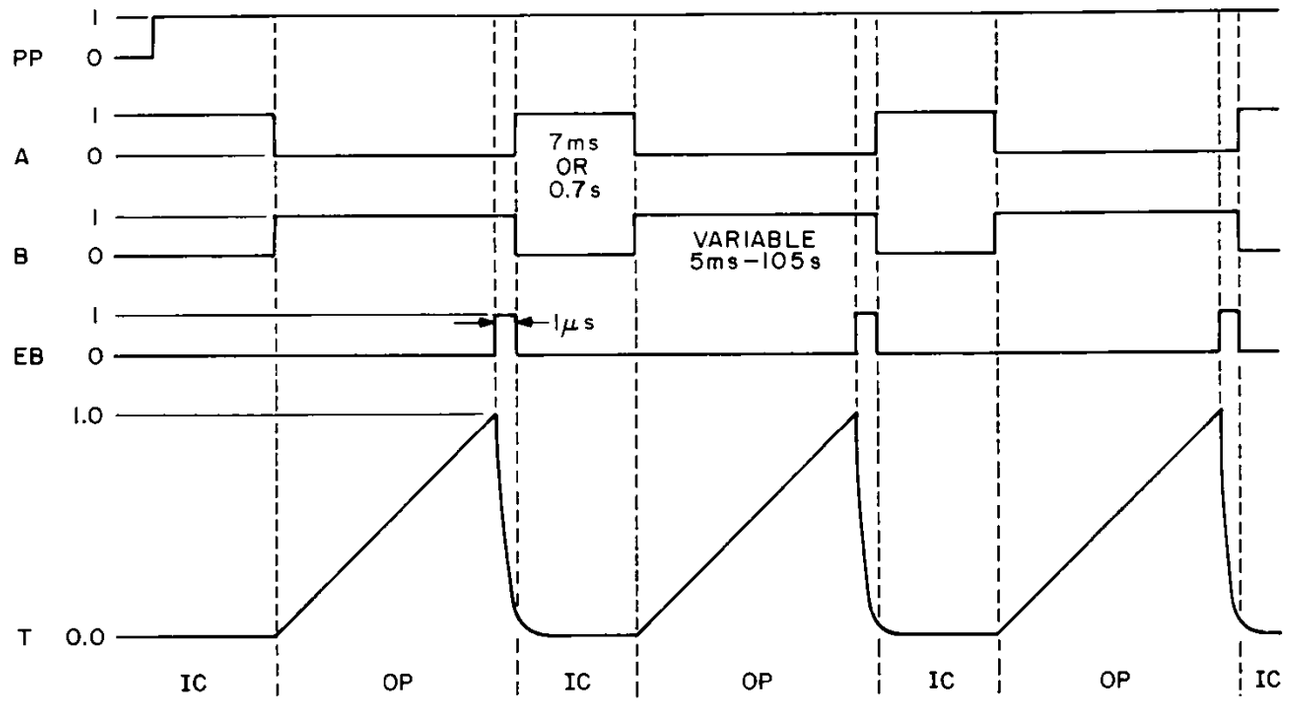
1. Replace all manual switches that are changed during the solution, with programmed switches (Paragraph 5.9).
2. Set the TIMER for T/500. T is the length of time the program runs when in slow time.
3. Press the RATE pushbutton. FAST will light.
4. Connect the signals for display on the scope (Paragraph 10.4).
5. Press the ANALOG MODE-PP pushbutton. The program will start.
6. Adjust the problem parameters as required to obtain the desired results.

10.3 THE TIMER

The TIMER provides logic signals A, B, EB, and an analog signal T. The A and B signals are timed intervals that are used for controlling computer modes. The A interval is about 0.7 second long when the TIMER-SECONDS switch is at 1 or .1, and 7.0 milliseconds when it is at .01 or .001. The B interval can be set for any duration from 5 milliseconds to 105 seconds using the TIMER X5-X105 control. The B interval is typically set for 5 milliseconds to 105 milliseconds in high speed Rep-Op. EB is a one-microsecond pulse that occurs just prior to the end of the B interval. T is an analog signal that increases linearily from 0.0 to 1.0 machine units in one B interval.

There are two logic control input terminations (RUN and H) for the timer. The Timer operates when RUN = 1 and PP is depressed. When RUN = 0 the timer goes to and remains in the A interval. When RUN goes high, the Timer goes through a normal A Time before going to B. RUN need only go low for one microsecond to force the Timer into the A

interval. The H termination can be used to hold the H interval. When H = 1, the Timer remains in the B interval and T = the analog value attained when H goes high. When H goes low, the Timer will complete the normal B interval. More detailed information is given in Chapter 29.



10.4 MiniAC TO EAI SCOPE

The EAI Model 0.34.0035-1 Scope can be used with the MiniAC as a Rep-Op display device. An interconnecting cable is provided with the display scope which plugs into the DISPLAY connector at the rear of the MiniAC and the scope input connector. The scope accepts four signal inputs (Y_1, Y_2, Y_3, and Y_4,) that may be plotted against time (SWEEP mode) or against Y_4 (X Plot mode). The 1-4 DISPLAY terminations (in the patch panel control field of the MiniAC) correspond to the $Y_1 - Y_4$ scope inputs. The sweep signal for the scope is normally obtained by patching TIMER-T to DISPLAY-5 and also provides automatic blanking during IC.

10.5 MiniAC TO NON EAI-SCOPE

If you do not have an EAI Model Scope for Rep-Op display, EAI can provide an optional general purpose data cable for interconnection with the non-EAI scope. This cable has eight output plugs that correspond to the DISPLAY 1-7 and START terminations on the patch panel.

NOTE

The START output at the DISPLAY connector is a buffered logic signal (1=5V and 0=0V) and is in phase with the patch panel START input.

If the non EAI scope has both X and Y inputs, it may be used like an X-Y/T-Y plotter. Consult the reference or operator's manual supplied by the scope manufacturer for instructions concerning X-Y and T-Y displays. Remember that OP time is typically between 5 milliseconds and 105 milliseconds in high-speed Rep-Op.

LOGIC AND ITERATIVE OPERATION

11.1 INTRODUCTION

The logic signal is defined as a time varying signal having one of two values: one (1) and zero (0). Logic signals are available at the output of any device terminated in the patch panel logic field, and (with the exception of the analog outputs SELECTOR 1 and 2, and TIMER-T) all red terminations in the control field. Refer to Figure 2.2.

With the exception of MTR-IN and DISPLAY-1 to 7, all green terminations in the logic and control fields respond to logic signal inputs or commands. This chapter describes the various methods of controlling analog modes using logic signals, the various logic components, and iterative operations.

11.2 LOGIC CONTROL OF ANALOG MODES

11.2.1 OVERLOAD DETECTION

One of the most basic of all automatic mode control logic programs is the automatic overload detection program. At the lower right corner of the CONTROL field, there are three terminations OLS, OL and ORH. OL provides a logic output that is high (equals logic ONE) whenever a component overloads (OverLoad).

If a momentary overload occurs, you may not see the corresponding overload indicator light up. You can store this overload condition by patching OL to OLS (OverLoad Store). Thus if a momentary, overload occurs, the light will remain on. An overload condition must exist for about two milliseconds before OL goes high.

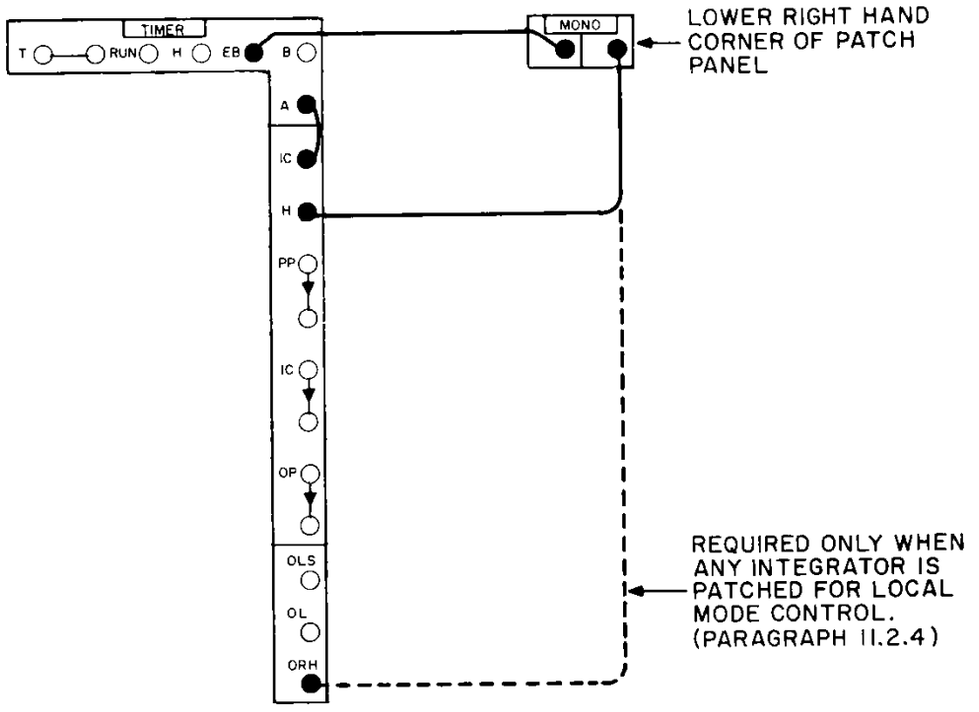
The occurrence of an overload means that a device has been forced to operate outside its proper operating range, invalidating the solution produced from that instant on. If OL is patched to ORH, the entire computer will be forced into the HOLD mode (OverRide Hold). Whenever ORH=1, the computer is in Hold *regardless* of any other patching or mode selection by pushbutton except SP.

11.2.2 THREE MODE REPETITIVE OPERATION

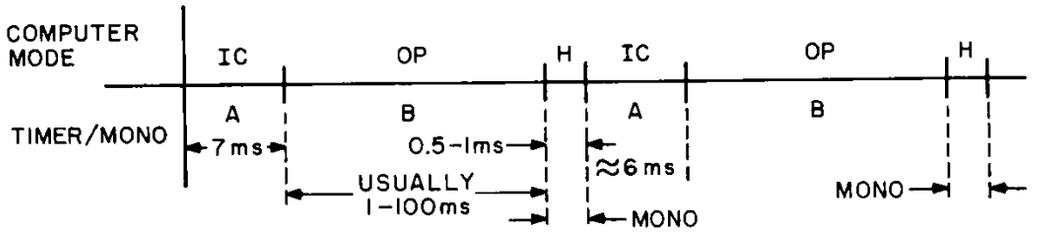
Features of the MiniAC permit two-mode (IC and OP) and three-mode (IC, OP, and HD) repetitive operations. Two mode Rep-Op is described in Chapter 10. Three-mode Rep-Op is often used during iterative calculations. For example, the program is initialized (IC), run (OP), and then stopped (HD) so that given measurement or observations can be made. The measurements are then translated into new or updated parameter values that are introduced during the next IC interval. Then, the program is run again (OP). Iterative operations on the MiniAC can be performed at the Normal (or slow) rate, or at the more usual FAST rate. The time interval for each mode at each rate is shown in the following table;

Rate	IC (Fixed)	OP	HD (Fixed)	Note
SLOW	0.7 sec.	5ms-105S	0.5-1 ms	RATE and OP are separate selections.
FAST	7.0 ms	5ms-1.05S	0.5-1 ms	

Typical patching for three-mode operation is as follows:

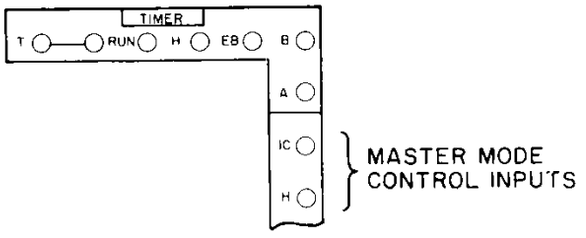


When the PP button is depressed, a signal is sent internally to start the TIMER which responds by providing a signal equivalent to a full A (IC) interval. At the end of the B interval, a signal EB (End of B) is sent to the mono which sends a pulse of 0.5 to one millisecond duration to the analog mode H and ORH control inputs. At the end of this H-period, the mode is IC for the remainder of the A interval. Then, another B or OP interval occurs, and the cycle begins anew.



11.2.3 MASTER MODE CONTROL BY LOGIC SIGNALS

So far we have discussed how the MiniAC can be controlled from the mode control pushbuttons and by the Timer, in either two or three-mode Repetitive Operation. In general, the computer can be controlled from any set of logic signals patched to the IC and H input terminals in the CONTROL field.



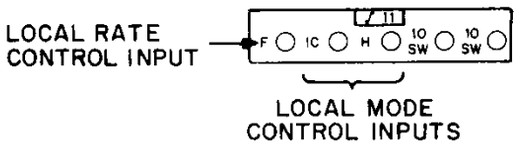
The computer takes on modes as indicated by the following table:

Logic Signal Input Patched To:		Resulting Mode	Note
IC	H		
0	0	OP	When unpatched, IC input terminal is high (1) and H input terminal is low (0).
0	1	H	
1	0	IC	
1	1	H	

Thus, two-mode operation, (switching between IC and OP) requires only one logic signal, patched to IC. If this signal is called M, then M=0 is OP and M=1 is IC. Three-mode operation, of course calls for two logic signals. To initiate such logic control, press the PP button. This also enables the TIMER.

11.2.4 LOCAL MODE CONTROL BY LOGIC SIGNALS

In addition to pushbutton control and master mode control, both of which control all integrators as a group, you can control individual integrators from the local control terminals in the CONTROL field. The local control area for ∫11 is shown below:



We have already discussed the two terminals marked 10 SW (Chapter 5). The IC and H terminals are exactly analogous to the IC and H terminals of the master mode control, and follow a similar mode control truth table. When unpatched, or both inputs are high, the integrator follows the master mode selection.

The F terminal is used to locally control the integrating rate of the particular integrator. It is analogous to the RATE (FAST) button on the control panel. There is no master rate control logic input.

F Input Signal	Rate
Not Patched	Selected by RATE button
0	Normal, X1
1	Fast, X500

The symbol and patching for a locally fast integrator is:

and for a locally normal integrator:

NOTE

A source of logic 0 is terminated to the left of each comparator logic area in the CONTROL FIELD. Logic 1 can be obtained from the true output of any unused AND-gate, or from the two terminals above GATE 5.

11.2.5 MODE CONTROL PRIORITIES

1. *SP* – this is the highest priority mode and is selected by pushbutton only.
2. *ORH* – this overrides all other mode control, be it pushbutton or logic signal except SP. Only when ORH=0 or is not patched can the MiniAC follow other modes.
3. *Local Mode* – any mode control signals patched to the IC/H terminals of an integrator block will take precedence over logic signals patched to the master mode inputs and over button-selected mode control.
4. *Master Mode, Pushbuttons* – pushbutton mode control commands all non-locally controlled components.
5. *Master Mode, Logic Signals* – logic signals to the master mode IC/H are only effective if the PP button is pressed.

11.3 LOGIC DEVICES

11.3.1 AND, OR, NOT GATES

Gates are devices for combining logic signals in an algebraic fashion. This particular algebra is called Boolean algebra and has three basic operations: AND, OR, NOT.

Inputs		Outputs		
a	b	a AND b (a·b)	a OR b (a+b)	NOT a ($\bar{a}$)
0	0	0	0	1
0	1	0	1	1
1	0	0	1	0
1	1	1	1	0

Expression	Symbol
a AND b	a•b, ab
a OR b	a+b
NOT a	$\overline{a}$

Some of the basic laws of Boolean algebra are summarized below (X, Y are Boolean variables):

$X \cdot Y = Y \cdot X$

$X+Y = Y+X$

$(\overline{X}) = \overline{X}$

$(X \cdot Y) \cdot Z = X \cdot (Y \cdot Z)$

$(X+Y)+Z = X+(Y+Z)$

$X \cdot (Y+Z) = X \cdot Y + X \cdot Z$

$X+(Y \cdot Z) = (X+Y) \cdot (X+Z)$

$(\overline{X+Y}) = \overline{X} \cdot \overline{Y}$

$(\overline{X} \cdot \overline{Y}) = \overline{X}+\overline{Y}$

$X+X = X$

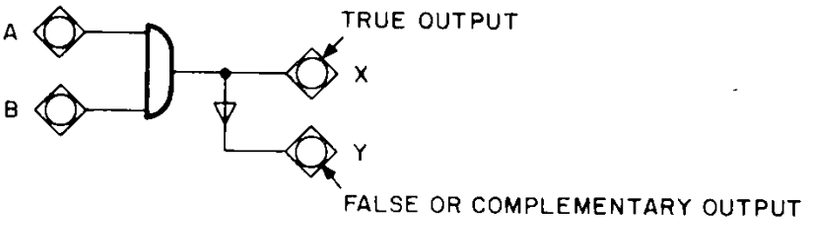
$X \cdot X = X$

$X+\overline{X} = 1, X \cdot \overline{X} = 0$

$X+0 = X, X \cdot 0 = 0$

$X \cdot 1 = X, X+1 = 1$

The AND-gate supplied with the MiniAC is capable of performing all three Boolean operations.

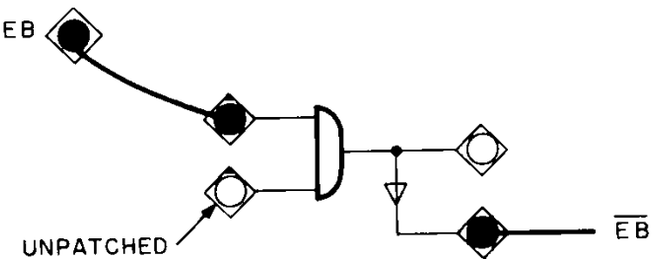


A	B	X	Y
0	0	0	1
0	1	0	1
1	0	0	1
1*	1*	1	0

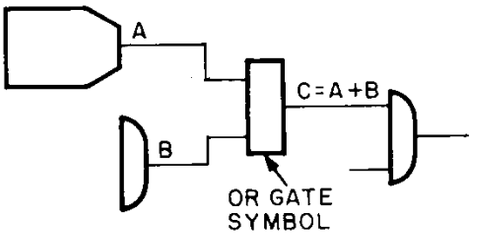
or not patched

Note that $Y = \overline{X}$, and that an unpatched AND-gate behaves as if its inputs are patched to ones. This is true not only for the six two-input gates (1-6) but also for the four four-input gates (7-10); all unpatched inputs behave as if a logic 1 were patched.

Every device producing a logic signal has both TRUE and FALSE outputs terminated at the patch panel except for the MONO, PULSES, OL and the TIMER EB. If you need the complement of one of these signals, say, $\overline{EB}$, use an AND-gate as indicated.



Should you need the OR function, as shown below



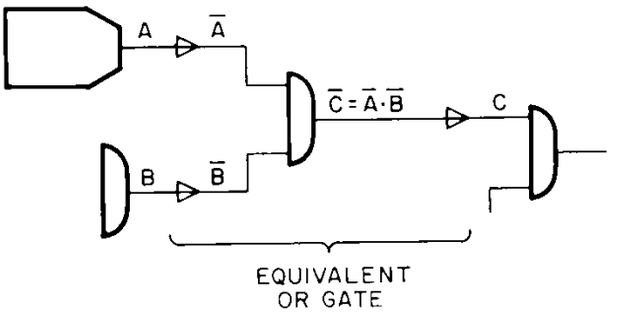
use an AND gate with the following Boolean algebra

$$C = A+B$$

$$\overline{C} = (\overline{A+B})$$

$$\overline{C} = \overline{A}\cdot\overline{B}$$

$$\overline{\overline{C}} = C = (\overline{\overline{A}\cdot\overline{B}})$$

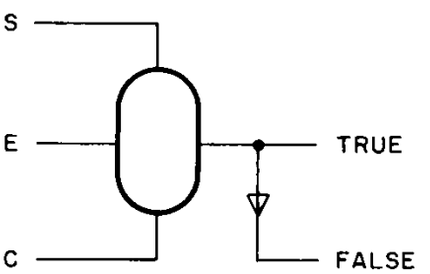


11.3.2 FLIP-FLOPS

11.3.2.1 Flip-Flop Operation

Flip-flops are logic *memory* devices, somewhat analogous to light switches found on the walls of your home. When you press the switch *up*, the lights go on ("SET") and the switch stays up even when you take your finger away. When you press the switch *down*, the lights go out ("CLEAR") and the switch stays down even when you take your finger away. In a similar manner, when you input a logic 1 to the S (SET) input of a flip-flop, it will become SET (the TRUE output becomes 1) and will stay SET even when you remove the 1 (apply a logic 0), and conversely.

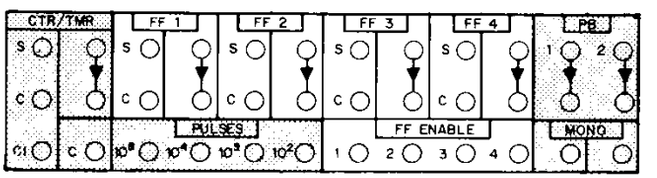
The flip-flop has five terminals, as shown below:



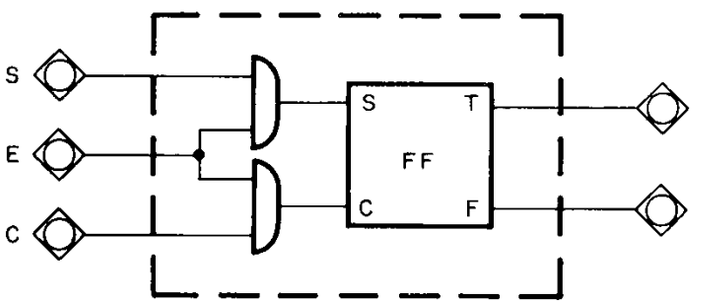
Inputs – S(SET) and C(CLEAR):

Outputs – TRUE and FALSE

Control – E(ENABLE)



The ENABLE can be better understood with a somewhat revised flip-flop diagram:



If E = 1 or unpatched, then the two AND-gates which are an integral part of the flip-flop device can respond to S and C signals. If E = 0, then a logic 1 is inhibited from getting to the internal S and C connections, and the flip-flop *cannot change state.*

A – internal SET = S•E

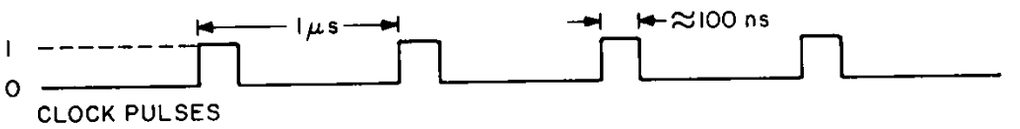
B = internal CLEAR = C•E

Recall that X•1 = X and X•0 = 0, and the purpose of ENABLE should be clear.

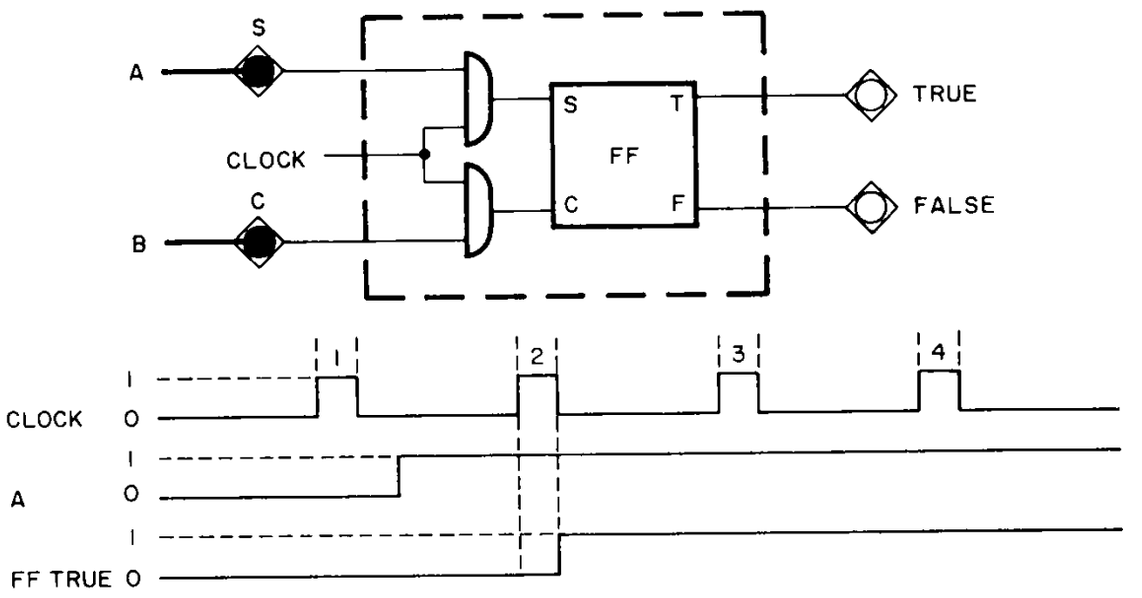
11.3.2.2 Flip-Flop

11.3.2.2 Clocked Flip-Flops

The MiniAC flip-flops are synchronized to an internal logic system clock. The basic reason for synchronization is to resolve the uncertainties caused by delays in signal propagation that occurs in unsynchronized systems. The clock signal is a series of narrow pulses occurring at the rate of one per microsecond.



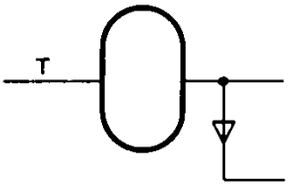
The key to synchronization is the concept that actions can occur *only* during the time that the clock pulse is switching from a high to a low. It is as if we had a synchronized flip-flop gated to the clock as shown.



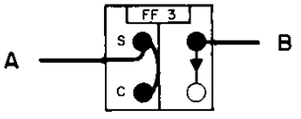
Suppose the flip-flop is cleared with the clock running (logic mode RUN), and a signal A is applied which goes high sometime between clocks 1 and 2. Between 1 and 2, the flip-flop is inactive and does not respond. However, during clock 2, the clock enables the flip-flop to see A = 1 as a SET command. The actual change takes place *abruptly* on the *trailing edge* of clock pulse 2.

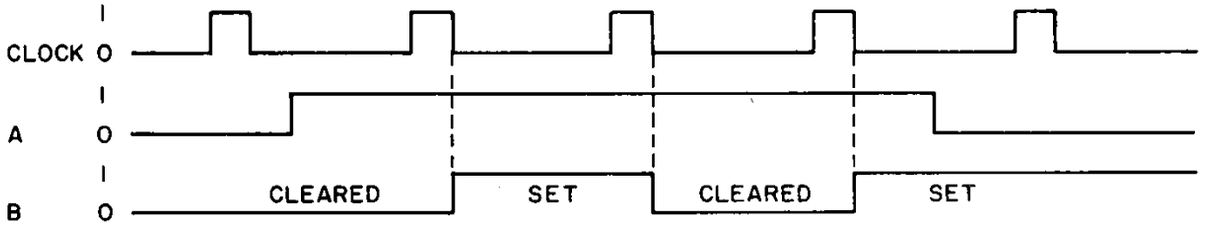
11.3.2.3 A Triggered Flip-Flop (Toggle Flop)

A triggered flip-flop (or toggle flop) is one that changes state with each clock pulse. Symbolically it is shown as:



The patching for the triggered flip-flop in the MiniAC is as follows:





11.3.3 SHIFT REGISTER

A shift register can be patched from flip-flops. A shift register is a collection of two or more flip-flops connected in such a manner as to store a pattern of 1's and 0's; and transfer these across the register. A 3-bit register, the program symbol and the basic timing requirements are illustrated in Figure 11.1. Operation of this register is described in Table 11.1.

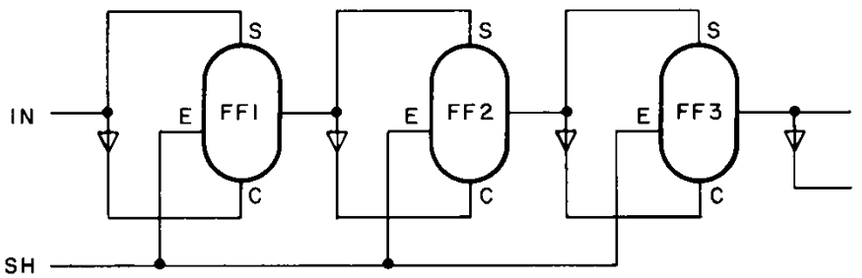
Table 11.1. Shift Register Operation

Clock	IN	SH	Action	Bit Pattern
1	1	0	None	000
2	1	1	Shift 1-bit from IN to Flip-Flop	100
3	0	0	None	100
4	0	1	Shift 1-bit from Flip-Flop 1 to Flip-Flop 2 and 0-bit from IN to Flip-Flop 1.	010
5	0	0	None	010
6	0	1	Shift 1-bit from Flip-Flop 2 to Flip-Flop 3, 0-bit from Flip-Flop 1 to Flip-Flop 2, and 0-bit from IN to Flip-Flop 1.	001
7	0	0	None	001

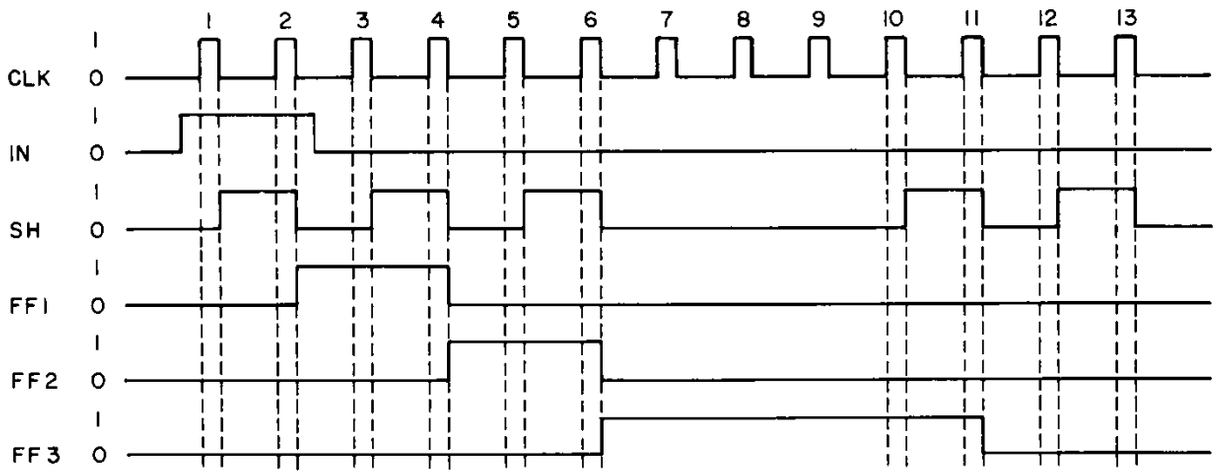
(Cont)

Table 11.1. Shift Register Operation (Cont)

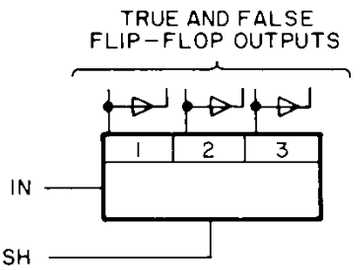
Clock	IN	SH	Action	Bit Pattern
8	0	0	None	001
9	0	0	None	001
10	0	0	None	001
11	0	1	Shift 1-bit from Flip-Flop 3 out of register, 0-bits into all Flip-Flops.	000
12	0	0	None	000
13	0	1	Shift 0-bits into all Flip-Flops.	000



a. 3-BIT REGISTER



b. BASIC TIMING



c. SYMBOL

Figure 11.1. Shift Register Operation

11.3.4 BINARY COUNTING

Many logic programs involve counting of events that can be represented by logic signals. We approach counting these events by preselecting a certain number of these occurrences and then counting them with a logic upcounter (Figure 11.2). The preselected number, N, can be anything in the range 1 - 127, and allows as many as 128 occurrences of an event to be counted. The preselected count is set into the counter by depressing one or more of the numbered pushbuttons (64, 32, 16, 8, 4, 2, 1) such that their total is *one less* than the desired count.

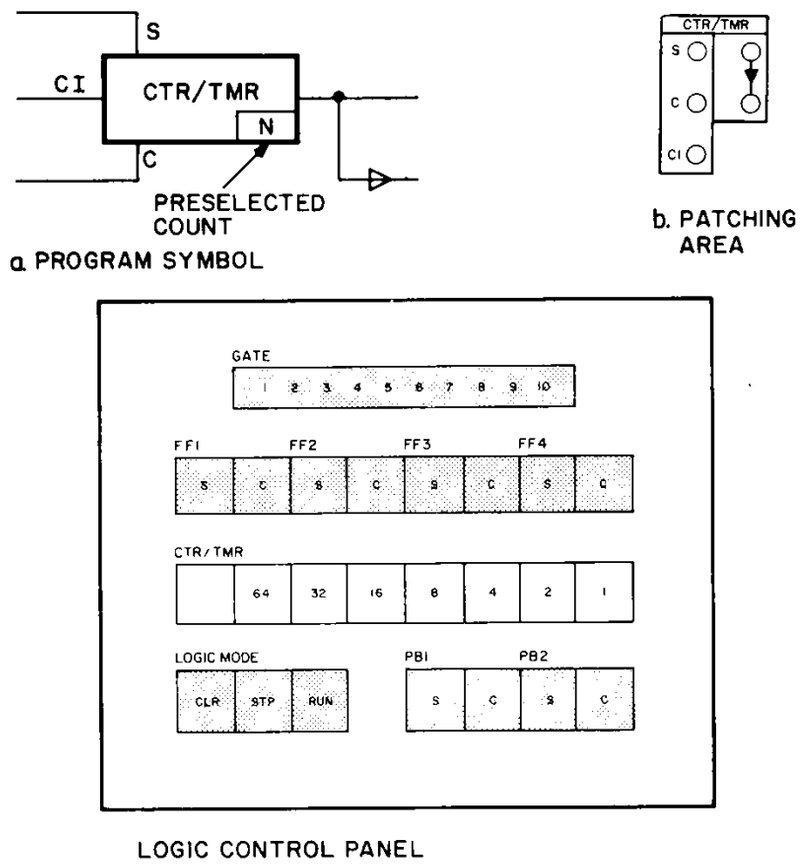
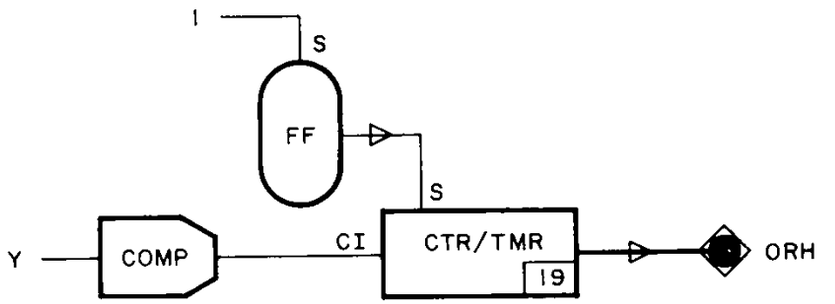


Figure 11.2. Logic Counter

The rules for counter usage are simple:

1. The counter must be set before attempting to count. This is accomplished by inputting a short duration high to the S terminal.
2. The CI (CARRY In) signal is a logic signal. The counter increments each time CI goes from 0 to 1, but NOT from 1 to 0.
3. A high into the C (Clear) terminal makes the counter output low on the next clock, but does not clear the register associated with the 1, 2, 4, . . . 64 pushbuttons. The logic mode CLR button clears the register.
4. If C goes high while CI is low the counter will increment one and the indicated output may be invalid.

A simplified circuit diagram is shown in Chapter 29 of the Reference Handbook. Let us examine a simple counting application that is typical of the use of counters. Suppose we wish to place the MiniAC in Hold whenever a certain variable, y, becomes positive for the 20th time.



When you change the logic mode from CLR to RUN, a short duration set signal is output from the flip-flop, this makes the counter/timer output 1 and the ORH input 0.

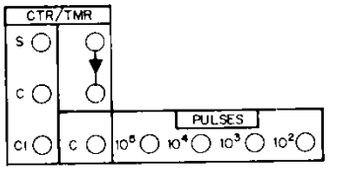
Whenever Y goes from a negative value to a positive one, the comparator output goes from 0 to 1 and the counter increments. After 20 increments, the ORH input becomes high and the MiniAC enters the HOLD mode.

If you need to merely count the number of times an event occurs, convert that event to a logic signal which goes high to signify occurrence and patch to the CI of the counter with N = 127. When the run is completed, do *not* change the logic mode to CLR, as this would clear the pushbutton register. Add up the numbers on the still-lighted pushbuttons and you will have the number of times the event occurred.

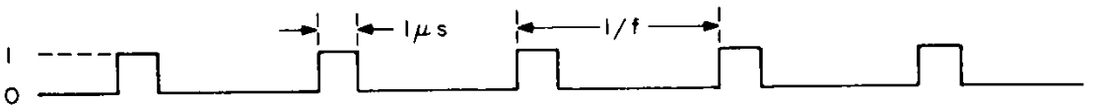
11.3.5 LOGIC TIMING

We have used the analog mode or rep-op timer previously, and have seen that the smallest time that could be set is 5 milliseconds. Our logic, on the other hand, operates with a clock having a period of one microsecond – 5000 times faster. Thus, a separate time is required for timing logic events.

Logic timing can be accomplished by counting events which occur regularly. Such "events" on the MiniAC are usually pulses terminated on the patch panel adjacent to the CTR/TIMER.

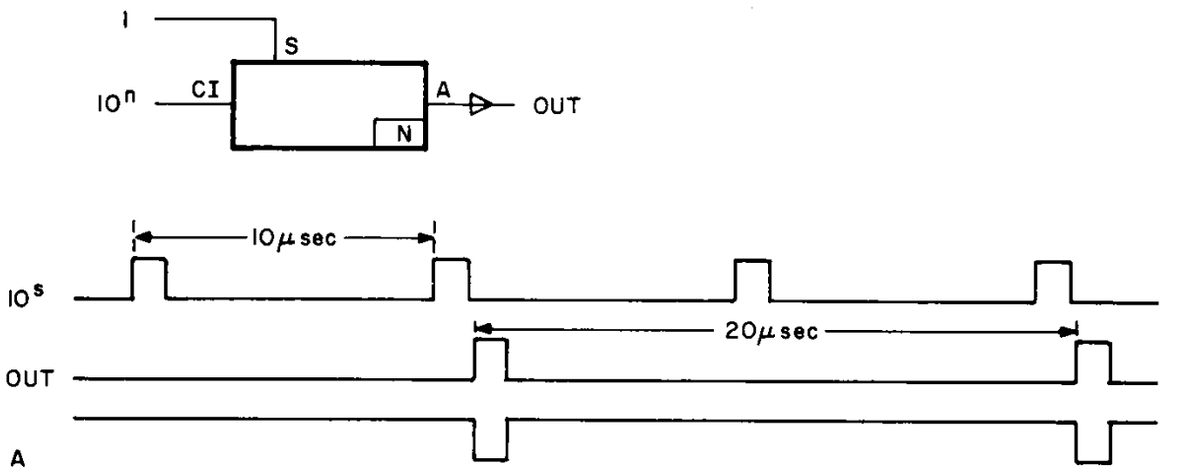


Each pulse train looks like this:



where f = 10^5, 10^4, 10^3, or 10^2, as selected.

In other words, you can choose a pulse train of 1 microsecond pulses occurring 100,000 times/sec, 10,000 times/sec, 1,000 times/sec, or 100 times/sec. The timer configuration is:



In the most general case, the time between pulses from the false output of the CTR/TMR is:

$$(N+1)\ 10^{-n} \text{ seconds}$$

Thus, if you need a timing pulse that becomes high for 1 microsecond every second, press the 64, 32, 3(=99) buttons and connect the 10^{-2} pulse output to the CI.

Considering the extreme and several intermediate cases, we have the following table.

Time Between Pulses	N	10^{-n}
100 microseconds	9	10^5
1 millisecond	99 9	10^5 10^4
10 milliseconds	99 9	10^4 10^3
100 milliseconds	99 9	10^3 10^2
1 second	99	10^2
1.28 second (Maximum)	127	10^2

11.3.6 SWITCHABLE LOGIC LEVELS

The MiniAC has two logic pushbuttons (PB) which control two sources of 1 and 0.

Each button has two controls on the logic control panel (Figure 2.6), one labeled S and the other C. When S is pressed, it lights and the true output of the PB goes high. When C is pressed, the S light is extinguished and the false output goes high. PB's are very useful in checkout and for introducing manual control in the logic program.

11.4 ANALOG MEMORY

The ability of the analog computer to simulate at high speed becomes the ability to automatically iterate when memory is added to the analog "side" of the computer. Analog memory is performed by a device that samples an analog signal continuously and, on logic command, holds a single value of that signal. The device is known as a track-store, or T-S amplifier.

11.4.1 THE T-S UNIT (Figure 11.3)

When the Σ/TS switch is depressed, devices 14, 24, 34 can be used as T-S units.

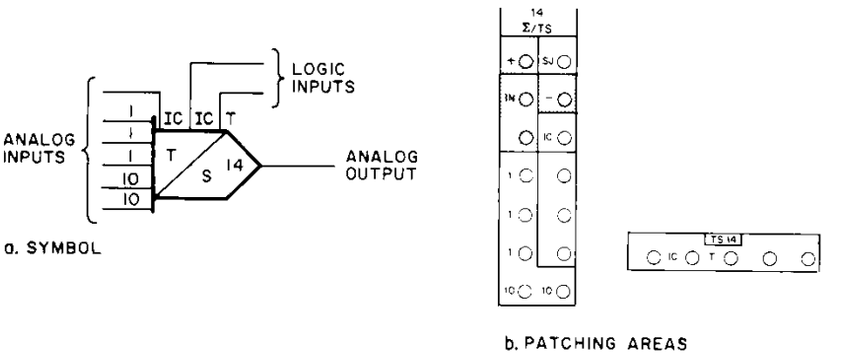


Figure 11.3. The Track-Store Unit

To understand how the T-S behaves, refer to Figure 11.4.

A is the logic IC signal: whenever A = 1, the output is the negative value of the analog IC input (in our example, −0.5 is the analog IC).

B is the logic T-S signal – B=1 is TRACK, B=0 is STORE. Note that A overrides B.

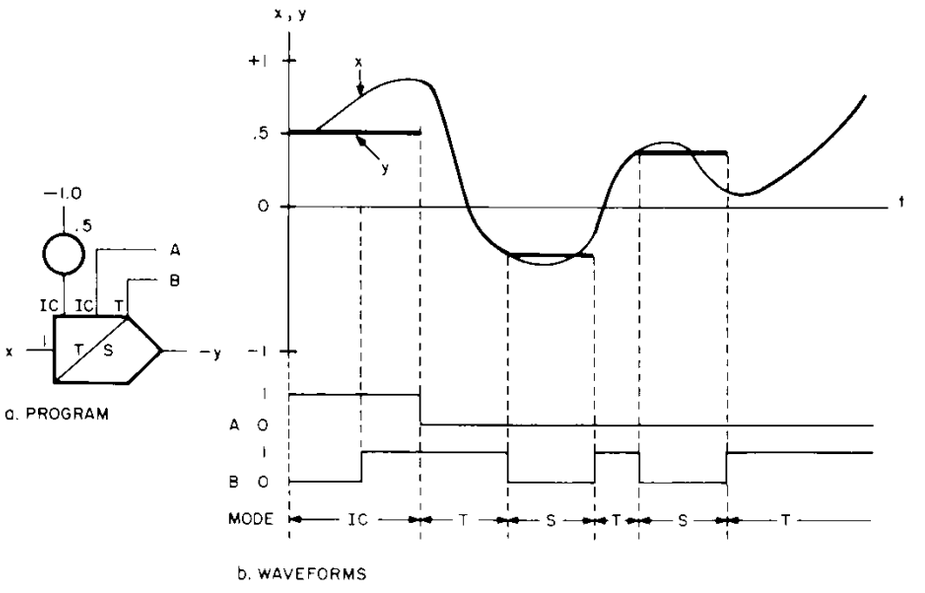


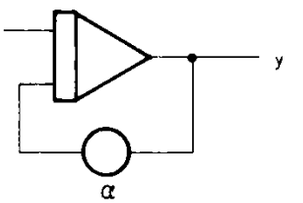
Figure 11.4. Track-Store Behavior

11.4.2 PARAMETER SWEEP PROGRAM

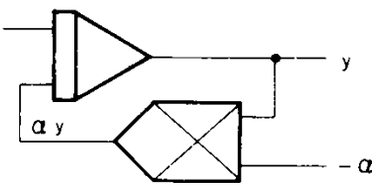
One of the most common uses of analog simulation is the investigation of a system behavior when a single parameter is changed. If operating the simulation manually, you will probably follow a regimen as:

1. Adjust the parameter to its initial value, the computer is in the SP mode.
2. Select IC, then OP and plot the variable of interest, say Y.
3. Go to H at the end of the run and note the value of Y, calling it $(y_{end})0$.
4. Select SP and set the parameter to the next value. Repeat 2 and 4.

The major ingredients of this recipe are adjust the parameter, control the modes, record the final value of y. We already know how to control modes with logic (Paragraph 11.2). Parameters can be introduced with multipliers in place of pots. That is

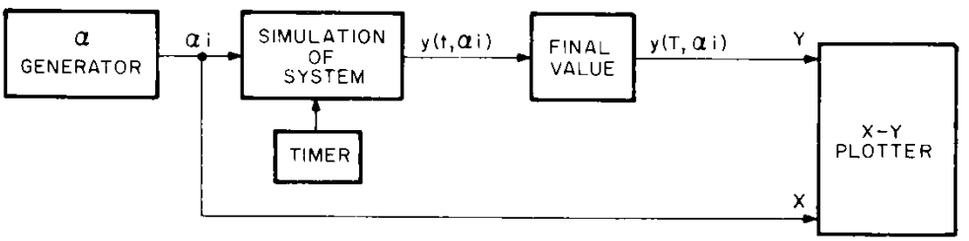


can be replaced by



if α is now a signal (a parameter which changes during IC but stays constant otherwise) instead of manually adjusted constant. We'll discuss how to generate α as a signal below. The question now is the record of the final value of y.

The basic set-up is:



Refer to Figures 11.5 and 11.6 for programming and patching information. The timer is set up to give 3-mode rep-op, the three modes being IC, OP, H occurring in that order. Adjust the B (OP) interval of the timer to give the desired OP time; use the oscilloscope display as a guide. The A(IC) interval and the H interval are fixed. The FAST integrator rate button must be depressed for high-speed operation.

Now, we use the T-S unit to pick up the final value of y. As the T-S output goes to the plotter, we would like to TRACK *only* when the simulation has gone to the H mode, and STORE otherwise. This ensures that the (T,αi) function will be a staircase-type function as shown.

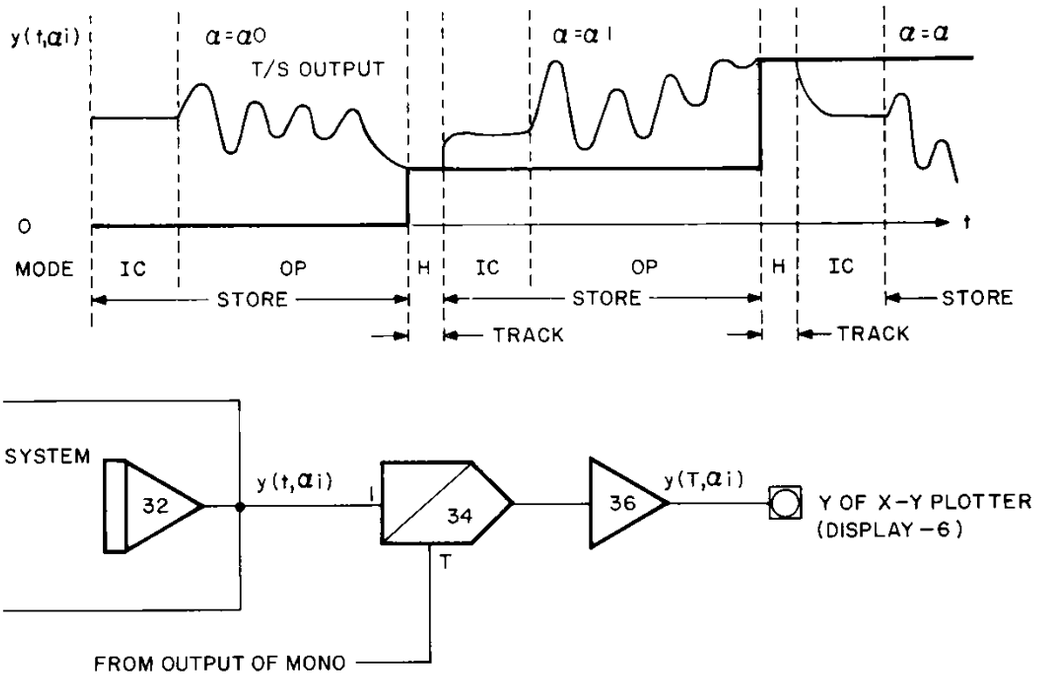


Figure 11.5. Parameter Sweep: Basic Program and Waveforms

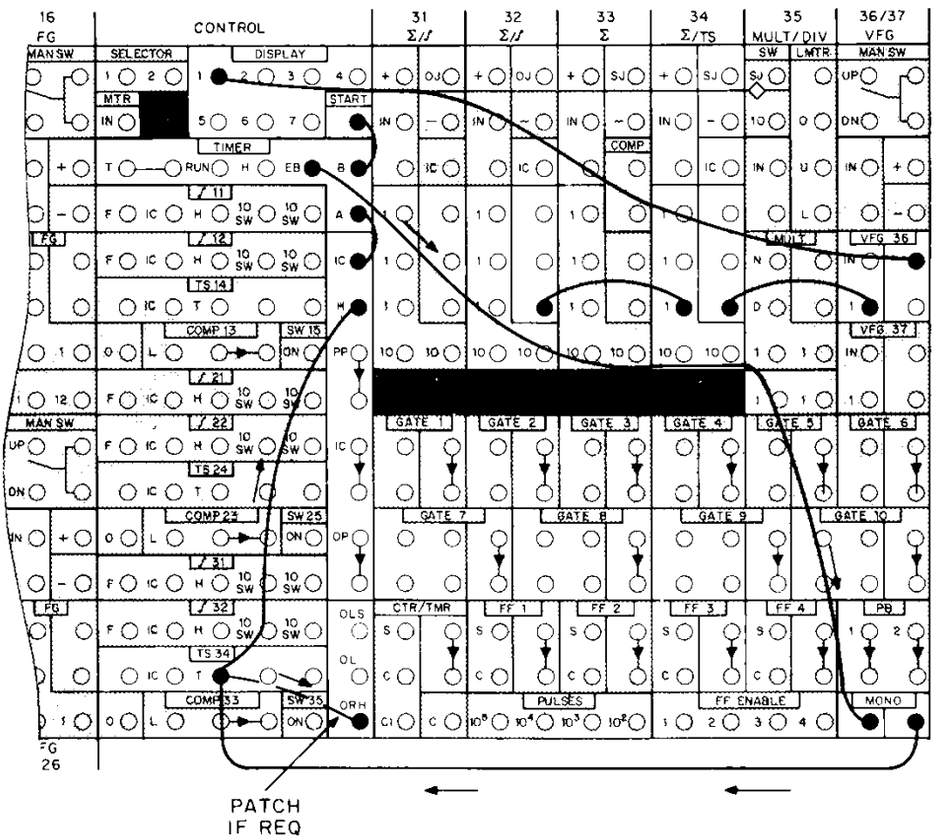
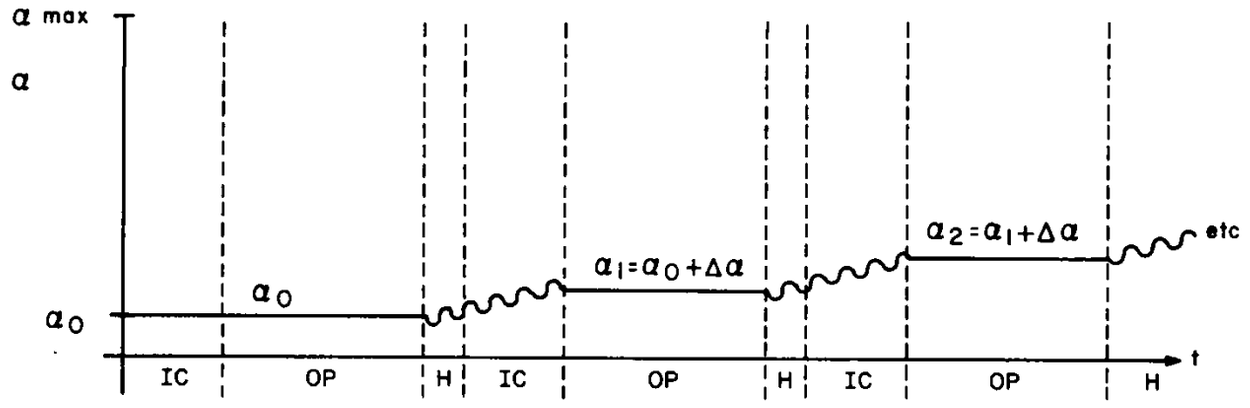


Figure 11.6. Parameter Sweep: Patching

11.4.3 PARAMETER IMPLEMENTATION

As mentioned above, we face the requirement for generating parameter values as if they were signals and not merely constants. There are several ways, but they all hinge on producing the following plot in rep-op:



The irregular line is meant to call your attention to one important fact: we don't care how we get from one α value to the next, as long as α is constant when required (in our case, during OP!)

11.4.3.1 Using an Integrator

The following scheme shown in Figure 11.7 causes the integrator to cycle between H and OP. Use the connection to ORH to establish the HOLD mode as this integrator will have local mode control.

There are several things to note:

1. This integrator runs in slow time.
2. The integrator initial condition is not the initial value α_0, it is the previous value, α_{-1}. We integrate over the first A interval to set $\alpha_0 = -\alpha_{-1} + \Delta\alpha$.
3. Time T is approximately 6 milliseconds and corresponds to the A interval minus the MONO time.

11.4.3.2 Using a Counter

If you can spare the flip-flops and switches, a counter can be used to handle 7 values of a given parameter, without the need for a multiplier.

Each IN pulse triggers Flip-Flop 1 (makes it become 1 if it was 0 and vice versa). Flip-Flop 1 is set and, when an IN pulse is applied, Flip-Flop 2 triggers. If Flip-Flop 1 and Flip-Flop 2 are set, the next IN pulse triggers Flip-Flop 3.

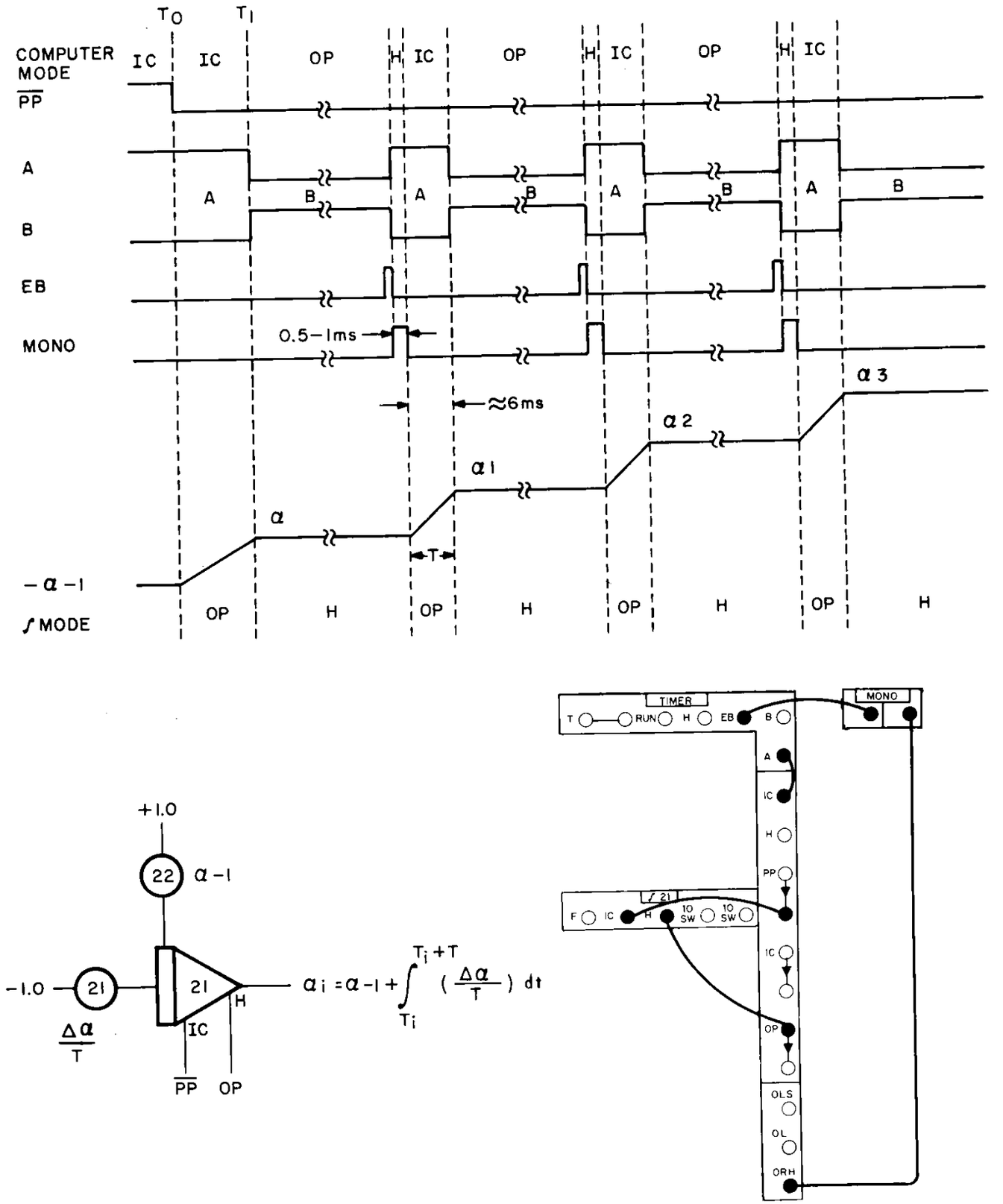


Figure 11.7. Parameter Implementation with an Integrator

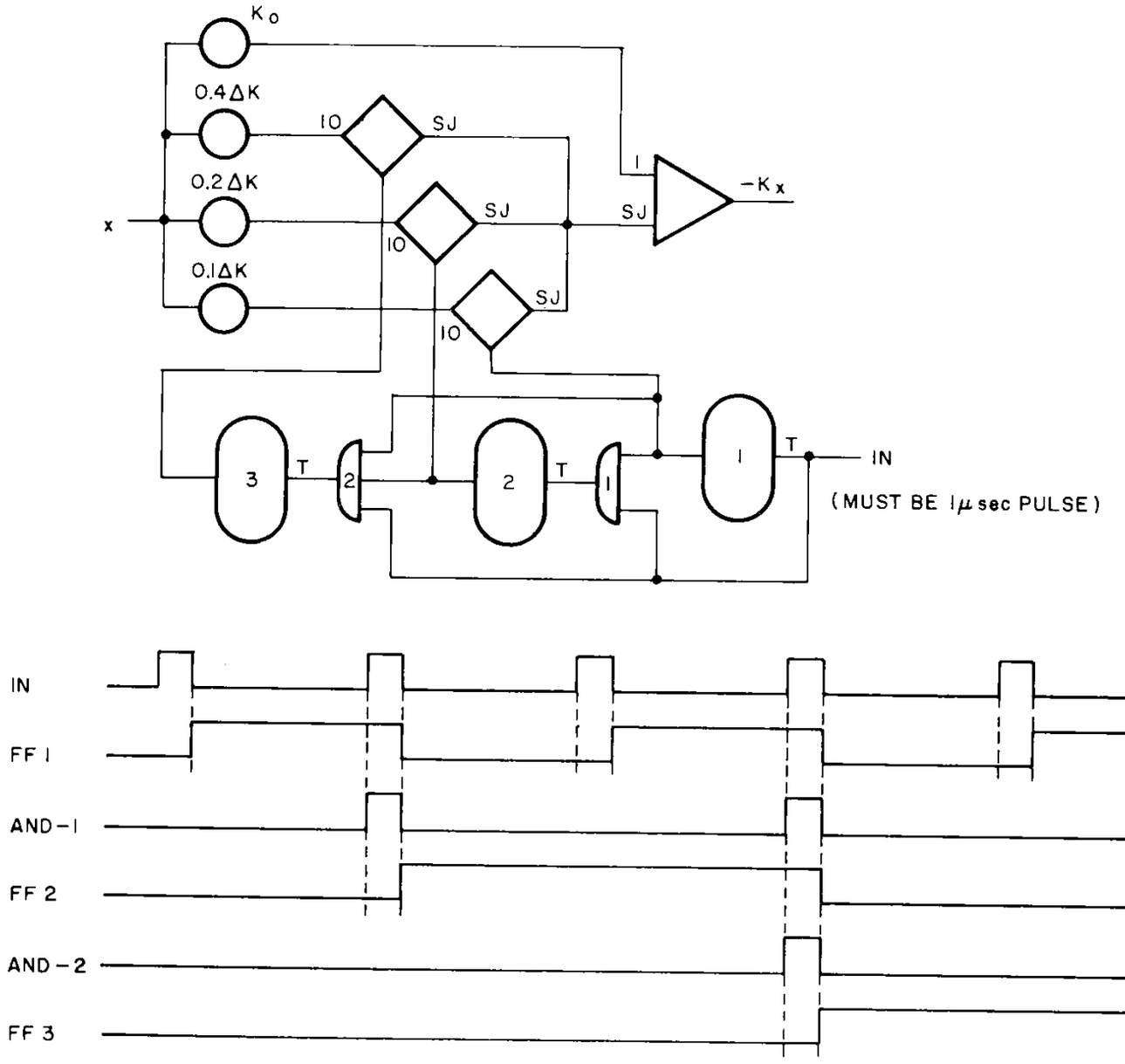


Figure 11.8. Parameter Implementation with a Counter

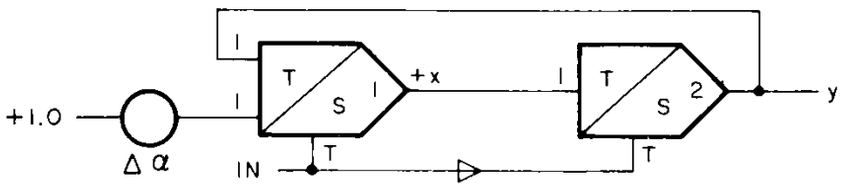
IN Pulses	Flip-Flop 3	Flip-Flop 2	Flip-Flop 1
0	0	0	0
1	0	0	1
2	0	1	0
3	0	1	1
4	1	0	0
5	1	0	1
6	1	1	0
7	1	1	1
8	0	0	0
Etc.			

In other words the combination shown is an 8-count (0-7) binary upcounter. The states of the switches produce the following output:

IN Pulses	K
0	K_o (all flip-flops are cleared)
1	$K_o + 10 (\cdot 1\Delta K) = K_o + \Delta K$
2	$K_o + 2\Delta K$
3	$K_o + 2\Delta K + \Delta K = K_o + 3\Delta K$
4	$K_o + 4\Delta K$
5	$K_o + 5\Delta K$
6	$K_o + 6\Delta K$
7	$K_o + 7\Delta K = K_{MAX}$
8	K_o
Therefore, $\Delta K = (K_{MAX} - K_o)/7$	

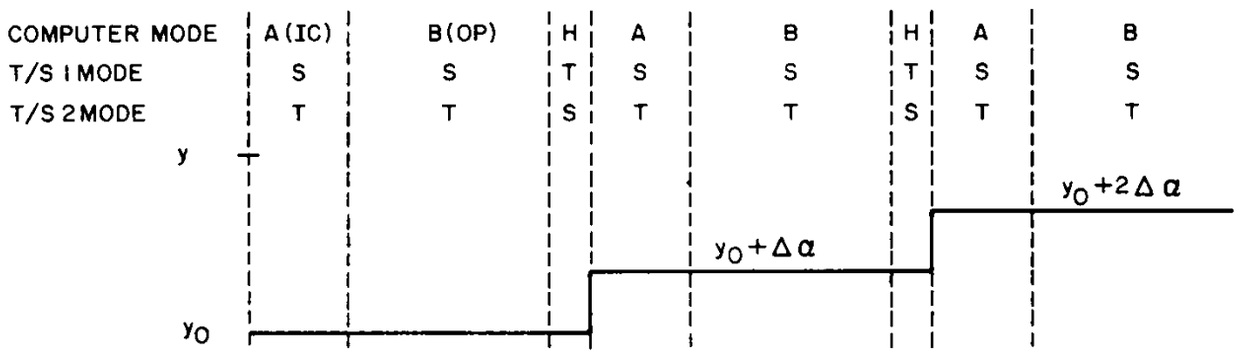
11.4.3.3 Using an Analog Accumulator

Another method for providing incremental parameter values uses a device called an analog accumulator.



IN	X	Y
1	$-\Delta a$	0
0	$-\Delta a$	Δa
1	$-2\Delta a$	Δa
0	$-2\Delta a$	$2\Delta a$
1	$-3\Delta a$	$2\Delta a$
0	$-3\Delta a$	$3\Delta a$
	etc.	

If IN is H (Hold), we have:



This circuit can also be patched using integrators as track-store units with certain limitations. Refer to Chapter 21.

11.4.4 BOUNDARY VALUE PROBLEM

The analog accumulator (Paragraph 11.4.3.3) has one restrictive characteristic: it uses up two T-S units. On the other hand, it is very useful in solving boundary problems, using proportional correction $a(i + 1) = a(i) + K\epsilon(i)$

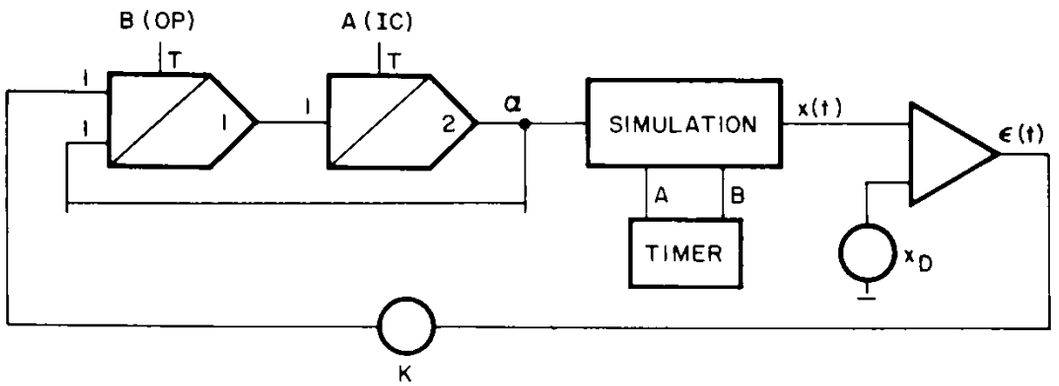
where

$a(i)$ is the value of a for the ith run.

$\epsilon(i)$ is the error in the final value of the variable at the boundary, i.e., ϵ = (x desired − x actual)/at the boundary.

and

K is the proportional constant.



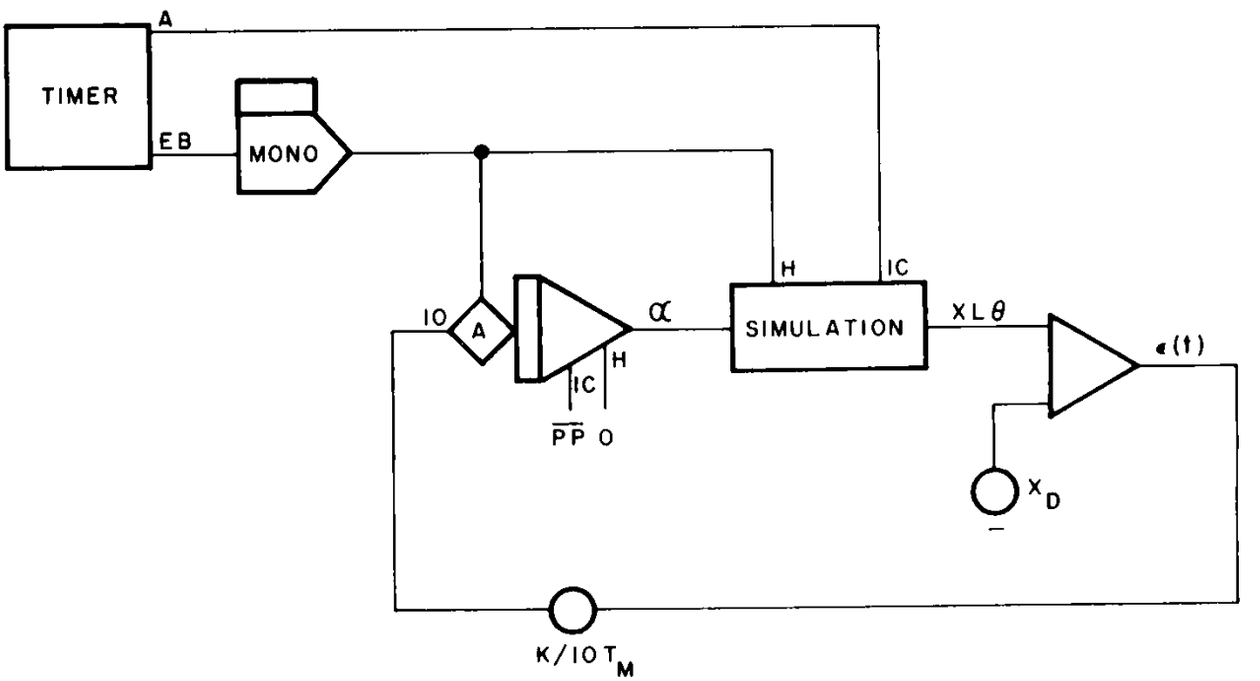
Note that the output of the summer, $\epsilon(t)$, is only meaningful at the end of B (and in HOLD). Since T-S 1 tracks during B, it has the correct $K\epsilon(i)$ at the end of B and supplies the proper $\alpha(i+1)$ to T-S 2 during A.

You may have to vary K to find the correct sign for convergence and a reasonable value for fast convergence. The sign of K is changed, of course, by adding an inverter after the summer which produces $\epsilon(t)$.

The single-integrator parameter increment circuit may also be used to solve boundary-value problems, as shown below. In this case, the entire simulation is put into the HOLD mode by the mono, so that when the mono output is high, the final value of the error is stored. During this interval, the input to the parameter updating integrator is turned on, so that the integrator output will change by an amount proportional to the input error. When the monostable period is over, the integrator input is turned off, and the simulation is put into the IC mode.

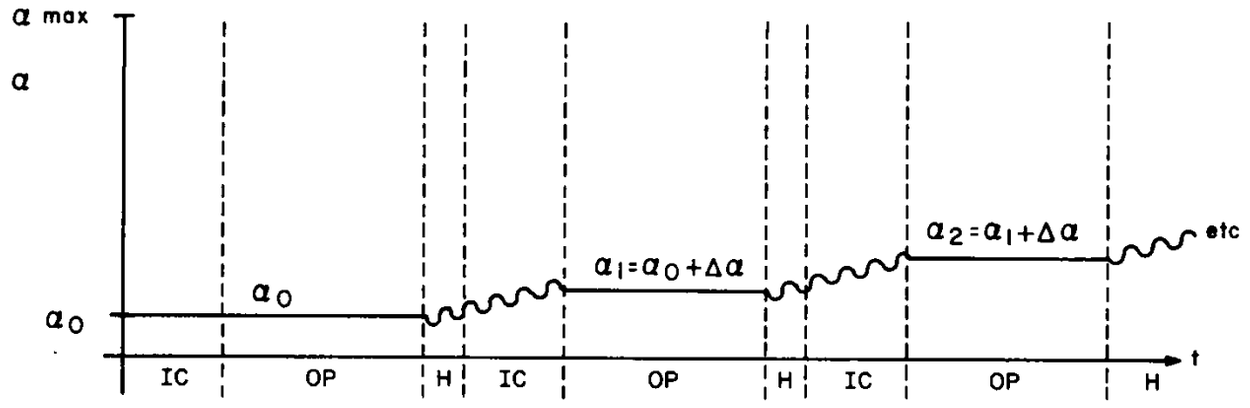
The parameter updating integrator is in the IC mode whenever the PP signal is low (i.e., whenever the iteration is being reset). Of course, any other logic signal may be used to control the resetting of the iteration scheme. The mono controls the mode of the simulation by means of the "HOLD" mode input terminal (*not* the "ORH" terminal, as this would prevent the parameter updating integrator from changing).

Note that the amount by which the parameter changes is proportional to the error, *and* to the mono interval T_M (nominally 0.5 to 1.0 milliseconds). For this reason, the pot-setting is made inversely proportional to T_M. In practice, the precise value of T_M is not significant, as the pot-setting is determined empirically in any case.



11.4.3 PARAMETER IMPLEMENTATION

As mentioned above, we face the requirement for generating parameter values as if they were signals and not merely constants. There are several ways, but they all hinge on producing the following plot in rep-op:



The irregular line is meant to call your attention to one important fact: we don't care how we get from one α value to the next, as long as α is constant when required (in our case, during OP!)

11.4.3.1 Using an Integrator

The following scheme shown in Figure 11.7 causes the integrator to cycle between H and OP. Use the connection to ORH to establish the HOLD mode as this integrator will have local mode control.

There are several things to note:

1. This integrator runs in slow time.
2. The integrator initial condition is not the initial value α_0, it is the previous value, α_{-1}. We integrate over the first A interval to set $\alpha_0 = -\alpha_{-1} + \Delta\alpha$.
3. Time T is approximately 6 milliseconds and corresponds to the A interval minus the MONO time.

11.4.3.2 Using a Counter

If you can spare the flip-flops and switches, a counter can be used to handle 7 values of a given parameter, without the need for a multiplier.

Each IN pulse triggers Flip-Flop 1 (makes it become 1 if it was 0 and vice versa). Flip-Flop 1 is set and, when an IN pulse is applied, Flip-Flop 2 triggers. If Flip-Flop 1 and Flip-Flop 2 are set, the next IN pulse triggers Flip-Flop 3.

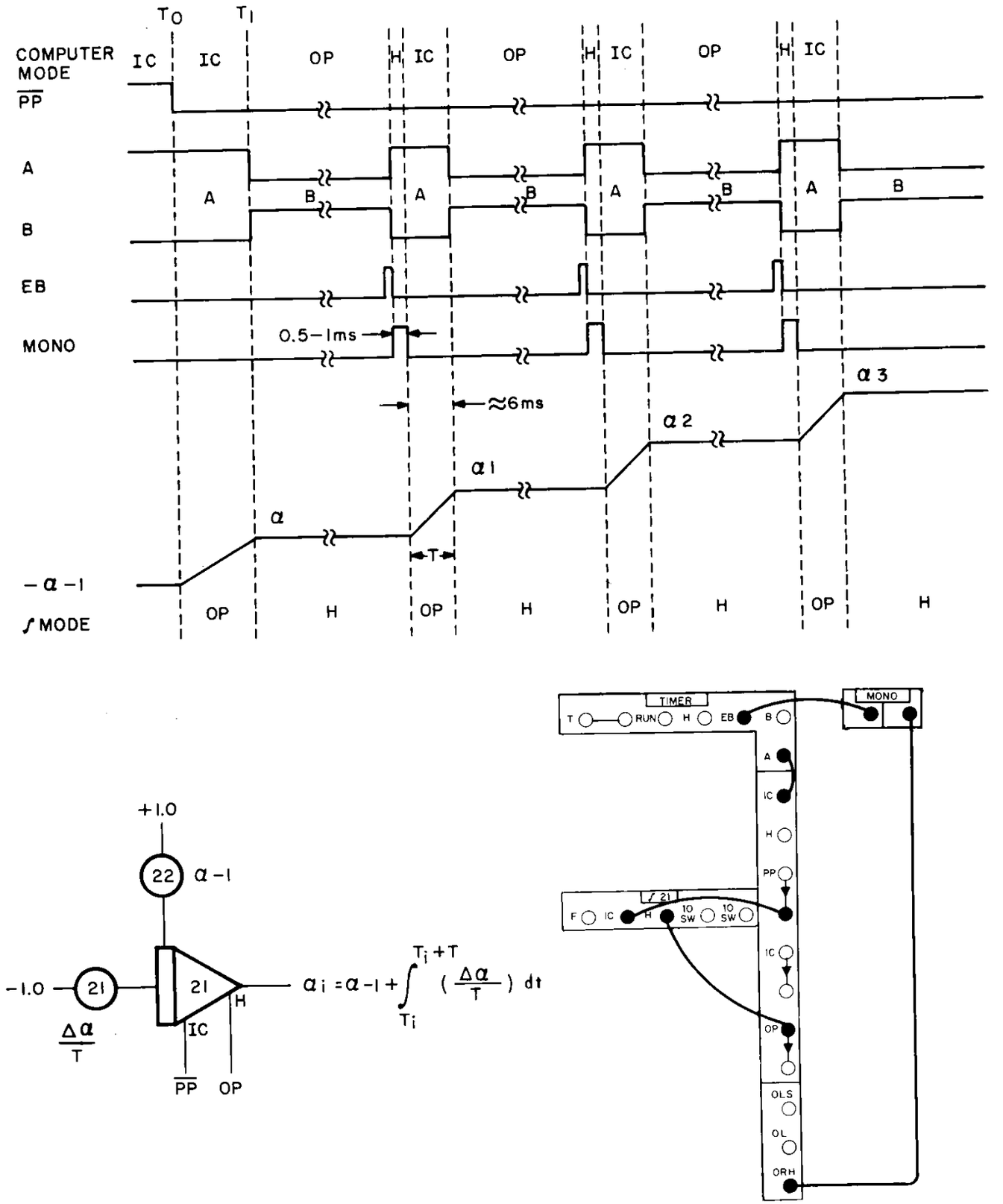


Figure 11.7. Parameter Implementation with an Integrator

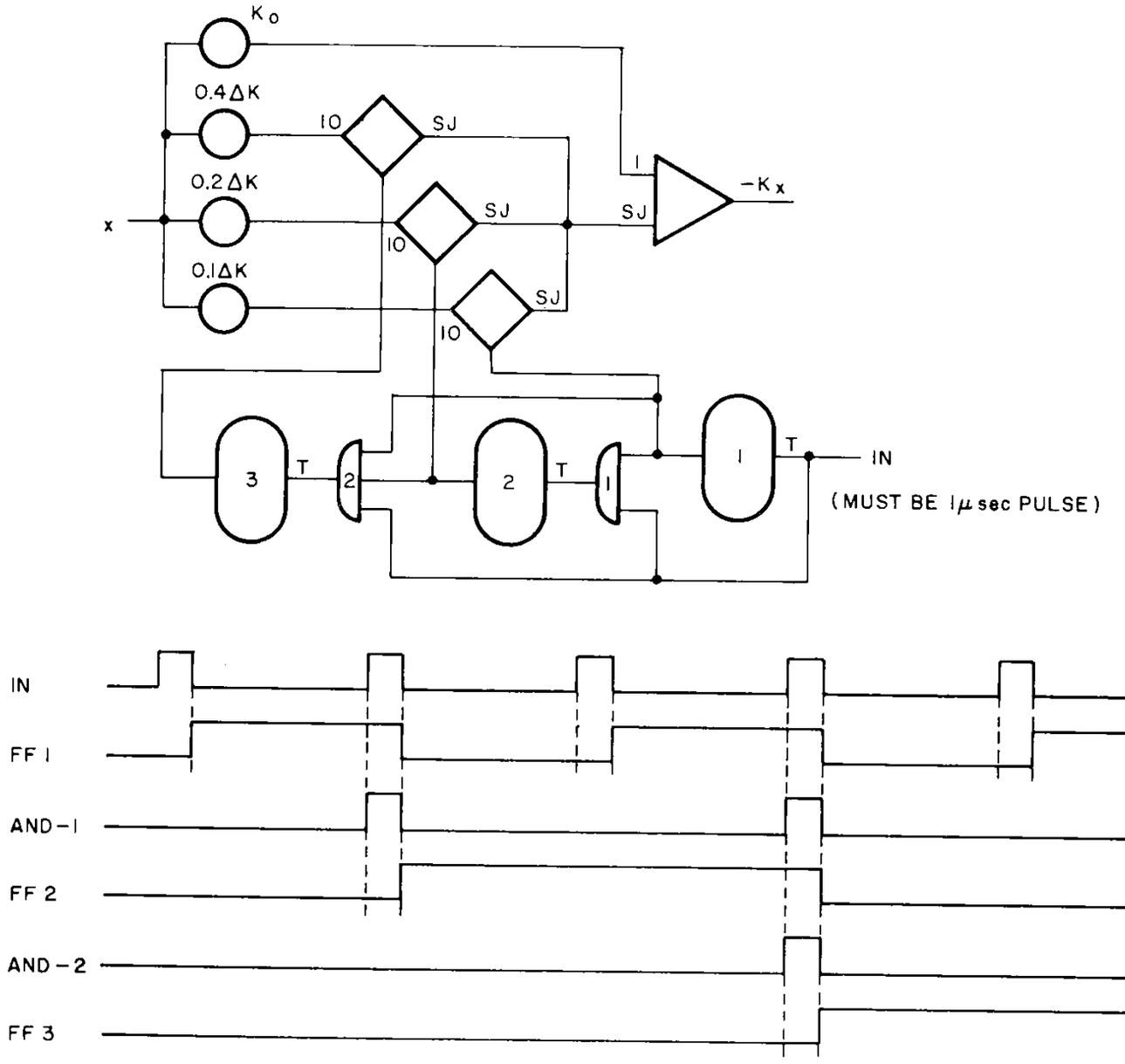


Figure 11.8. Parameter Implementation with a Counter

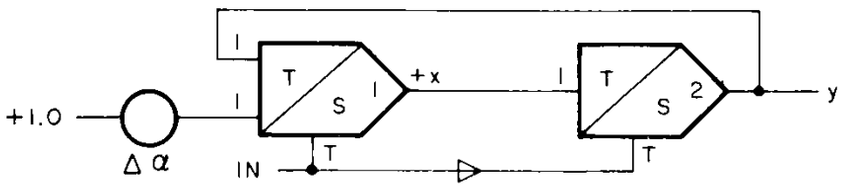
IN Pulses	Flip-Flop 3	Flip-Flop 2	Flip-Flop 1
0	0	0	0
1	0	0	1
2	0	1	0
3	0	1	1
4	1	0	0
5	1	0	1
6	1	1	0
7	1	1	1
8	0	0	0
Etc.			

In other words the combination shown is an 8-count (0-7) binary upcounter. The states of the switches produce the following output:

IN Pulses	K
0	K_o (all flip-flops are cleared)
1	$K_o + 10 (\cdot 1\Delta K) = K_o + \Delta K$
2	$K_o + 2\Delta K$
3	$K_o + 2\Delta K + \Delta K = K_o + 3\Delta K$
4	$K_o + 4\Delta K$
5	$K_o + 5\Delta K$
6	$K_o + 6\Delta K$
7	$K_o + 7\Delta K = K_{MAX}$
8	K_o
Therefore, $\Delta K = (K_{MAX} - K_o)/7$	

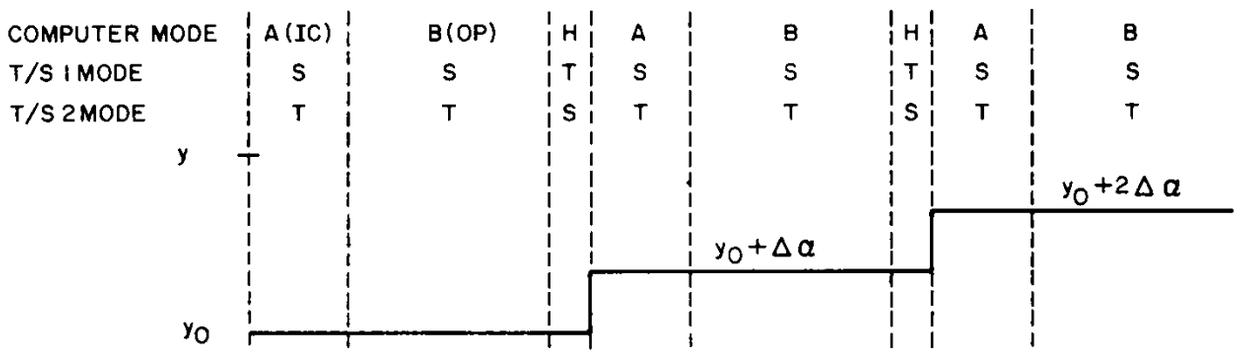
11.4.3.3 Using an Analog Accumulator

Another method for providing incremental parameter values uses a device called an analog accumulator.



IN	X	Y
1	$-\Delta a$	0
0	$-\Delta a$	Δa
1	$-2\Delta a$	Δa
0	$-2\Delta a$	$2\Delta a$
1	$-3\Delta a$	$2\Delta a$
0	$-3\Delta a$	$3\Delta a$
	etc.	

If IN is H (Hold), we have:



This circuit can also be patched using integrators as track-store units with certain limitations. Refer to Chapter 21.

11.4.4 BOUNDARY VALUE PROBLEM

The analog accumulator (Paragraph 11.4.3.3) has one restrictive characteristic: it uses up two T-S units. On the other hand, it is very useful in solving boundary problems, using proportional correction $a(i + 1) = a(i) + K\epsilon(i)$

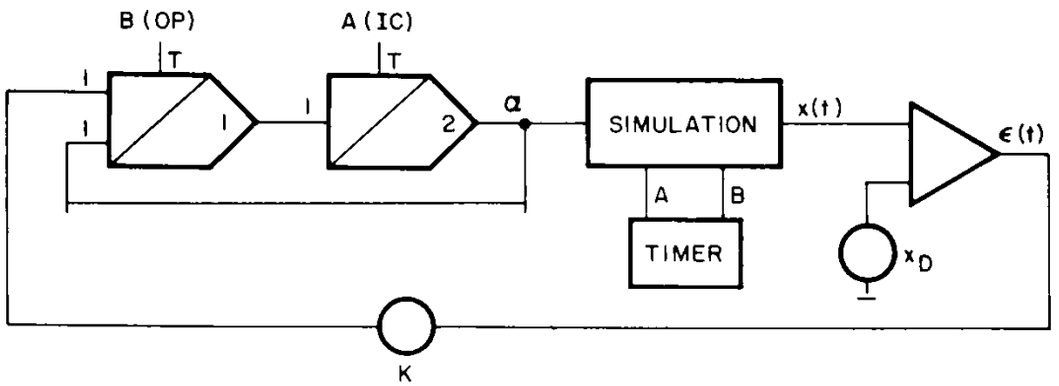
where

$a(i)$ is the value of a for the ith run.

$\epsilon(i)$ is the error in the final value of the variable at the boundary, i.e., ϵ = (x desired − x actual)/at the boundary.

and

K is the proportional constant.



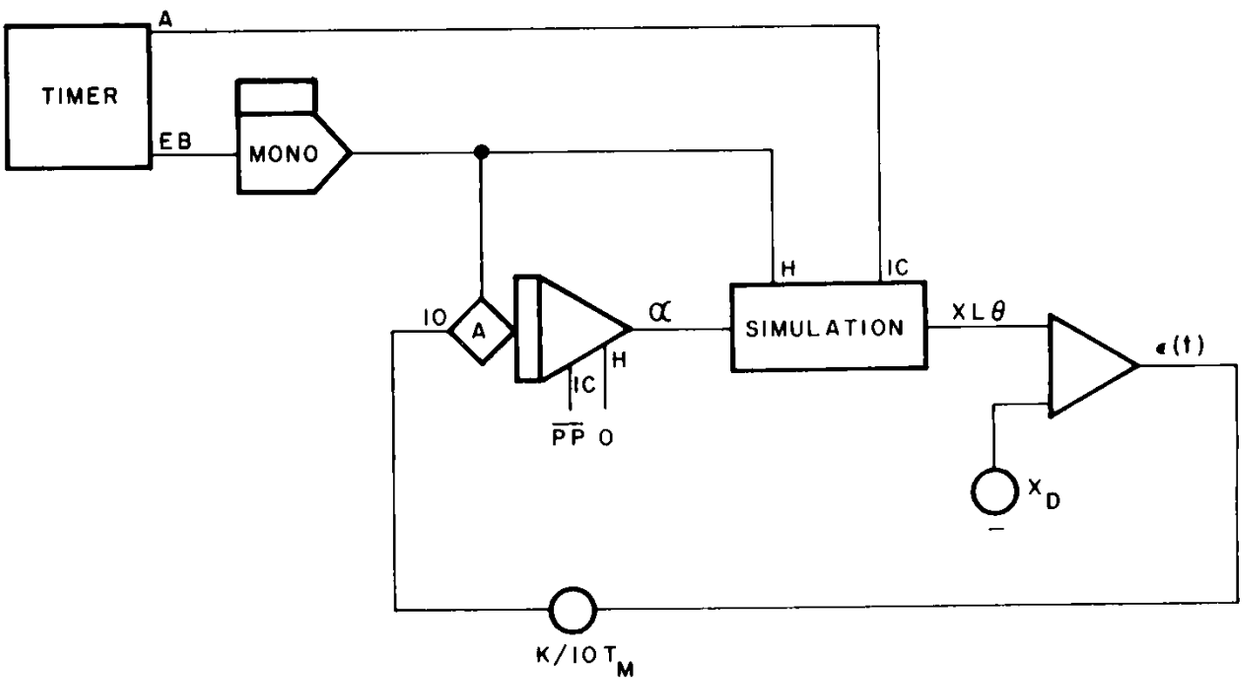
Note that the output of the summer, $\epsilon(t)$, is only meaningful at the end of B (and in HOLD). Since T-S 1 tracks during B, it has the correct $K\epsilon(i)$ at the end of B and supplies the proper $\alpha(i+1)$ to T-S 2 during A.

You may have to vary K to find the correct sign for convergence and a reasonable value for fast convergence. The sign of K is changed, of course, by adding an inverter after the summer which produces $\epsilon(t)$.

The single-integrator parameter increment circuit may also be used to solve boundary-value problems, as shown below. In this case, the entire simulation is put into the HOLD mode by the mono, so that when the mono output is high, the final value of the error is stored. During this interval, the input to the parameter updating integrator is turned on, so that the integrator output will change by an amount proportional to the input error. When the monostable period is over, the integrator input is turned off, and the simulation is put into the IC mode.

The parameter updating integrator is in the IC mode whenever the PP signal is low (i.e., whenever the iteration is being reset). Of course, any other logic signal may be used to control the resetting of the iteration scheme. The mono controls the mode of the simulation by means of the "HOLD" mode input terminal (*not* the "ORH" terminal, as this would prevent the parameter updating integrator from changing).

Note that the amount by which the parameter changes is proportional to the error, *and* to the mono interval T_M (nominally 0.5 to 1.0 milliseconds). For this reason, the pot-setting is made inversely proportional to T_M. In practice, the precise value of T_M is not significant, as the pot-setting is determined empirically in any case.



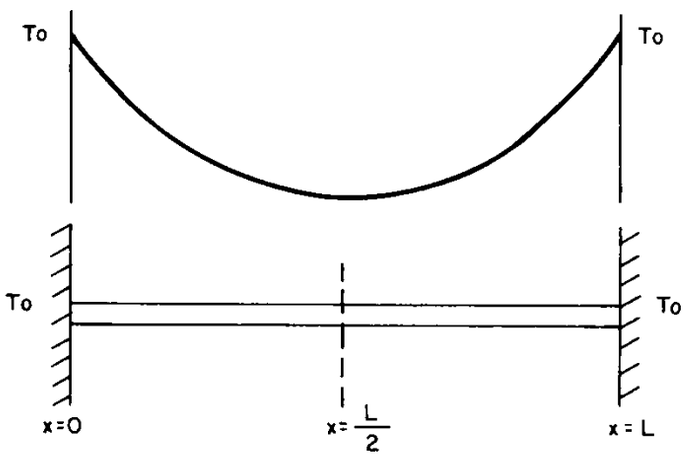
AUTOMATIC ITERATIVE OPERATION EXAMPLE (A TWO-POINT BOUNDARY VALUE PROBLEM)

The following problem is presented as a guide to programming, checking and running a typical iterative program. We will first present the problem and then "Walk" through the solution. Finally, you are asked to patch it, check it and run it, with a procedure that is documented.

12.1 THE PROBLEM STATEMENT

12.1.1 INTRODUCTION

Consider a slender, homogeneous rod of length L with both ends maintained at a constant temperature T_o, as shown below.



Assuming heat transfer by conduction and radiation, we wish to determine the steady state temperature along the rod.

12.1.2 EQUATIONS

The heat transfer equation is given by

$$\frac{d^2T}{dx^2} = kT^4, 0 \leq x \leq L \qquad (1)$$

The boundary conditions can be written as

$$T(0) = T(L) = T_o$$

Applying symmetry, the temperature gradient at the midpoint is

$$\frac{dT}{dx} = T' = 0 \text{ at } x = L/2 \qquad (2)$$

Equation (1) can be readily implemented on the analog computer. However, a problem arises because the boundary conditions are specified at opposite ends of the rod (i.e., T(L/2) is unknown) the initial gradient, dT/dx (0), is unknown.

To solve this problem, the following iteration scheme is to be employed.

$$T'_{i+1}(0) = T'_i(0) - \text{GAIN} \bullet T'_i(L/2) \tag{3}$$

Iteration will be regarded to have converged when

$$\frac{dT(L/2)}{dx} \leqslant \text{an arbitrary tolerance} \tag{4}$$

12.1.3 VALUES

The following values are to be used:

$T(0)$	=	2000 deg. Kelvin
L	=	0.5 meters
k	=	8×10^{-8} (deg. Kelvin)$^{-3}$ (meters)$^{-2}$
$\lvert T \rvert$	$\leqslant$	2000 deg. Kelvin
$\left\lvert \frac{dT}{dx} \right\rvert$	$\leqslant$	40,000 deg. Kelvin/meter

12.2 PROCEDURE

12.2.1 PREPARING THE SIMULATION

We will now designate an analog/hybrid simulation to implement the heat transfer equation and the iteration scheme. The iteration will be accomplished automatically in two-mode high-speed repetitive operation, and will include iteration reset, automatic stop when the correct IC is reached, and continuous display of temperature.

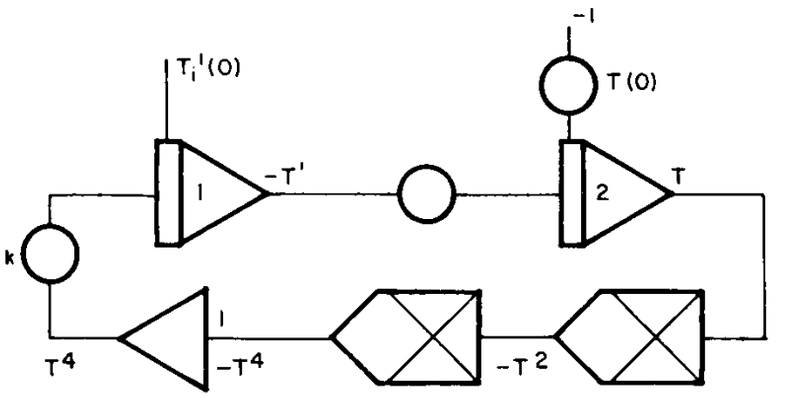
To initiate the design, let us draw an unscaled analog diagram for the heat transfer equation.

$$T'' = kT^4 \quad , 0 \leqslant x = L$$

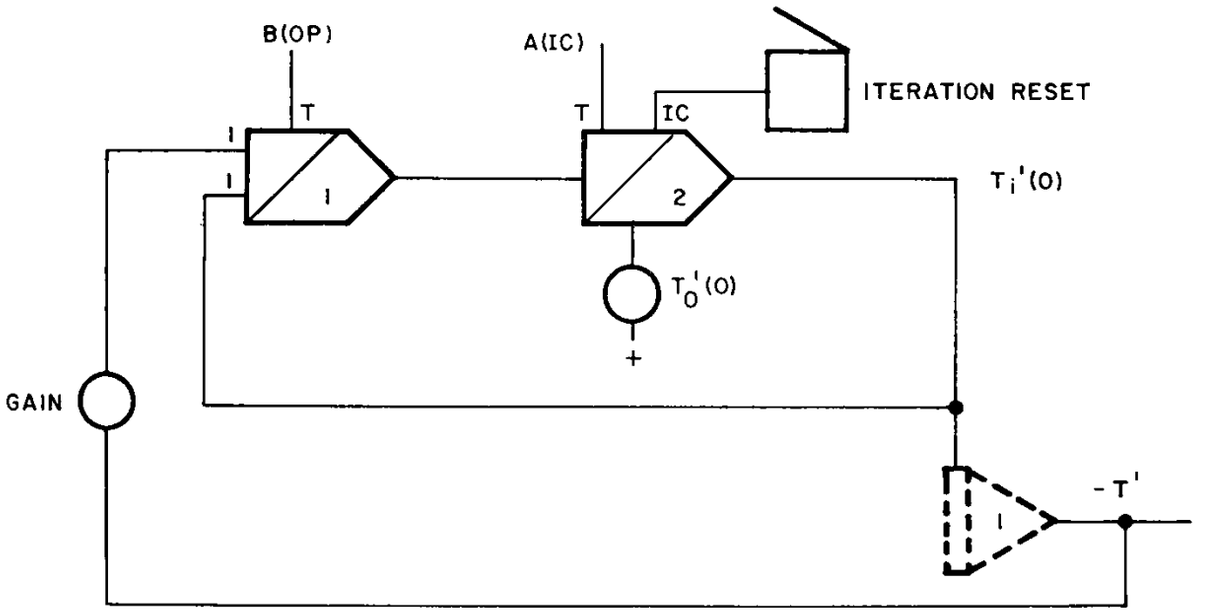
where $T' = dT/dx$

$T'' = d(dT/dx)/dx$

and x is represented by time, t, on the computer. This implies that we must carefully choose the B(OP) time to correspond to x = L/2, as we are using the symmetry of the system.



T'_i (0) is the i-th value of the unknown parameter T' (0), the initial value of T'. We know the value of T' at x = L/2 but not at x = 0. We will use an analog accumulator (Paragraph 11.4.3.3) to obtain the values of T' (0) in accordance with Equation 3.



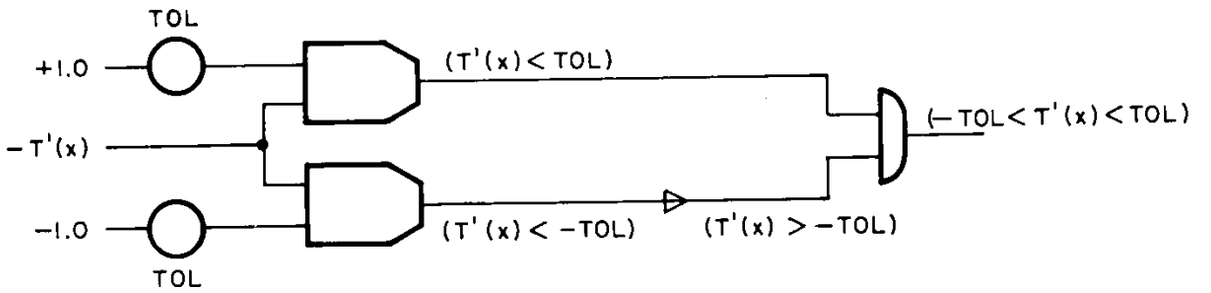
The pushbutton allows you to reset the iteration at any time. Pushing the button forces T-S 2 to the IC-mode, forcing the IC of integrator 1 to be the initial guess for T' (0).

We are also interested in setting up a program for the automatic stop feature. That is when T' (x) = 0 at x = L/2, we wish to stop iterating as we have arrived at the solution. Since zero is difficult to attain, we will use the criterion of Equation (4):

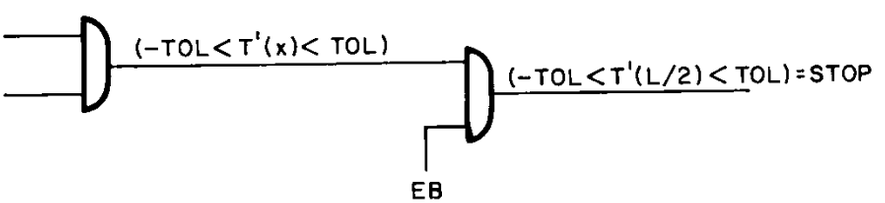
$$|T'(L/2)| \leq TOL$$

First consider

$$|T'(x)| \leq TOL \text{ or}$$
$$-TOL \leq T'(x) \leq +TOL$$

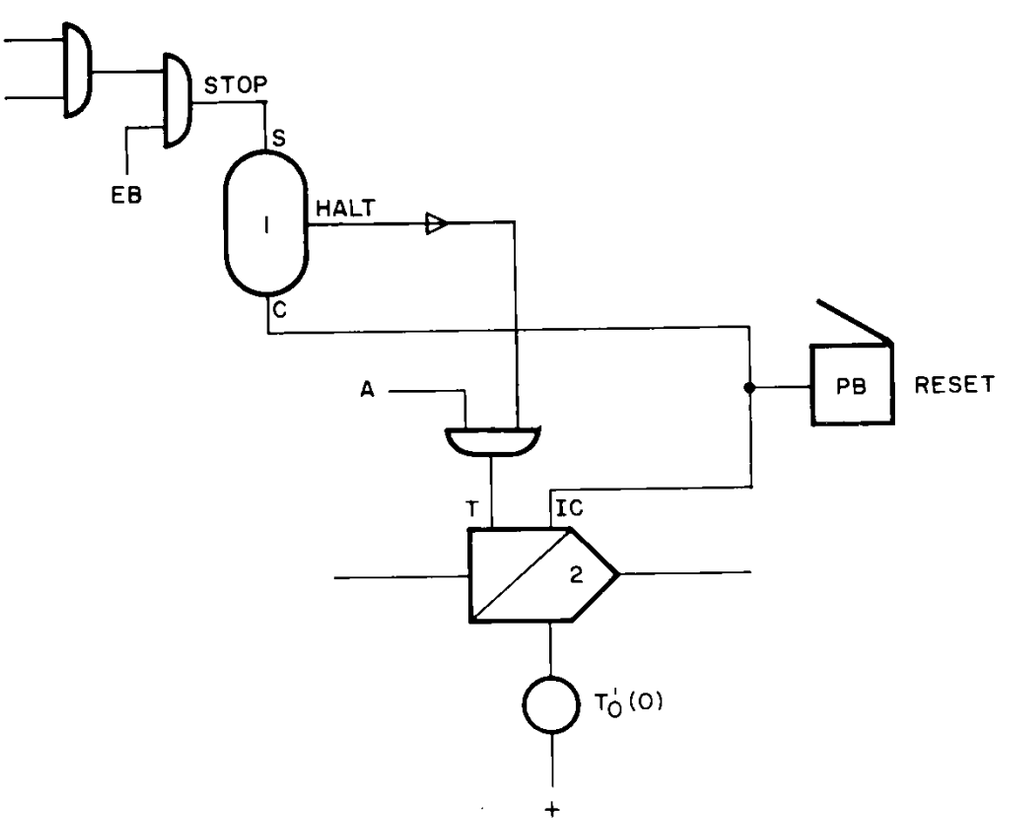


How do we now take care of (−TOL < T' (x) < TOL) at x = L/2? Recall that we chose the B (OP) time to correspond to x = L/2; therefore, the EB signal from the timer occurs at the exact instant (± 1 microsecond, negligible in the analog time scale) that x = L/2. Therefore we can add the following circuit.



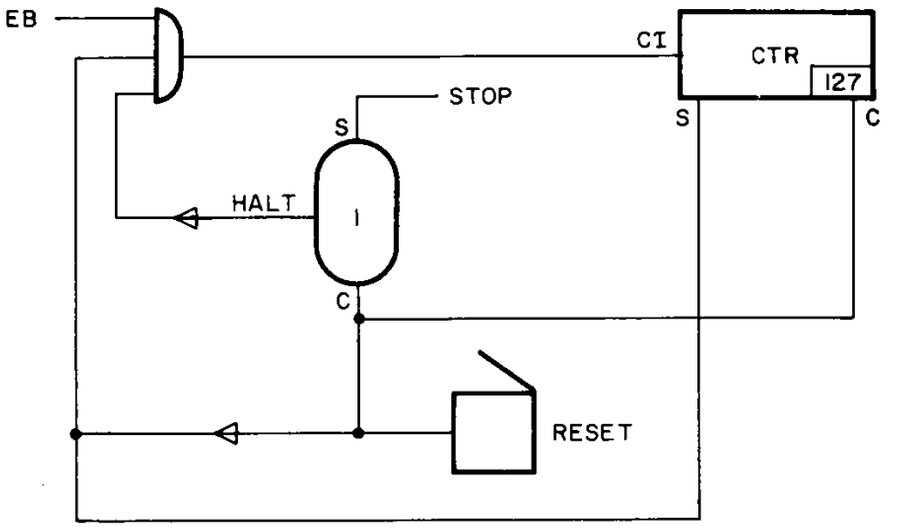
STOP is a logic signal that will be high for one microsecond at the end of each B(OP) interval, if and only if the tolerance condition is met.

Now we take the one microsecond STOP signal, change it to a logic 1 of arbitrary length (HALT) and apply it to the accumulator. Forcing T-S 2 into the STORE mode inhibits further changes in T'_i (o).



Let's see how this works. We reset the problem by pushing the PB. This clears the flip-flop and initializes T-S 2. Then we release the PB, allowing the iteration to begin. Whenever A=1, T-S 2 will TRACK; whenever A=0, T-S 2 will STORE. Now along comes a run for which $|T' - (L/2)| \leq TOL$. The short STOP pulse sets the flip-flop inhibiting the gate controlling T-S2. In other words, that gate feeds a logic 0 to T-S 2, forcing it into STORE until the flip-flop is reset!

In order to study the effects of the gain, K, on the iteration, we have included a counter to count the number of runs (up to 128) that are needed for the iteration to converge.



12.2.2 SCALING

The computer variables are

$$T/2000$$

and

$$T'/40000$$

Therefore the output of MULT 15 is $-(T/2000)^2$, the output of MULT 25 is = $(T/2000)^4$, and pot 11 is:

$$\frac{(2000)^4 k}{40000\beta} = \frac{4 \times 10^8 k}{\beta}$$

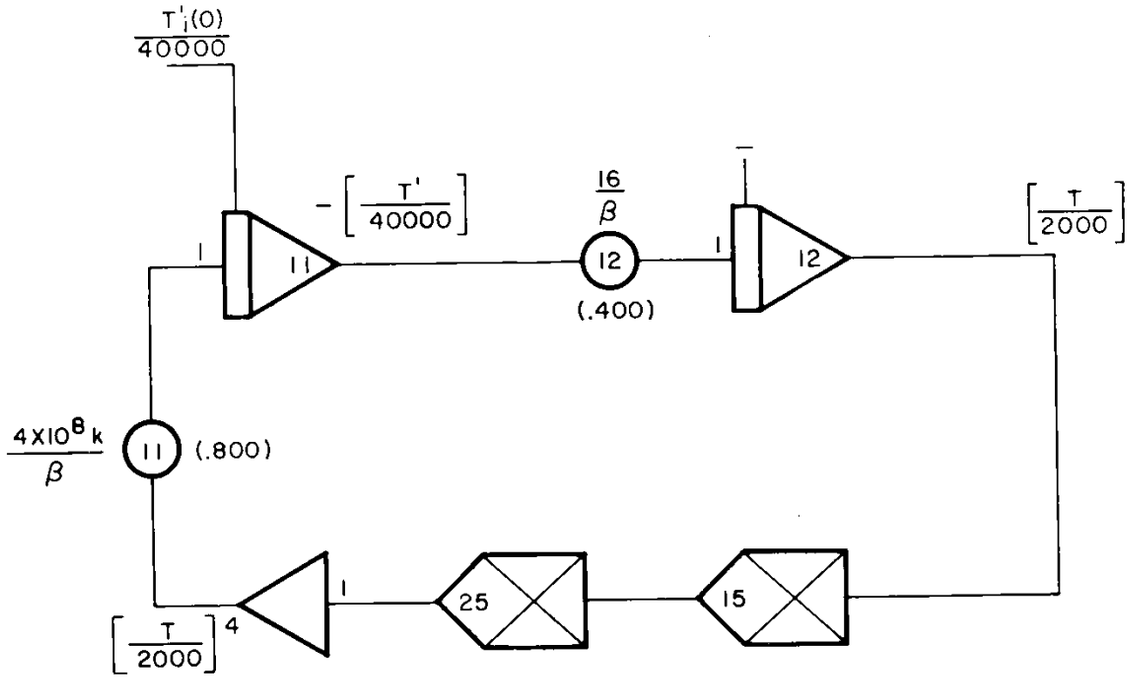
Also, pot 12 is $40000/2500\beta = 16/\beta$

β is the scale factor which allows computer time to represent the length of the bar. Suppose we choose to let 20 seconds represent the total length of the bar, L. Since L = 0.5 meters,

$$\beta = 40 \text{ sec./meter}$$

When we go to high speed rep-op, we will get a speed-up of 500 to 1 so that an OP time of 10 milliseconds will represent L/2.

Our analog diagram now looks like this:



12.3 CHECKING THE PATCHED PROGRAM

12.3.1 STATIC TEST

After all the pots have been set, a conventional static test can be made.

1. Analog mode: IC
2. Logic mode: CLR (actual mode will be STP)
3. Set PB 1 and PB 2.
4. Using Selector-1, read the outputs of the mathematical elements. If any output is incorrect, check the elements in the order that they effect each other. For example, T-S 14 provides the initial condition (IC) for ∫ 11; MULT 25 provides the input to INV 26.
5. Read the pot outputs.

12.3.2 LOGIC TEST

A partial test of the logic can be made by checking the status on the analog and lotic control panels as follows (Analog mode: IC; Logic mode: CLR):

1. COMP 13: ON
2. COMP 23: ON
3. AND-1: OFF
4. Depress COMP 13 button; COMP 13 should go OFF and AND-1 should go ON. Release the pushbutton.

5. FF1: OFF
6. FF2: OFF
7. CTR/TMR: OFF

12.3.3 DYNAMIC CHECK

1. Clear PB-2
2. Logic mode: RUN
3. Set TIMER for 10 milliseconds
4. RATE: depress FAST
5. Analog mode: PP

The oscilloscope should now display the solution of $T'' - kT^4$ from $x = 0$ to $x = L/2$ with the initial condition $T(0)$ and $T'_o(0)$.

12.4 RUNNING THE PROGRAM

All that is needed at this point is to clear PB-1. This allows the iteration to commence. If Selector 1 is at 14, the digital voltmeter will display the value of T'(0)/40000 which is presently being used.

To reinitialize and begin another iteration cycle, simply set and clear PB1. If a parameter change is to be made, proceed as follows:

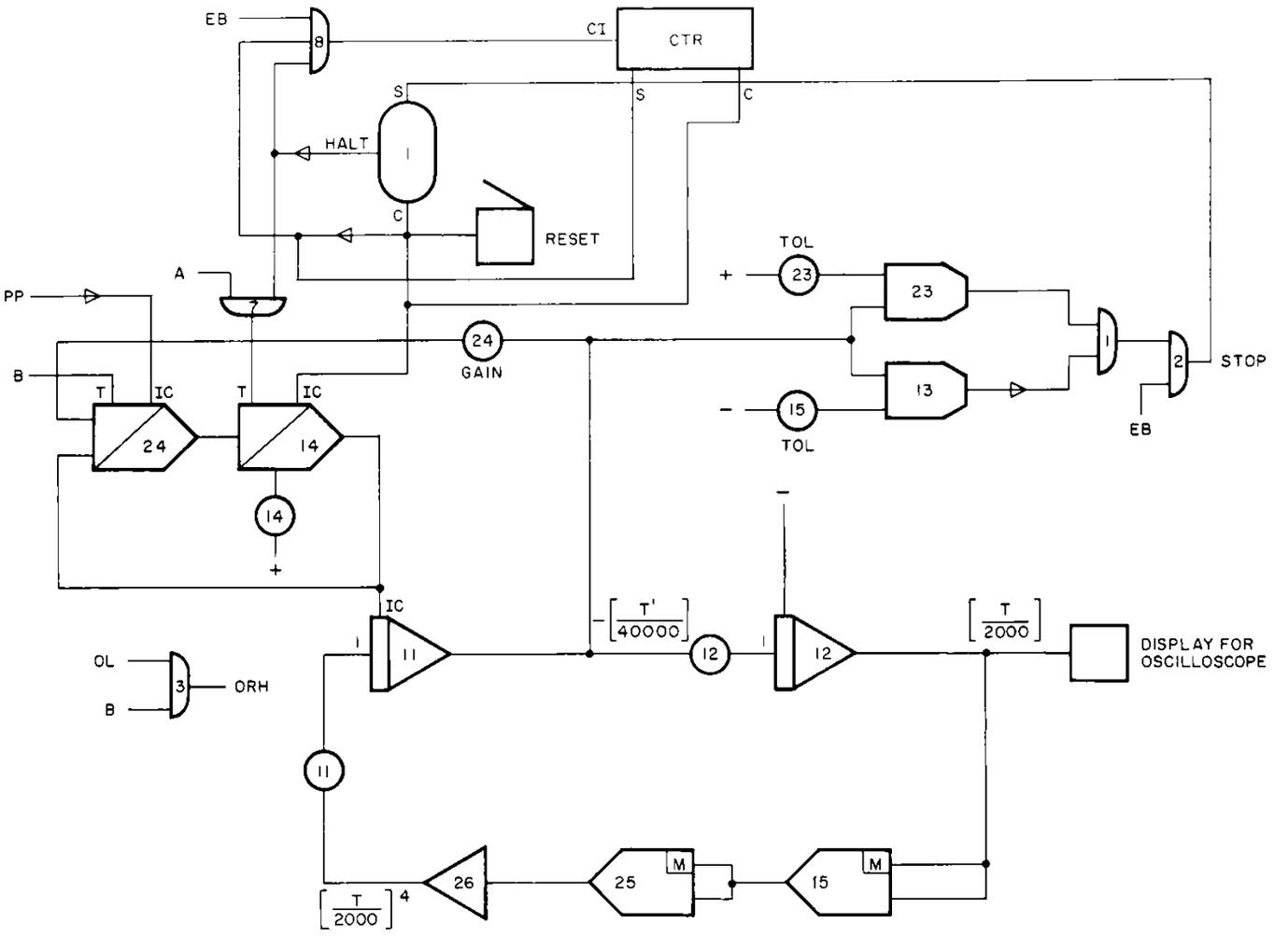
1. Logic: CLR
2. Set PB-1
3. Analog = SP
4. Now you can change K (the iteration gain) or any other parameter if you like.

   Analog = IC

   Logic = RUN

   Analog = PP
5. The computer is now in "rep-op" with T'(o) equal to whatever you have chosen as a starting point.
6. Clear PB-1 to begin iteration.

In case an overload occurs during the B (OP) mode, this fact is detected by gate 3, and the OVERRIDE HOLD (ORH) feature is activated, putting the simulation into the HOLD mode for the duration of the run. (Why not simply connect the OVERLOAD signal directly to ORH and eliminate the gate?)



NOTE:
CONNECT TIMER A OUTPUT TO PATCH PANEL IC INPUT.

EAI MiniAC™

PROGRAM _____ NAME _____

_____ DATE _____

MATHEMATICAL FUNCTION

	FUNCTION	NOTES	OUTPUT
11	Σ (I)	$-T'/4000$	.900
12	Σ (I)	$T/2000$	1.000
13	Σ HG		
14	Σ (TS)	$Ti'(0)/4000$	-.900
15	(M) D Σ	$-(T/2000)^2$	-1.000
16	LOG EXP Σ		
21	Σ ∫		
22	Σ ∫		
23	Σ HG		
24	Σ (TS)	$-Ti'(0)/40000$	0
25	(M) D Σ	$-(T/2000)^4$	-1.000
26	LOG EXP (E)	$(T/2000)^4$	+1.000
31	Σ ∫		
32	Σ ∫		
33	Σ HG		
34	Σ TS		
35	M D Σ		
36	11 20 INV		
37	11 INV INV		

COEFFICIENT

	NOTES	SETTING	OUTPUT
11	$4 \times 10^8 k/\beta$	.800	.800
12	$16/\beta$	.400	.360
13			
14	$Ti(0)/40000$	.900	.900
15	Tolerance	.005	-.005
16			
21			
22			
23	Tolerance	.005	+.005
24	Gain	.200	.180
25			
26			
31			
32			
33			
34			
35			
36			

VFG GAIN: 1 3 10 30 — X Y

GAIN: 1 3 10 30 — X Y

	VFG X	VFG Y	GAIN X	GAIN Y
0				
1				
2				
3				
4				
5				
6				
7				
8				
9				
10				

GATE

1	2	3	4	5	6	7	8	9	10
0	0	0				0	0		

FF: (1) 2 3 4

CTR/TMR: 127

PB: (1) 2

MAN SW

	16	26	36
UP			
OFF			
DN			

COMP: (13) (23) 33

RATE: (F)

TIMER: 010 X .001

NOTES _____

PUBLICATION NO. 955 0108-0

M1665

PART

PART III
REFERENCE HANDBOOK
(FOR THE EXPERIENCED USER)

CHAPTER 13

INTRODUCTION TO REFERENCE HANDBOOK

The following chapters (14 through 29) are oriented toward the frequent or relatively experienced user. This portion of the manual provides detailed descriptions of the MiniAC operations subsystem-by-subsystem and component-by-component. Sufficient information is provided to permit the experienced programmer to fully utilize all programmable features to devise unique patching schemes on a subsystem and individual component basis.

Detailed operating and patching information concerning the various MiniAC Control and Monitoring Systems are described in Chapters 14 through 17. These include the following:

Analog Mode and Time Scale Control

Logic Clock and Logic Mode Control

The Signal Selector (Readout)

The Overload and Status Indicators

Individual components, or small groups of closely related components are described in Chapters 18 through 29. Basically, each component chapter includes the following information.

Location and Identification:

- Where is it terminated?
- How is it identified?

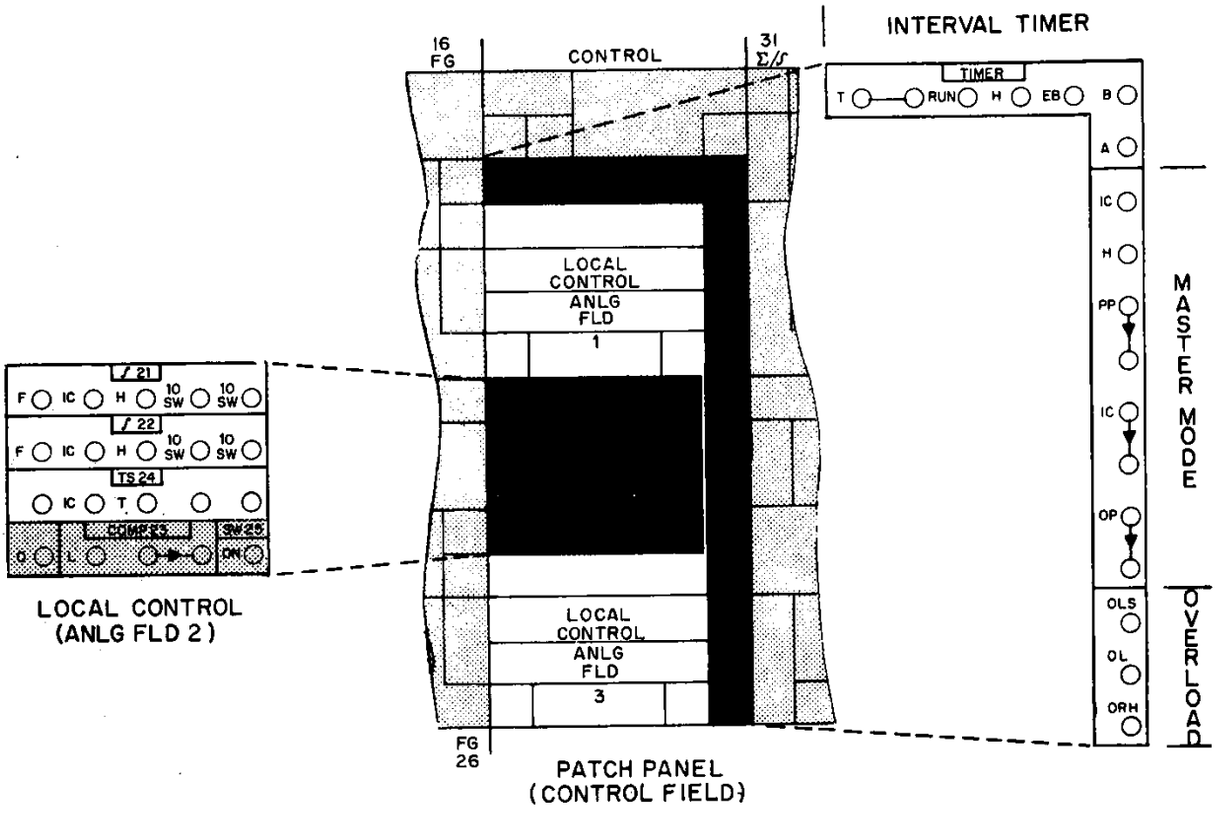
Device Description or General Operation

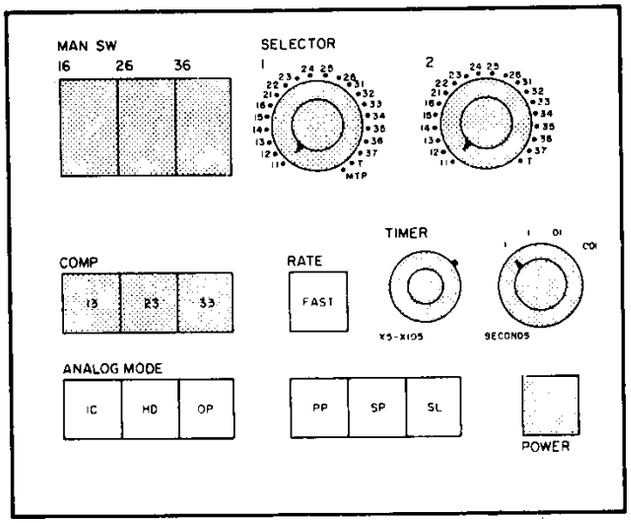
- What is it?
- How does it operate?

Device Use:

- How to use the component.

Since it is virtually impossible to describe all conceivable patching configurations and methods of operation, many of the component chapters include simplified and equivalent circuits. The simplified circuits are provided for (but not limited to) the knowledgeable user to delve into the technical aspects (at his discretion) and develop unique applications and/or patching configurations. The equivalent circuits permit the non-technical user to gain a better understanding of component operation and thereby utilize the MiniAC to its fullest capabilities.





ANALOG CONTROL PANEL

Figure 14.1. Analog Mode and Time Scale: Controls and Patching Areas

ANALOG MODE AND TIME SCALE CONTROL

14.1 INTRODUCTION

The MiniAC is equipped with a versatile analog mode and time scale control system that simplifies initial programming and subsequent program changes. Essentially, there are three computer modes, four analog modes, a special mode, and two integrator time scale selections available. The three computer modes are Master Mode Pushbutton Control, Master Mode Patch Panel Control ((PP) and the Slave Mode (SL). The four analog modes are: Set Pot (SP); Initial Condition (IC); Hold (H); and Operate (OP). SP is a "set up" mode normally used when setting potentiometer coefficients and when initially testing the program. The IC, H and OP modes are "operating" modes normally used during a problem solution. Override Hold (ORH) is a special method of operation used to "freeze" the problem solution and overrides the IC, H, and OP modes. The time scale (integrator rate) control system permits selection of either of two values of feedback capacitance for the integrators. Selection of computer time scale is made using a single pushbutton control.

Control of analog mode and time scale is performed at two priority levels (local and master). Control at the master level has the lower priority, and can be performed by pushbutton at the analog control panel, by another MiniAC Computer (when SL is selected) or by patching (when PP is selected) at the Master Mode terminations in the patch panel control field.

The local level of control permits operating mode and time scale selection of individual integrators and track/store amplifiers to be controlled by logic patching in the control field. With the exception of the Set Pot (SP) and Override Hold (ORH), local control patching overrides any master control on an individual component basis.

14.2 THE ANALOG MODES

14.2.1 SET POT (SP)

This is a pre-operational setup mode used when initializing the computer. When SP is selected +1.0 machine units (+10V) is applied to the Hi side of all pots. Additionally, switching networks in the various summers, summer/track-store units, and summer/integrators ensure that all amplifiers are provided with resistive feedback to prevent overloading. The summing junction provides a load to the input source equivalent to the load present while the computer is in the Operate (OP) mode. This permits setting potentiometers under normal load conditions.

14.2.2 INITIAL CONDITION (IC)

This operating mode is used prior to each computer run to set the output of all integrators to the predetermined initial values. Essentially, all components except integrators are in the Operate mode. The IC mode can also be used to static check the patched program.

14.2.3 HOLD (H)

The Hold mode disconnects all integrator inputs and holds the integrator outputs at their existing values. Essentially, all other components remain in the Operate mode.

14.2.4 OPERATE (OP)

In this mode, the integrators are released and respond to their respective input levels. Placing the computer in OP initiates the dynamic solution to a problem.

14.2.5 OVERRIDE HOLD (ORH)

This special method of operation overrides any selected local or master mode to force all integrators into the Hold mode. ORH can only be controlled by patching and is primarily used to freeze the problem solution when an overload occurs.

14.3 THE TIME SCALES

The time scale control system permits selection of either of two different values of feedback capacitance for all integrators terminated at the patch panel. The value of feedback capacitor for any given integrator determines the integration rate.

Control of the analog time scales is performed at both levels (master and local). The control circuits operate relays in the integrators that select the value of feedback capacitance. This is accomplished using two time scale ranges (Normal and FAST). A one second time scale is equivalent to an input of one machine unit (1 MU = 10V) to a gain-of-one or: the rate of integration is 1.0 machine unit/second. If the time scale is 2 milliseconds, the integration is 1 machine unit/2 milliseconds. Table 14.1 defines the basic correlation between the time scale range, the input gain, the time scale, and the capacitor values for each.

Table 14.1. The Time Scales

Time Scale Range	Capacitor Value	$\int$ Input Gain	Time Scale (1 MU per unit of time)
Normal	1.0 μF	1	1.0 second
		10	0.1 second
Fast (F)	0.002 μF	1	2.0 milliseconds
		10	0.2 milliseconds

14.4 PUSHBUTTON MODE AND TIME SCALE CONTROL

With the exception of Override Hold (ORH), all analog modes of operation and computer time scale can be selected manually using pushbuttons on the Analog Control Panel (Figure 14.1). The four pushbuttons designated IC, HD, OP, and SP correspond to the setup and operating modes described in Paragraphs 14.1 and 14.2. The pushbutton designated RATE-FAST is the time scale selector. When released (lamp extinguished), the normal time scale range (Paragraph 14.3) is selected. When depressed (lamp lit), the time scale is 500 times faster than normal. Depressing the appropriate pushbutton selects the desired mode and time scale of this computer, and also controls the mode and time scale of a remote MiniAC when a remote computer is in the SL (Slave) mode.

Prior to operating the analog pushbuttons, ensure that this computer is selected as the "master" (the SL pushbutton is not depressed). Mode and time scale cannot be controlled locally by pushbutton when this console is slaved to another. Details of slave operation are given in Paragraph 14.7.

The following procedure outlines the typical sequence in which the analog pushbuttons are operated.

1. Ensure that all device setup switches are properly set.
2. Ensure that the SL pushbutton is released. This computer has control.
3. Depress SP and set all patched pots to the coefficient values required by the program. Pot setting procedures are given in Chapter 18.

4. Select the appropriate time scale per program requirements (Paragraph 14.3). When the FAST pushbutton is depressed (lamp lit) the computer time scale is 500 times faster than normal.
5. Depress IC to place all patched integrators at their initial values.
6. Verify the predetermined voltage values existing at the various check points in the program. See Users Guide, Chapter 8.
7. When ready to start the program run, depress OP.
8. If at a given point in time (or after some event) it is desired to examine (readout) any integrator(s), depress HD.
9. If the program is to be continued, depress OP. If it is desired to restart the program, depress IC and then OP.
10. At the end of a program run, or when the desired results have been obtained, depress IC or SP.

14.5 PATCH PANEL MASTER MODE CONTROL

14.5.1 GENERAL OPERATION

The master mode input terminations designated IC and H (Figure 14.1) operate in conjunction with the PP (Patch Panel) pushbutton to control the three basic analog modes (OP, H and IC). When patch panel control is selected, (PP pushbutton depressed) the IC and H input terminations respond to logic inputs (1 = +5 volts and 0 = 0 volt) to permit the analog program to be controlled by logic patching. Note that mode control patching at the local level overrides master mode patching and pushbutton control on an individual basis. Using the patch panel to obtain master mode control has numerous useful applications; particularly in problems requiring repetitive or iterative operation.

14.5.2 THE INPUT CONTROL TERMINATIONS

When the PP pushbutton is depressed (lamp lit), the green input terminations (IC and H) respond to logical ones and zeros to select the computer mode of operation. The logic control inputs may be from any logic level source of predetermined duration on the patch panel. Table 14.2 is a truth table showing the logic signals (1 or 0) required at the IC and H inputs to select any one of the three operating modes.

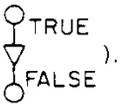
Table 14.2. PP Master Mode Truth Table

Mode	Input Logic Signals		Notes
	IC	H	When unpatched IC termination is normally high (logic 1), and H termination is normally low (logic 0).
IC	1	0	
OP	0	0	
H	0	1	
	1	1	

An important consideration when using the master mode input terminations for control is the fact that when unpatched, the IC input termination is normally high, while the H termination is normally low. This permits switching between IC and OP with a single control input patched to IC. In other words, for normal two-mode rep-op control (Paragraph 14.5.5), the H termination is left unpatched.

If the overall program requires automatic selection of all three operating modes (such as during iterative operations, Paragraph 14.5.6), the H termination as well as IC must be patched into the program. With H patched, the integrators will enter the HOLD mode whenever H goes high.

14.5.3 THE OUTPUT TERMINATIONS

There are three pairs of output terminations (PP, IC and OP) associated with patch panel mode control. Each termination pair provides a true and a false logic output as indicated by the logic inverter symbol (

).

The false output is the complement of the true output.

The true IC and OP outputs go high (logical ONE) whenever either of the corresponding analog modes (IC or OP) are selected. Table 14.3 shows the IC and OP output levels present during each computer mode of operation.

Table 14.3. Master Mode Patch Panel Outputs

Mode	Output Termination			
	IC		OP	
	True	False	True	False
IC	1	0	0	1
OP	0	1	1	0
H	0	1	0	1
SP	1	0	0	1

Unless slaved to another computer, these logic level outputs are synchronous and can be used in numerous applications such as; operating and controlling a given logic function, starting a logic sequence when a given analog mode is selected, and even providing a start signal to an XY plotter or some other external device.

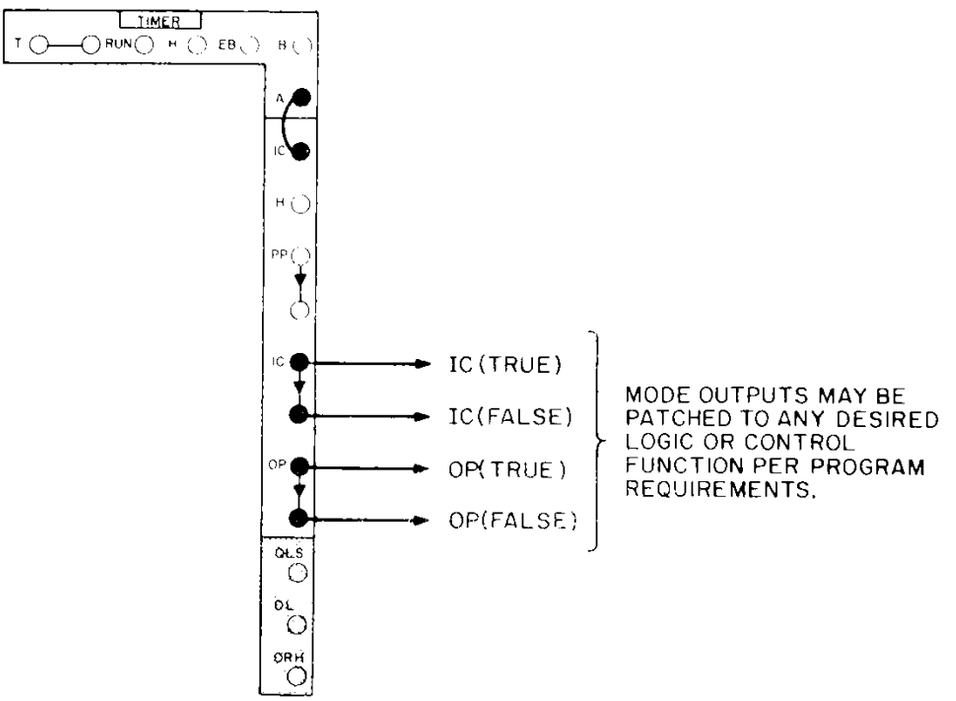
The true output of the PP (patch panel) termination only goes high when the PP pushbutton is depressed. This output remains high until another mode is selected manually. The PP outputs can be used to enable any desired logic function, or as a start signal.

14.5.4 SELECTING PATCH PANEL MODE CONTROL

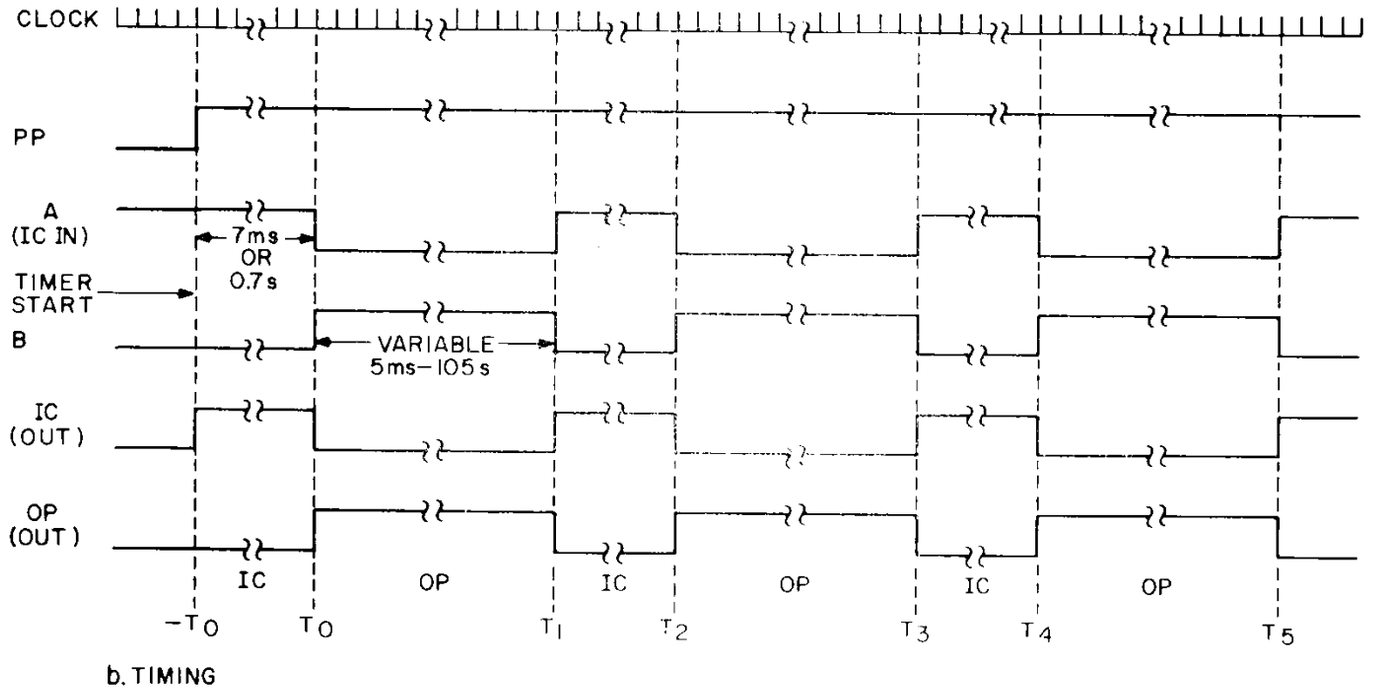
Patch Panel Mode Control is normally used to obtain repetitive or iterative operation (see Paragraphs 14.5.5 and 14.5.6). Depressing the PP pushbutton will initiate the program run unless the internal timer run termination is patched low (Chapter 29).

14.5.5 REPETITIVE OPERATION

Figure 14.2 shows the method of using the Master Mode input terminations to obtain repetitive operation (rep-op). The goal in rep-op is to automatically cycle the computer between the analog IC and OP modes at some predetermined rate. This permits problem solutions to be displayed on an oscilloscope. To accomplish this, the internal for rep-op timer (Chapter 29) must be used as the mode control element. The timer provides two time intervals A and B. The duration of each interval is determined by the timer control settings. Interval A can be either of two durations (about 7 milliseconds or about 0.7 second) depending on speed selection. Interval B is variable and can be precisely set at any duration from 5 milliseconds up to 105 seconds as required by the overall program.



a. PATCHING



b. TIMING

Figure 14.2. Typical Repetitive Operation: Patching and Timing

As shown in Figure 14.2, the A and B intervals go high in turn and are never high (true) at the same time. Interval B goes high when interval A drops and conversely, A goes high when B drops. The A interval corresponds to the computer IC mode and the adjustable B interval corresponds to the OP mode.

With the patching shown, the computer is forced to the IC mode when A goes high, and into the OP mode when A goes low. When the PP pushbutton is depressed (lamp lit), the timer starts and the problem run is initiated. If it is desired to start the problem from a source other than PP, patch RUN to the output of the device selected for control. When the run input goes high (and if PP's is selected) the timer will start.

14.5.6 ITERATIVE OPERATION

The Master Mode input terminations can be patched to automatically control all three operating modes for use in iterative operations. Iterative operation is similar to rep-op (Paragraph 14.5.5) and is generally used to start a normal analog sequence and then place the computer in Hold while a parameter is updated or a logic sequence is performed. The problem in this case is to place the computer in IC at time minus zero ($-T_0$), and at a given time (T_0) switch to the Operate mode (starting the analog program). Then at some other specific point in time (T_1) place the computer in Hold. At the end of the Hold interval, the computer will return to the IC mode and the entire sequence of events will repeat.

Figure 14.3 shows the patching and basic timing requirements for automatic three-mode iterative operation. Note that the monostable multivibrator (MONO) must be patched to provide the Hold mode or H interval. With the patching arrangement shown, depressing PP starts the timer and initiates the problem run.

As in the case of rep-op: interval A is patched to the IC control input; intervals A and B go high in turn; one microsecond before interval B ends, EB (end of B) is generated. This one microsecond pulse triggers the MONO (Chapter 29). The MONO output goes high for .5 to 1.0 millisecond and is used to force the Master Mode H control input high. When H goes high, the computer enters the Hold mode. Note that A is also high. When the MONO times out, the computer returns to the IC (A = 1) mode and the entire sequence is repeated.

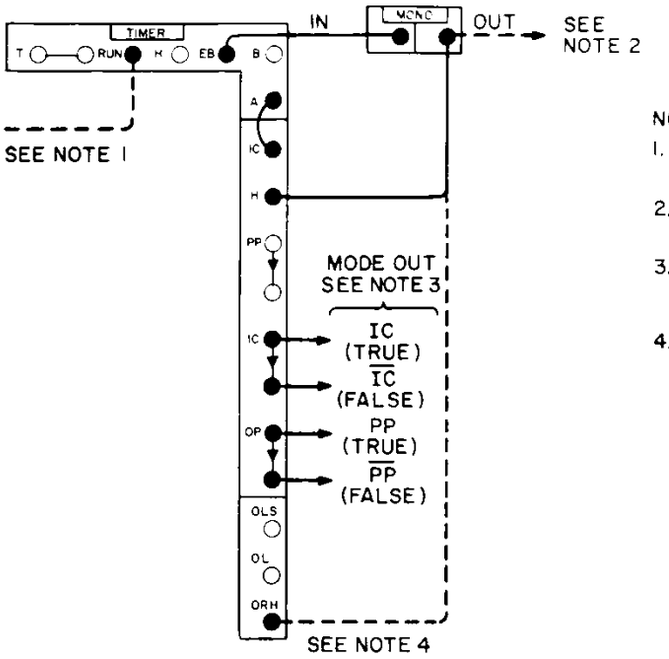
14.5.7 OVERRIDE HOLD AND OVERLOAD STORE OPERATIONS

14.5.7.1 Introduction

Override Hold (ORH) is a special method of operation that forces all integrators into the Hold (H) mode. ORH can only be controlled by patching and completely overrides all master and local control. Override Hold can be used in any application where it is desired to place all integrators in Hold regardless of local mode control patching.

In many problem solutions it may be desirable to place the computer in Hold when an overload occurs. Overload Store (OLS) provides a programmable means of detecting and storing an overload. When OLS is used, the first overload forces the master overload indicator to light. This indicator will remain lit until the overload condition is corrected and OLS is reset.

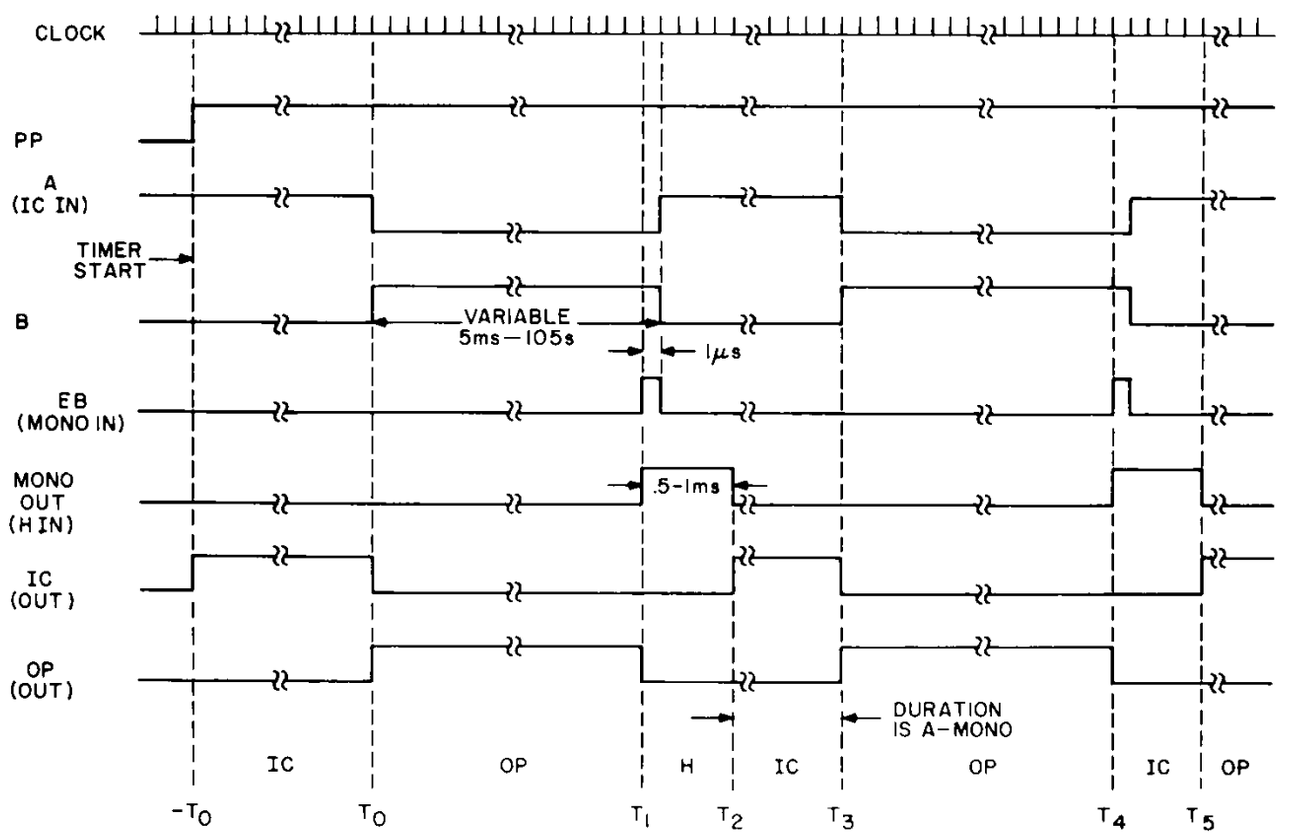
Combining ORH and OLS permits storing the overload and freezing the problem solution before other amplifiers overload. With the problem frozen at the values attained when the overload occurred, the operator can analyze and correct the condition causing the overload, or at least determine its affect on the problem solution.



NOTES:

1. RUN INPUT CAN BE USED TO PREVENT TIMER FROM STARTING WHEN PP IS DEPRESSED.
2. MONO OUT CAN BE USED TO INITIATE LOGIC FUNCTION.
3. MODE OUT MAY BE PATCHED TO ANY DESIRED LOGIC OR CONTROL FUNCTION PER PROGRAM REQUIREMENTS.
4. IF ONE OR MORE ƒ ARE UNDER INDIVIDUAL (LOCAL) CONTROL, PATCH ORH TO MONO OUT. THIS WILL ENSURE THAT ALL ƒ ENTER HOLD REGARDLESS OF LOCAL PATCHING. LEAVE H PATCHED TO MONO OUT SO THAT MODE OUTPUTS WILL INDICATE HOLD.

a. PATCHING



b. TIMING

Figure 14.3. Typical Three-Mode Iterative Operation: Patching and Timing

The ORH and OLS patching area is directly below the Master Mode output terminations as illustrated in Figure 14.1. The OL (overload) and OLS (overload store) terminations are directly related to the overload system. The ORH termination is a control input and is independent of the overload system. ORH is normally low, and when patched high, forces all integrators into the HOLD mode. OLS is the control input used to force the overload indicator to remain lit. The OL termination is a logic output that goes high whenever an overload occurs. The following Paragraphs (14.5.7.2 through 14.5.7.4) outline the various methods of performing ORH and OLS operations. Basic patching is illustrated in Figure 14.4.

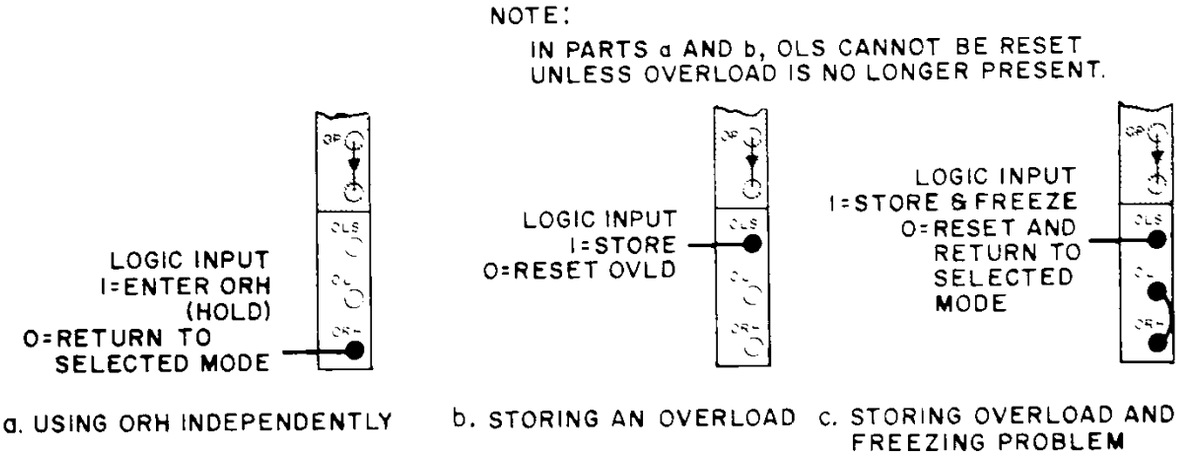


Figure 14.4. Basic ORH and OLS Patching

14.5.7.2 Using ORH Independently

Override Hold can be used independently of the overload functions to force all integrators into the Hold Mode at some given point in time (and/or for a given time period), or after a given logic event has occurred. A prime example of using ORH in this manner is during iterative operations where the computer must cycle repetitively through the IC, OP and H modes (Paragraph 14.5.6). In some iterative applications, the Master Mode H termination can be patched to automatically select the Hold mode. However, if any one or mode integrators are under individual (local) control, ORH must be used to ensure that all integrators enter Hold during the H interval. This is accomplished by patching the logic control signal tha represents the H interval to the ORH termination (Figure 14.4a). Then whenever ORH goes high, all integrators are automatically forced into Hold.

14.5.7.3 Storing an Overload

As previously mentioned, OLS can be used to store an overload. This can be accomplished by simply patching OLS high (Figure 14.4b). Then, whenever an overload occurs, the System Overload indicator will light and remain lit. Additionally, the output at the OVL termination will remain high. This method of using OLS only provides an indication that an overload has occurred and does not prevent other amplifiers from subsequently overloading. When patched high, OLS can only be reset by removing the control level patched to OLS, and if the overload is other than momentary, correcting its cause. Using the output of a pushbutton flip-flop (Chapter 27) to select OLS provides a convenient means of resetting the OLS function.

14.5.7.4 Storing an Overload and Freezing the Problem

To store an overload and freeze the problem solution, all three terminations (OLS, OL and ORH) must be patched as illustrated in Figure 14.4c. Note that OL is patched to ORH. Therefore, when an overload occurs OL goes high and

enables ORH. Since OLS is patched high, OL remains high and all integrators enter and remain in the Hold mode. The integrators can only be returned to the previous operating mode when OLS goes low. Using a pushbutton flip-flop to generate OLS provides a convenient means of resetting OL and returning the integrators to the selected mode.

14.6 LOCAL PATCH PANEL CONTROL

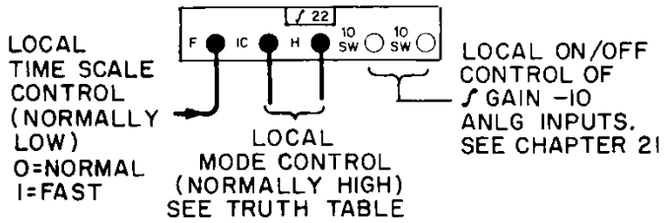
14.6.1 INTRODUCTION

The local control logic permits each integrator and track/store summer to be controlled at the patch panel, and on an individual basis. With the exception of the Set Pot (SP) and Override Hold (ORH), local control has the highest order of priority, and when selected, overrides all other methods of master control. The local control patching areas are located in the patch panel control field as illustrated in Figure 14.1. Local control of integrator mode and time scale is limited to the three operating modes (IC, OP and H) and the two time scales (Normal and FAST). Control of track/store summers is limited to three modes (Track – T, Store – S, and IC). Note that control facilities for comparators (in modules 13, 23, and 33) and D/A switches (modules 15, 25 and 35) are also provided in each local patching area. Since local control and general operation of D/A switches and comparators is not directly related to the computer modes, control of these elements is not described in this chapter. For control and patching information concerning D/A switches and comparators, refer to Chapter 28.

14.6.2 LOCAL CONTROL OF INTEGRATORS

Local patch panel control of integrators provides a program feature that permits individual integrators to operate at a different time scale and/or in a different mode then that selected at the master level. In other words, locally controlled integrators can be forced to operate in a specified mode or time scale while all other integrators (not patched at the local level) follow the master program. Individual integrator control patching areas (Figure 14.1) are identified by the integrator symbol ($\int$), and the numerical module address. This symbol and address corresponds to similar designations in the analog fields. For example, the $\int 21$ local patching area controls the integrator designated $\Sigma/\int -21$ in the analog field. There are five local control input terminations for each integrator. The terminations designated 10 SW are controls for switching gain-10 analog inputs to the integrator. Refer to Chapter 21 for their use and operation. The termination designated F responds to logic level inputs to select either the Normal or Fast time scale. This termination is normally low and if left unpatched, the integrator time scale is controlled by the master time scale, FAST pushbutton. Patching a constant logic ONE to this termination (Figure 14.5a), forces the integrator to operate at the Fast rate, regardless of the master selection. If a constant logic ZERO is patched to F, the integrator is forced to operate at the Normal rate. If a changing logic level is patched to F, the selected integrator can be made to change time scale at any given point in the program. Integrator modes are controlled locally by patching the IC and H terminations. Figure 14.5a illustrates the basic patching requirements for local mode control and defines the logic levels required for each operating mode.

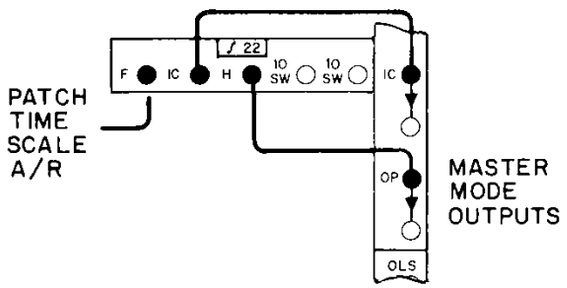
The input logic levels used to control individual integrators may be patched from any logic source available at the patch panel. If for example, the mode selection is dependent upon the analog program, the true output of a comparator (Chapter 28) may be patched to the appropriate IC or H termination. If under logic program control (such as switching analog modes on completion of a sequence of logic events), the true output of any applicable logic element may be patched to these terminations. If the goal is to simply switch the individual integrator(s) to Operate while all other integrators are in Hold (a common technique), the local control inputs may be patched to the Master Mode output terminations as illustrated in Figure 14.5b. With this arrangement, *all* integrators enter IC when the Master IC mode is selected. When OP is selected, all integrators with unpatched controls enter OP and the individually patched integrators enter Hold (retain their IC values). When the Master H mode is selected, all unpatched integrators enter hold and those with patched control inputs start to integrate.



a. BASIC PATCHING

LOCAL MODE TRUTH TABLE

LOCAL MODE	LOGIC INPUTS		NOTES
	IC	H	
IC	1	0	TO SWITCH BETWEEN IC AND OP, PATCH CONTROL SIGNAL TO IC AND PATCH H TO CONSTANT LOGIC ZERO FOR THREE MODE CONTROL, BOTH IC AND H MUST BE PATCHED WITH CONTROL INPUTS.
OP	0	0	
H	0	1	



b. TYPICAL USE

LOCAL MODE VERSUS MASTER MODE TRUTH TABLE

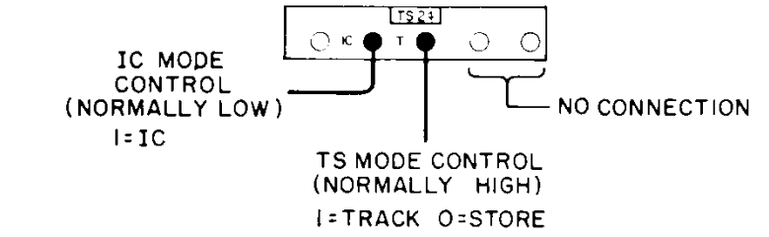
MASTER MODE	MM OUTPUTS		INDIVIDUAL INTEGRATOR		
			LOGIC INPUTS		LOCAL MODE
	IC	OP	IC	H	
IC	1	0	1	0	IC
OP	0	1	0	1	H
H	0	0	0	0	OP

Figure 14.5. Integrator Mode and Time Scale: Local Control Patching

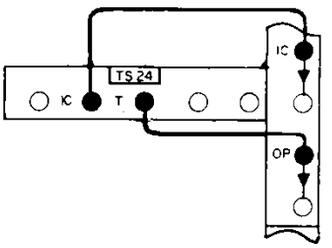
14.6.3 LOCAL CONTROL OF T-S AMPLIFIER

When operated in the Track/Store function, Summer/Track-Store units (in modules 14, 24, and 34) can be controlled individually at the patch panel. As shown in Figure 14.1, each unit has two logic inputs (IC and T) terminated in the control field (Figure 14.1). The individual track/store control patching areas are identified by the designation TS, and the numerical module address. This designation and address corresponds to similar designations in the analog fields. For example, TS24 local patching area controls Σ/TS-24 in the analog field.

When unpatched, the termination designated IC (Figure 14.6a) is low. If patched high, the T-S summer will enter the IC mode. This corresponds to the integrator IC mode and permits the TS summer output to go to a predetermined initial condition value. The termination designated T controls the track and store modes. T is normally high and when unpatched maintains the unit in the track mode. When in Track, the unit acts like a summer in that it follows or tracks the analog inputs. Refer to Chapter 20 for more detailed information concerning the track/store function. The Store mode is comparable to the Hold mode of integrators. With the IC control input low or unpatched, the unit can only enter Store (S) when the T input goes low. As in the case of integrator local control, the T-S control inputs (IC and T) may be patched from any logic source available at the patch panel, depending on program functions. It is sometimes desired to have the T-S unit switch mode in step with the computer modes of operation. This is accomplished by patching the IC and T control inputs to the Master Mode IC and OP output terminations as shown in Figure 14.6b.



a.BASIC PATCHING



b.TYPICAL USE

LOCAL TS MODE VERSUS COMPUTER MASTER MODE TRUTH TABLE

MASTER MODE	MM OUTPUTS		INDIVIDUAL T/S		
			LOGIC INPUTS		MODE
	IC	OP	IC	T	
IC	I	O	I	O	IC
OP	O	I	O	I	TRACK
H	O	O	O	O	STORE

Figure 14.6. Track-Store: Local Control Patching

14.7 COMPUTER SLAVING AND ANALOG TRUNKS

The MiniAC has provisions that permit this console to be controlled by another analog computer. Basically, all slave operations are performed at the master control level and any local ORH patching or local control (individual) patching on the slaved computer overrides the master.

When slaving this console to another, merely depress the ANALOG MODE–SL pushbutton (Figure 14.1). This transfers control of the IC, OP and H modes and the Normal and FAST time scales to the other computer. The SL pushbutton must be depressed a second time (released) to re-gain control locally. If during a slave operation, the interval timer in the slaved computer is used in the problem, the PP pushbutton must also be depressed (lamp lit).

When using the MiniAC in a two console system (slaved or otherwise), the TRUNK 1-12 terminations (Figure 2.2) are normally used for exchanging analog data between consoles.

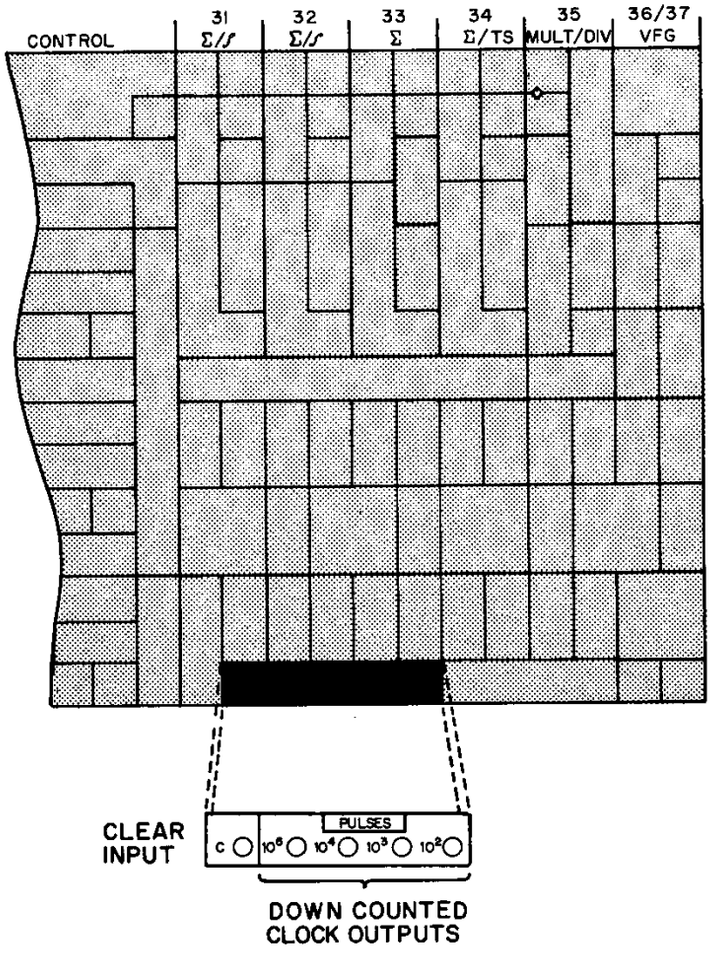


Figure 15.1. Logic Clock and Logic Mode: Patching Areas and Controls

LOGIC CLOCK AND LOGIC MODE CONTROL

15.1 INTRODUCTION

The MiniAC logic clock and logic mode control system provides synchronization and timing between the analog and logic devices. The clock is crystal controlled and operates at a fundamental frequency of one megahertz. A frequency divider and downcounter provide output clock pulses at rates of 100 kilohertz (10^5), 10 kilohertz (10^4), 1 kilohertz (10^3) and 100 hertz (10^2).

Three logic modes (Run, Stop and Clear) control the clock outputs as well as the various logic elements. The Run mode enables the output clock pulses, while the Stop mode inhibits them. The logic elements (flip-flops and counter/timer and downcounted clock) are reset in the Clear mode.

Control of the logic modes and the downcounted clock outputs is performed at a level equivalent to the master level of analog mode control (Chapter 14). Local control is not applicable. The logic modes are controlled by pushbuttons, at the local logic control panel (Figure 15.1). The down counted clock outputs are controlled by the selected logic mode of operation and can be inhibited by patching in the logic field.

15.2 THE LOGIC MODES

15.2.1 RUN

This operating mode applies the one megahertz system clock to all logic elements (except AND gates). When in the Run mode, all clocked logic elements will operate in synchronization with the clock. Generally the Run mode can be compared to the analog Operate (OP) mode, and is synchronized with the OP mode.

15.2.2 STOP/STEP (STP)

The Stop/Step mode disconnects the clock from all clocked logic elements (including the clock downcounter), permitting these elements to hold the state (condition) attained at the time the Stop mode is selected. In general, this mode may be compared to the analog Hold (H) mode. When in the Stop mode, the logic sequence can be stepped manually for program checkout and maintenance purposes.

15.2.3 CLEAR (CLR)

The Clear mode is generally comparable to the analog Initial Condition (IC) mode. The Clear mode is used to initialize the logic elements. When clear is selected, all flip-flops, the counter/timer, and the clock downcounter return to their zero state. The logic then remains cleared, and in the Stop mode, until manually stepped, or RUN is selected.

15.3 LOGIC MODE CONTROL

All logic modes are selected manually at the logic control panel using the LOGIC MODE pushbuttons (Figure 15.1). These three pushbuttons (CLR, STP and RUN) correspond to the logic modes described in Paragraph 15.2. The clock downcounter outputs (PULSES) in the logic field are only active in the Run mode. When the STP pushbutton is initially depressed, the logic elements hold the state attained when Stop was selected. Subsequent operation of this pushbutton generates a single clock pulse each time it is depressed and will advance the logic functions a single step at a time. Manual stepping of the logic program is most useful during program checkout and maintenance operations.

The following procedure outlines the typical sequence in which the LOGIC MODE pushbuttons are operated.

1. Depress CLR to place all clocked logic elements in their zero state (reset).
2. Preset all flip-flops (Chapter 27) and the counter/timer (Chapter 29) as required by the overall program.
3. Repeatedly depress STP (while observing the logic indicators) to verify the logic sequence of events.

NOTE

While performing Step 3, use the manual control of flip-flops and the counter/timer to invert their logic outputs on an as required basis. See Chapter 27 and 29.

4. Depress CLR to reset all clocked logic elements.
5. Preset all flip-flops as in Step 2.
6. When ready to start the program run, depress RUN.
7. If at any point in time it is desired to arbitrarily examine (Readout) any logic event depress STP. To restart the program, depress RUN.
8. At the end of a program run, or when the desired results have been attained, depress CLR or STP.

15.4 THE CLOCK DOWNCOUNTER (PULSES)

The clock downcounter, is a key part of the logic clock system and adds to programming flexibility by providing clock output terminations with frequencies of 10^5, 10^4, 10^3 and 10^2 Hz. Each output is a positive pulse train developed by a countdown from the internal master clock.

As shown in Figure 15.1, there are five terminations (designated PULSES) associated with the clock downcounter. The C termination is a control input for clearing (resetting) the downcounter. This termination is normally low, and when patched high all PULSES outputs go low and remain low until C goes low. The four output terminations designated PULSES are primarily used to control the counter/timer (Chapter 29) and providing the C input is not patched high, are always present when the logic Run mode is selected. When the logic Clear mode is selected, the timer resets in the same manner as when controlled by the PULSES-C input. When the logic STP mode is initially selected, all outputs remain in the state attained at the clock just prior to entering STP. Subsequent operation of the STP pushbutton (without entering any other logic mode) will generate one count every time the button is depressed.

Table 15.1 shows the frequency and period (1/F) for each downcounter output. Since most logic components in the MiniAC operate on the trailing edge of a clock, the period of each output is defined as being from the trailing edge of one pulse in the train to the trailing edge of the next pulse.

These clock outputs can be used in any application requiring synchronization with the master clock, but at a lower frequency. This is accomplished by patching the appropriate output termination (10^5 10^4 etc.) to the input of the element to be controlled. Typically this will be the CI (count in) input of the counter/timer. Refer to Chapter 29 for patching information.

Table 15.1. Clock Downcounter Output Frequencies and Time Periods

Pulses Output	Period (Microseconds)
10^2	10,000
10^3	1,000
10^4	100
10^5	10

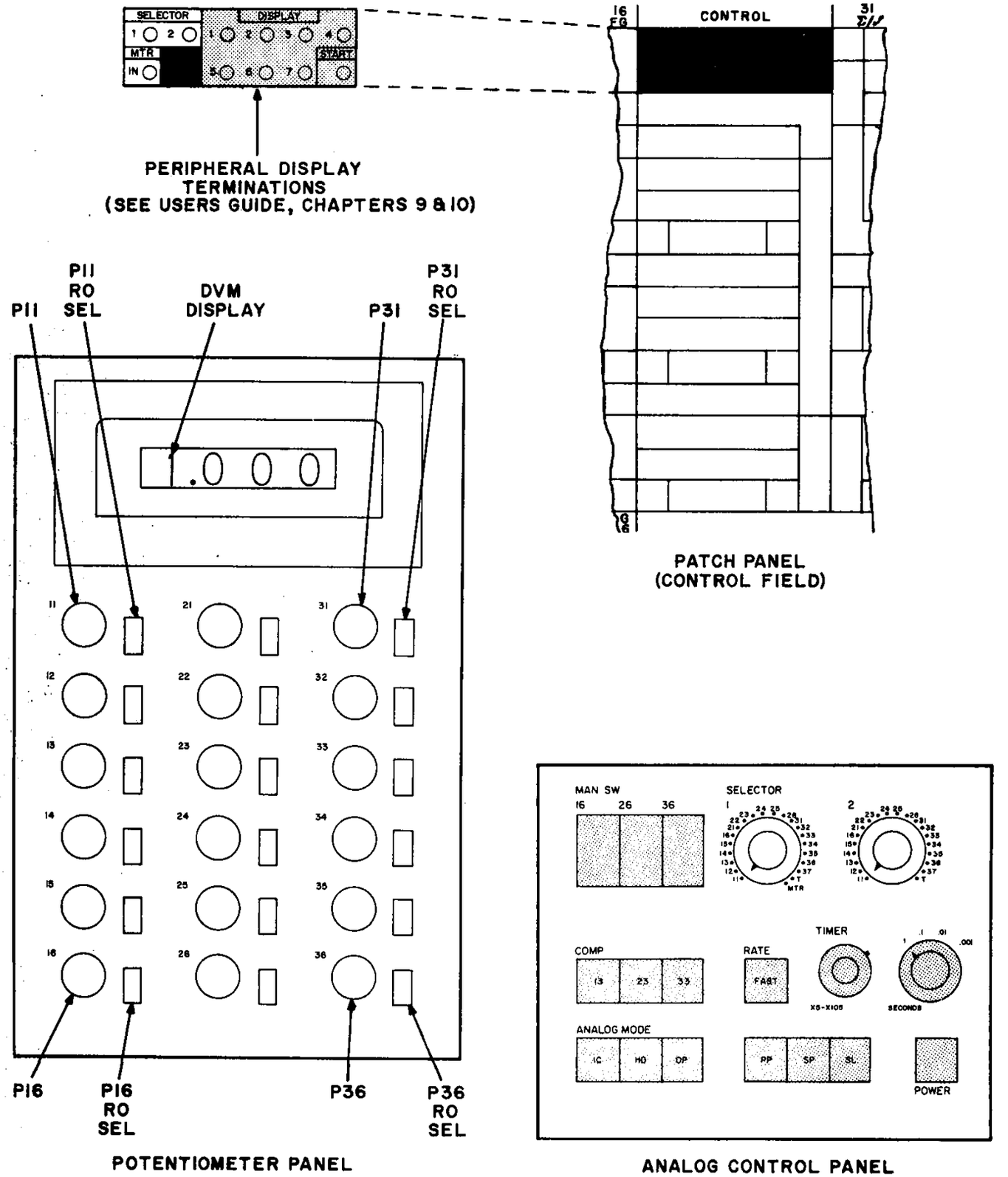


Figure 16.1. Readout Controls and Patching Areas

16.1 INSTRUCTION

The MiniAC Signal Selector System provides a convenient means of monitoring the output (analog value) of nearly all analog components in the computer. Additionally, the selector system permits readout of external devices (and/or the use of peripheral devices) to monitor the value of any component selected for readout.

Basically, the selector system (Figure 16.1) consists of two selector switches (SELECTOR 1 and 2) the digital voltmeter (DVM), a group of terminations on the patch panel (designated SELECTOR), and the individual potentiometer readout pushbutton switches. The DVM provides a numeric display representing the sign and value of the component selected for readout. SELECTOR 1 is the main readout control and connects the output of the selected device directly to the DVM and the SELECTOR 1 termination. Therefore, the output of any device addressed by SELECTOR 1 is displayed on the DVM and is also present (as a patchable signal) at the SELECTOR 1 termination. The switch designated SELECTOR 2 is primarily used in conjunction with external and peripheral devices. This switch merely connects the output of the addressed device to the SELECTOR 2 termination. Figure 16.2 is a simplified diagram of the selector system and is provided to illustrate the overall operation and flexibility of the readout capabilities. Note that the individual potentiometer pushbuttons permit readout of a potentiometer coefficient or value. Depressing one of these switches disconnects SELECTOR 1 from the DVM and the DVM displays the pot wiper voltage. Refer to Figures 16.1 and 16.2 as necessary throughout the remainder of this chapter.

16.2 READOUT OF ADDRESSABLE COMPONENTS

With the exception of; D/A and MAN switches, limiters, and comparators, all components terminated in the three analog fields can be monitored on the DVM without extraneous patching. In addition, the T output (a sawtooth) of the interval timer (Chapter 29) can also be selected. Numbered designations (11, 12, etc.) on each switch (SELECTOR 1 or 2) correspond to the component numbering system (module address) described in Chapter 2, and identifies the component selected for readout on the DVM. To address (select) any component in the analog fields for readout on the DVM, simply set SELECTOR 1 at the corresponding module number. Table 16.1 summarizes the type of component selected for each switch position of SELECTOR 1 and describes the basic switch functions. To select any addressable component for external or peripheral monitoring functions patched to the SELECTOR 2 termination, simply set the SELECTOR 2 switch at the desired module number. Note that SELECTOR 2 addressing is identical to SELECTOR 1 except it does not have an MTR position.

Table 16.1. Device Addressing (SELECTOR-1)

Switch Position	Addressed Device	Function
11, 12 21, 22 31, 32	$\Sigma/\int$	Selects addressed component for value readout on DVM. Also connects selected device to SELECTOR-1 termination.
13 23 33	Σ	
14 24 34	Σ/TS	

Table 16.1. Device Addressing (SELECTOR-1) (Cont)

Switch Position	Addressed Device	Function
15 25 35	MULT/DIV	Selects addressed component for value readout on DVM. Also connects selected device to SELECTOR-1 termination.
16 26	FG	
36 37	VFG	
T	Interval Timer	Selects interval timer SAWTOOTH OUTPUT (T) for readout on DVM. T is also connected to SELECTOR-1 Termination.
MTR	MTR IN Termination	Connects MTR IN termination to DVM. The value of any signal patched to MTR IN is displayed.

16.3 THE DVM DISPLAY

The value of a selected analog component is read out on the DVM display indicators. These indicators (Figure 16.1) are centrally located at the top of the potentiometer panel. The DVM display is active whenever computer power is applied and displays analog values in machine units. One machine unit is equal to computer reference or ± 10 volts. The left most indicator displays a minus sign whenever the signal being measured is a negative value. The next indicator to the right displays a numeral one if unity or greater is measured. The remaining indicators provide numeral readout of all values less than unity. Note that the display in this case (Figure 16.1) is .962 (the decimal is stationary). This is equivalent to 9.62 volts at the output of the selected analog device.

16.4 THE SELECTOR TERMINATIONS

16.4.1 SELECTOR 1 AND 2

The terminations in the patch panel control field designated SELECTOR (Figures 16.1 and 16.2) are an integral part of the signal selector system. These terminations may be used for monitoring or any other program purposes. The numeric designations 1 and 2 correspond to the SELECTOR 1 and 2 switches. Therefore, the value of any component selected for readout by switches 1 and 2 appears at the corresponding numbered termination. The signals at these terminations can be patched to the DISPLAY terminations for use by a peripheral device, to an output trunk, or can even be patched into the program if required. Using terminations 1 and 2 in combination permits simultaneous monitoring of two components or channels. The black (ground) termination may be used in conjunction with the 1 and 2 terminations when monitoring a component output on an external device such as a DVM or oscilloscope.

16.4.2 SELECTOR-MTR IN

When the SELECTOR 1 switch is at the MTR position, the output of an external analog device patched to MTR-IN (usually via an input trunk) is selected for readout on the DVM. When using this termination, ensure that the external analog signal source is compatible with the MiniAC (±10V). Note that the MTR-IN terminations can also be used for monitoring power voltages or logic signals during maintenance operations.

ANALOG FIELD	DEVICE					CONTROL FIELD
	Σ//	Σ	Σ/TS	MULT/DIV	FG	TIMER
1	11,12	13	14	15	16	T
2	21,22	23	24	25	26	
3	31,32	33	34	35	36/37	

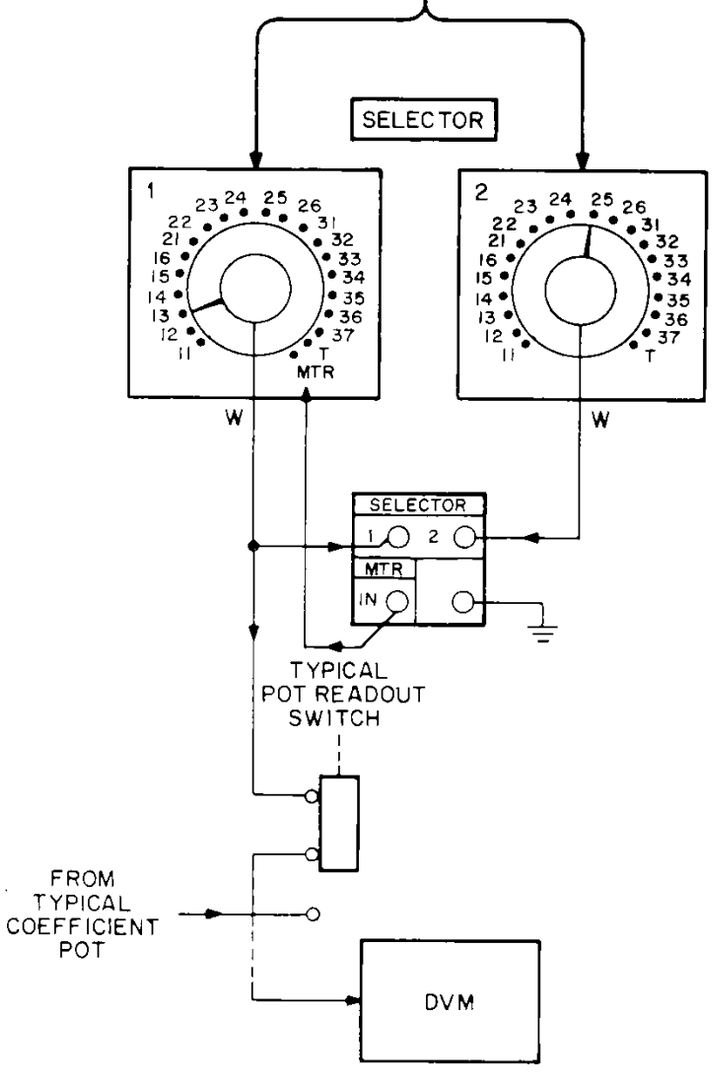
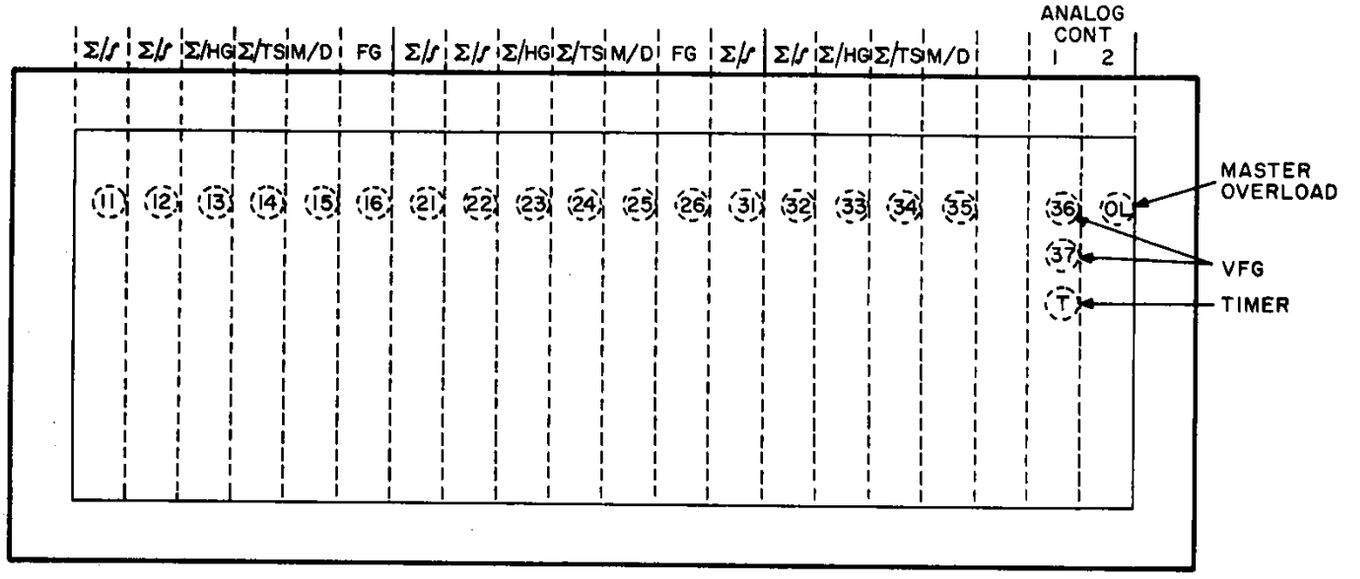
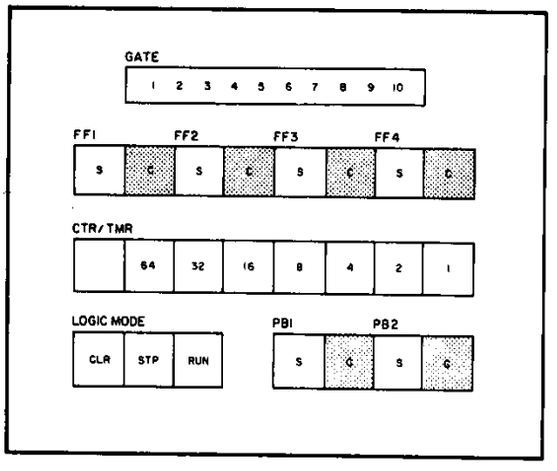


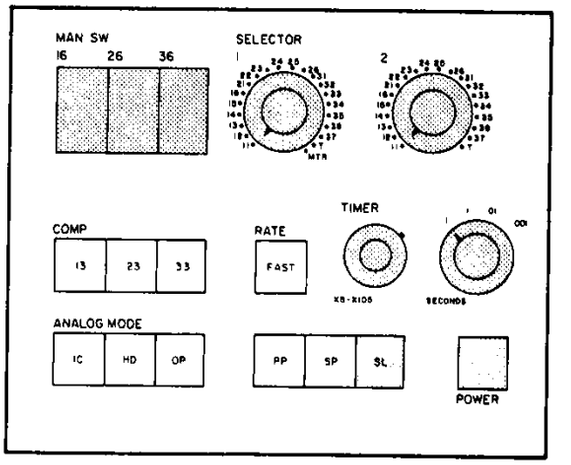
Figure 16.2. Signal Selector System: Simplified Diagram



SET-UP SWITCH/OVERLOAD INDICATOR PANEL



LOGIC CONTROL PANEL



ANALOG CONTROL PANAL

Figure 17.1. Overload and Status Indicator Locations

CHAPTER 17

THE OVERLOAD AND STATUS INDICATORS

17.1 INTRODUCTION

The MiniAC is equipped with an overload indicator system and various status indicators (Figure 17.1). The overload indicators provide a visual alarm to inform the operator of an amplifier overload condition. The status indicators permit the operator to monitor the status of: analog time scale; analog and logic modes; all logic elements (except the MONO); and comparator outputs. The following paragraphs describe the function and use of these indicators.

17.2 THE OVERLOAD INDICATOR SYSTEM

17.2.1 OVERLOAD CRITERIA

An amplifier is regarded as overloaded when its summing junction departs from virtual ground. This usually indicates that the output has been driven beyond its normal operating range and is generally referred to as a voltage overload. A current overload occurs when amplifier exceeds its current output capability. Current overloads are usually caused by illegal patching such as, connecting the output of two amplifiers together. Most amplifiers in the MiniAC are capable of producing outputs up to about 1.1 or 1.15 machine units (11.1 or 11.5 volts) before overloading. A multiplier or a function generator overloads when any input exceeds 1.0 machine unit (10 volts).

Overloads are generally caused by incorrect scaling of one or more problem variables, by improper time scale selection, or by patching errors. However, a component malfunction can also cause overloads.

17.2.2 THE OVERLOAD INDICATORS

As shown in Figure 17.1, the overload indicators are mounted on the cards behind the swing-down setup switch and overload indicator panel. In all, there are 21 indicators (20 component and 1 master). When an analog component overloads, the corresponding lamp glows and illuminates that portion of the etched panel directly in front of the lamp. Additionally, the master overload indicator (designated OL) also lights. This indicator is used in conjunction with the OLS and ORH function described in Chapter 14. The overloaded component (amplifier, multiplier, etc.) is identified by the etched number (11-37) that corresponds to the patch panel module address. Note that with the exception of the VFG's (36 and 37), the timer (T) and the master overload indicator (OL), the indicators are mounted directly on the component card in the card location corresponding to the module numbering system.

17.2.3 LOGIC LEVEL AND CONTROL

In addition to the Visual overload alarms, there is an overload output (OL) in the patch panel control area. This termination goes high whenever any component overloads and can be used in conjunction with programmable features that force the computer into the Hold mode (override hold) and/or store a transient overload (OLS). Selecting the Hold mode and storing an overload freezes the problem to permit locating a source of trouble. Details of these features are given in Chapter 14.

17.3 ANALOG STATUS INDICATORS

17.3.1 INTRODUCTION

All analog status indicators are located on the analog control panel as illustrated in Figure 17.1. Note that all of these indicators are associated with pushbutton switches. This description is oriented only toward the monitoring function. For a description of the pushbutton functions, refer to Chapter 14 and Table 2.5. The indicators on this panel are used to monitor the state of all comparators, the computer rate (time scale), and computer modes.

17.3.2 COMPARATOR INDICATORS

The COMP group consists of three indicators designated 13, 23, and 33. These designations identify the comparator by the module address described in Chapter 2. When illuminated, the COMP status indicators signify that the corresponding comparator output is at a logic ONE (the algebraic sum of the analog inputs is a positive value).

17.3.3 TIME SCALE INDICATOR

The RATE-FAST indicator is associated with the time scale control system. When extinguished, the normal computer time scale is selected. When illuminated, this indicator signifies that the scale selection is 500 times faster than normal.

17.3.4 ANALOG MODE INDICATORS

The ANALOG MODE group consists of four indicators that correspond to the analog modes and two indicators that signify the source of mode control. The indicators designated IC, HD, OP and SP correspond to the computer modes described in Chapter 14. When any one of the analog modes is selected, the corresponding indicator illuminates. The PP (Patch Panel) lamp illuminates when the patch panel is selected to provide programmed computer mode control. If PP is extinguished and SL is not depressed, all analog modes and computer time scale (RATE) are under local pushbutton control.

17.4 LOGIC STATUS INDICATORS

As illustrated in Figure 17.1, the logic control panel contains all indicators associated with the logic elements and logic mode control. Note that all indicators (except those designated AND GATES) are associated with pushbutton switches. This description is oriented only toward the monitoring function of the indicator. For a description of the pushbutton functions, refer to Table 2.6 and the individual chapters describing the corresponding logic elements.

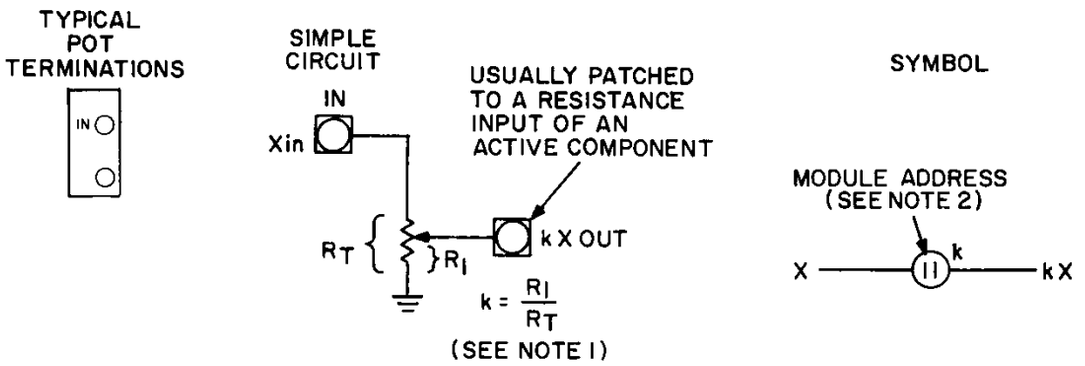
The GATE group illuminates when the corresponding AND Gate output is high. The numeric designation 1-10 corresponds to the patch panel gate numbering system. Up to ten AND Gates are terminated in the patch panel logic field and are designated GATE 1 to GATE 10. Note that unpatched gates always have a high output. Therefore, the indicator for any unpatched gate will always be illuminated. Refer to Chapter 26 for AND Gate operation.

The general purpose flip-flop indicator group is identified by the panel markings FF1 to FF4. Whenever a flip-flop is set (either manually or by program), the corresponding S set) indicator illuminates. When the flip-flop is reset the lamp extinguishes.

The CTR/TIMER group consists of eight pushbutton indicators. Seven of these (the numbered ones) illuminate in ascending order (from right to left) to display the BCD value to which the counter has advanced. For example, when a count of 21 is reached, the 16, 4 and 1 indicators are lit. As the counter cycles through its count, these lamps extinguish (in turn) and again illuminate when the count is completed. The counter and display will continue to cycle until the counter is cleared either manually, or by the patched program. The blank (undesignated) pushbutton lamp lights when the true output of the CTR/TMR goes high and remains lit until the preselected count is reached or the counter is cleared. For more information concerning the counter/timer, refer to Chapter 29.

The LOGIC MODE group consists of two indicators designated STP and RUN. When the Stop or Run logic mode is selected, the corresponding indicator illuminates. Note that the CLR pushbutton is not an indicator. When CLR is depressed, the logic elements are reset and the computer automatically reverts to the logic Stop mode. Refer to Chapter 15 for more information.

The pushbutton flip-flop (PBFF) indicators are identified by the PB1 and PB2 panel markings. When a PBFF is manually set, the S indicator lights. The lamp extinguishes when the flip-flop is reset.



NOTES:

1. THE CONSTANT (k) FOR A POT WITH ITS WIPER ARM PATCHED TO A LOAD (R_L) IS:

$$k = \frac{\frac{R_I}{R_T}}{1 + \left(\frac{R_I}{R_L}\right)\left(1 - \frac{R_I}{R_T}\right)}$$

2. FOR EASE OF IDENTIFICATION ON A PROGRAM DIAGRAM THE POT ADDRESS IS PLACED WITHIN THE SYMBOL. THE VALUE OF k (COEFFICIENT) IS PLACED IN CLOSE PROXIMITY TO THE SYMBOL.

Figure 18.1. Potentiometer: Circuit and Symbol

COEFFICIENT POTENTIOMETERS

18.1 IDENTIFICATION AND LOCATION

A fully expanded MiniAC contains 18 hand set coefficient, potentiometers (pots). These pots are physically located on potentiometer panel (Figure 2.4) directly behind the dial for each pot. The grounded potentiometers are arranged in vertical groups of six, with each group corresponding to one of the patch panel analog fields. The manual control for each pot is identified by the numerical panel marking (11, 12, etc.) that corresponds to the module numbering system described in Chapter 2.

For ease of identification and patching, the pot terminations are color coded yellow and one pot is terminated in each patching module.

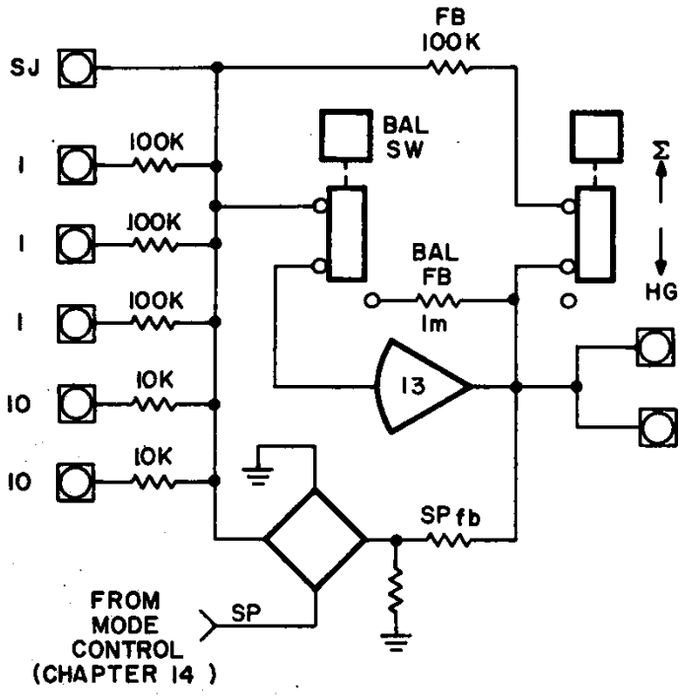
18.2 THE POTENTIOMETER

Potentiometers or pots (sometimes called attenuators), provide a means of multiplying a dc voltage by a constant (k) that is less than unity. The pot illustrated in Figure 18.1 is simply a fixed resistor with a movable arm. The grounded potentiometer is usually used in conjunction with computer reference (±10V) to obtain a fixed voltage less than reference, or to multiply a problem variable by a constant less than unity. The potentiometer input is applied at the high end (designated IN), and the resultant output is obtained at the wiper arm.

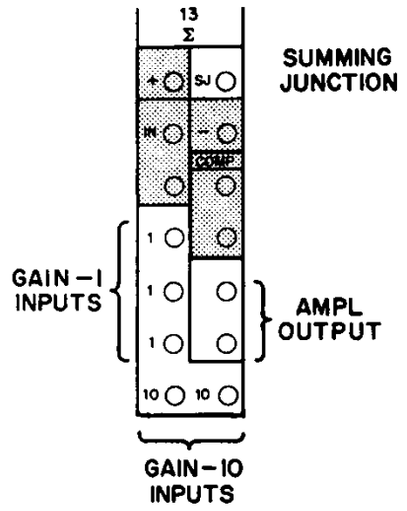
18.3 POTENTIOMETER SETUP AND MONITORING

The output of each pot (wiper arm voltage) may be monitored on the DVM in either of two modes: one mode for coefficient setup; the other for monitoring wiper levels with actual problem inputs applied. When setting coefficients, +10 volts (reference) is applied to the pot high side and the pot coefficient (setting) is read by observing the DVM. This is accomplished by selecting the SP mode (Chapter 6) and simply depressing and holding the pot RO (readout) pushbutton while adjusting the pot for the desired coefficient as displayed on the DVM. Note that pots normally operate into a resistive load and are subject to the effects of loading. Therefore, never attempt to set a pot coefficient without first patching the pot into the problem and selecting the SP mode. Failure to observe this precaution will result in an erroneous setting.

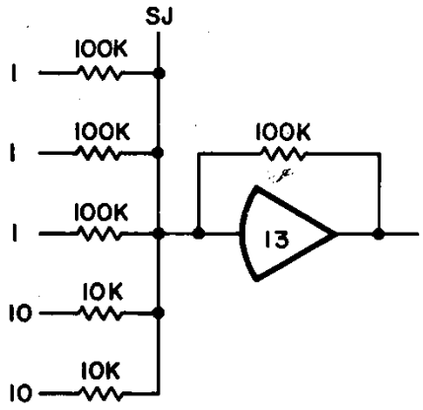
In the "Operational" Mode, the wiper voltage (the result of the pot coefficient setting multiplied by the problem variable or constant), is displayed on the DVM. This mode is obtained while the computer is in any operating mode (IC, OP or H) by simply depressing the pot RO pushbutton and observing the displayed value.



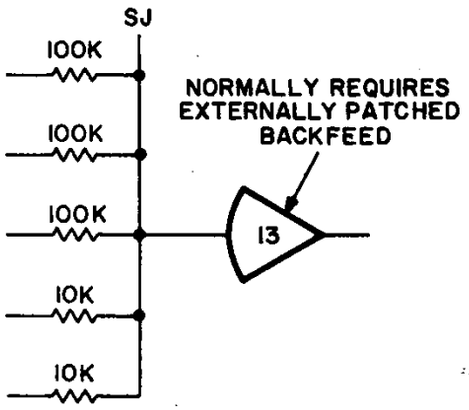
a. SIMPLIFIED CIRCUIT



b. PATCHING TERMINATIONS



Σ FUNCTION (Σ/HG SWITCH RELEASED)



HIGH GAIN FUNCTION
(Σ/HG SWITCH SET)

c. EQUIVALENT CIRCUITS

Figure 19.1. Summer/High Gain: Simplified and Equivalent Circuits

FIVE-INPUT SUMMER/HIGH-GAIN AMPLIFIER

19.1 LOCATION AND IDENTIFICATION

The fully expanded MiniAC contains three summer/high gain amplifiers. These units are housed in the card file directly behind the setup switched/overload indicator panel (Figure 2.7) and are terminated at the patch panel (Figure 2.2). Each unit is identified by the numeric designation (13, 23, 33), visible on the card file when the front panel is lowered. These numbers correspond to the patch panel module address described in Chapter 2.

19.2 THE AMPLIFIER

Figure 19.1 is a simplified circuit of the summer/high-gain amplifier (Σ/HG). This unit is provided with a 100k ohm feedback resistor, three 100k ohm (gain-1) inputs, two 10k ohm (gain-10) inputs and a summing junction input (SJ). The SJ termination is reserved for use with limiters or D/A switches. Refer to Chapters 22 and 28.

In addition, the unit is equipped with a setup switch and a balance switch. When the setup switch is released (extended), the unit operates as a summer with the standard 100k feedback resistor. When set (depressed), the 100k feedback resistor is removed from the circuit. This permits an externally patched device to provide amplifier feedback. The balance switch is a momentary pushbutton for use when checking amplifier balance. When the momentary balance switch is depressed all inputs are removed from the amplifier (a typical balancing technique), and the standard feedback resistor is replaced by a larger value (higher gain) balance feedback resistor. Amplifier balancing procedures are given in Chapter 3.

19.3 USED AS A SUMMER

When operated as a summer, the amplifier has a standard 100k feedback resistor and the amplifier output is minus the sum of all patched inputs. An unpatched input has no effect on the problem solution. To use this unit as a summer, simply release the Σ/HG switch and patch the unit as required by the overall program. Figure 19.2 illustrates some of the more common summer patching configurations. The equivalent circuit and program symbol is also shown for each configuration. Part a of Figure 19.2 illustrates patching the unit as a standard five input summer. Parts b through e show typical methods of obtaining nonstandard gains. Notice that none of these configurations show patching to the amplifier SJ termination. This termination is only used in conjunction with limiter functions and in cases when a switched gain-10 input is desired.

19.4 USED AS A HIGH-GAIN AMPLIFIER

When used as a high-gain amplifier, the standard 100k feedback resistor is removed from the feedback path. During high-gain operation, feedback is usually obtained from a non-linear impedance device such as a function generator. In such a case, gain is equal to the ratio of input impedance to feedback impedance Z_f/Z_{fin}. When used in high-gain applications $E_o = -E_{in} \cdot z_f/Z_{in}$. To use this unit as a high-gain amplifier, simply set (press) the Σ/HG switch. Patching information for the high-gain circuit configuration is shown in Figure 19.3. In addition to its use as a high-gain amplifier, the HG function can be used to obtain non-standard gains by patching gain-10 resistors into the feedback path. These techniques are shown in parts b and c of Figure 19.3.

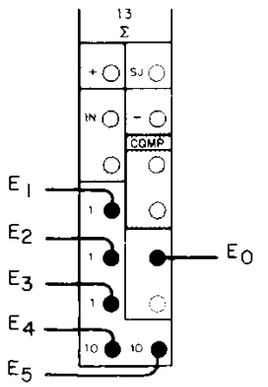
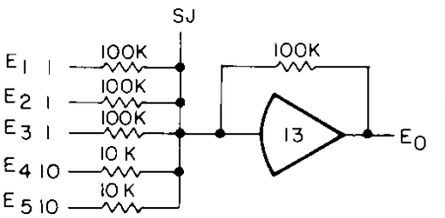
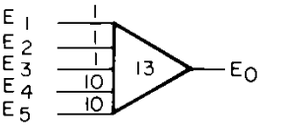
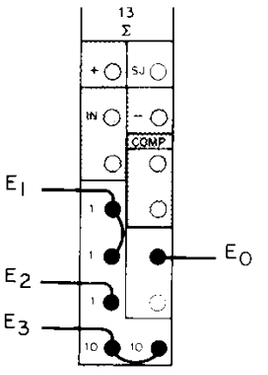
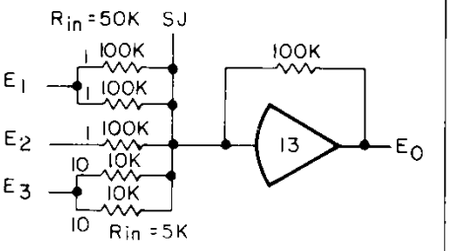
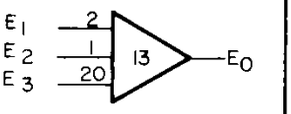
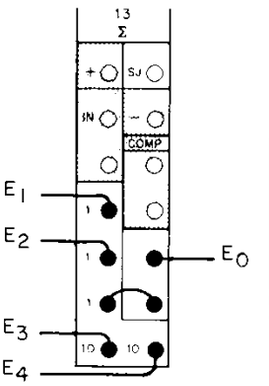
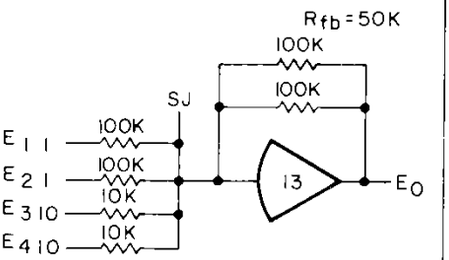
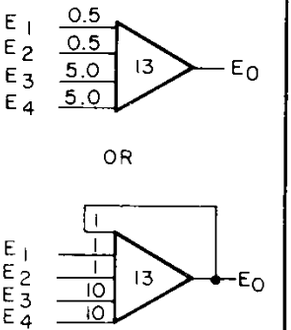
CONFIGURATION	PATCHING	EQUIVALENT CIRCUIT	PROGRAM SYMBOL
a. STANDARD FIVE INPUT SUMMER (GAINS OF 1 AND 10)		 $E_O = -(E_1 + E_2 + E_3 + 10E_4 + 10E_5)$	
b. OBTAINING GAINS OF 1, 2 AND 20		 $E_O = -(2E_1 + E_2 + 20E_3)$	
c. OBTAINING GAINS OF 0.5 AND 5.0		 $E_O = -(0.5E_1 + 0.5E_2 + 5E_3 + 5E_4)$	

Figure 19.2. Summer Patching: Σ/HG Switch Released (Out)

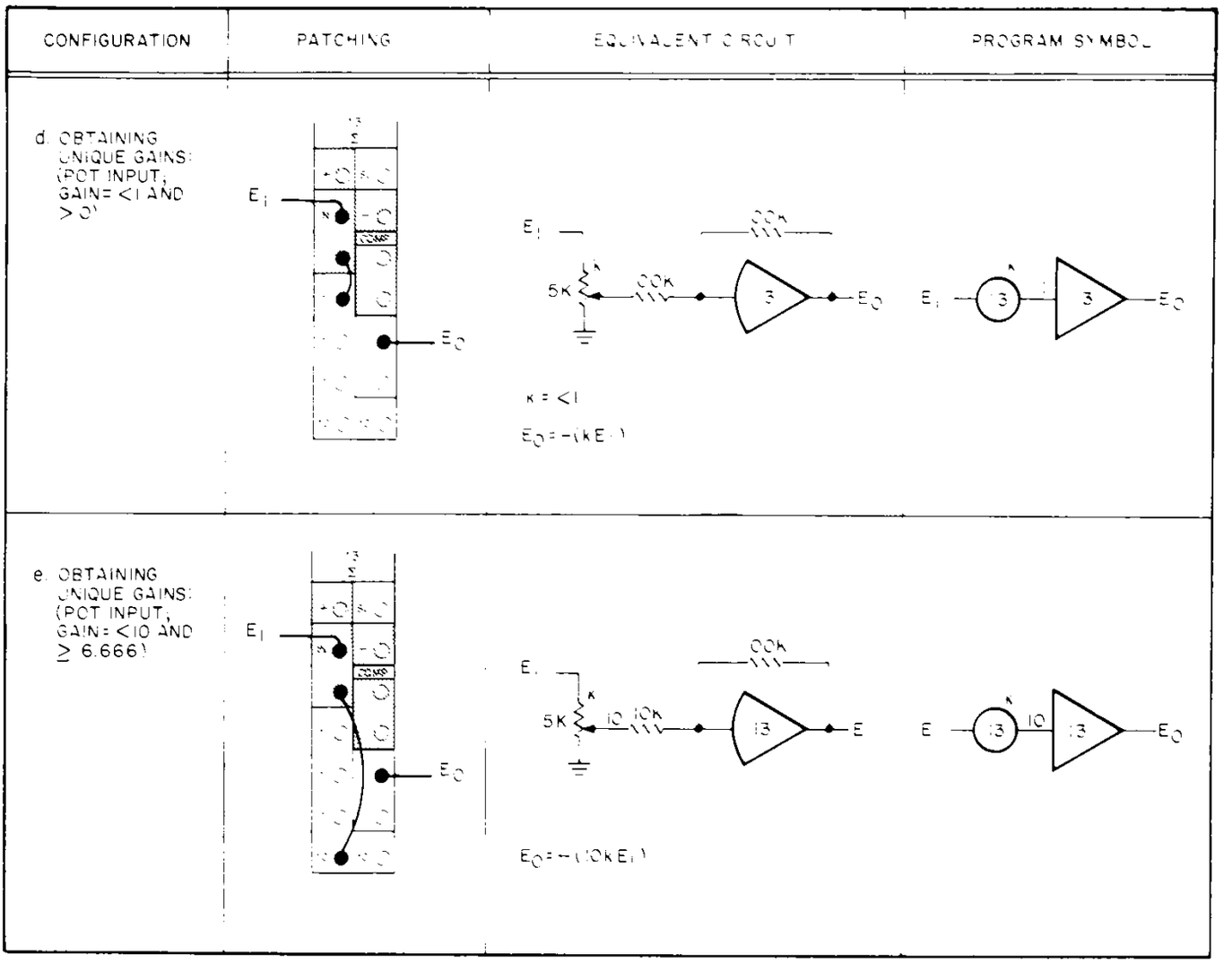


Figure 19.2. Summer Patching: Σ HG Switch Released (Out) (Cont)

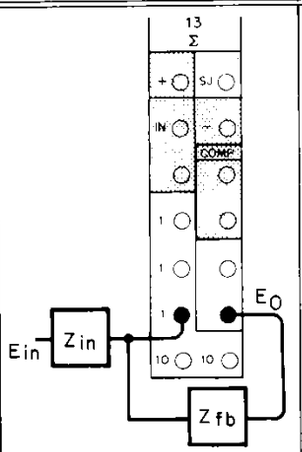
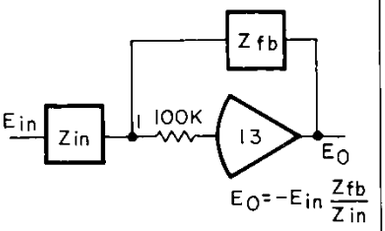
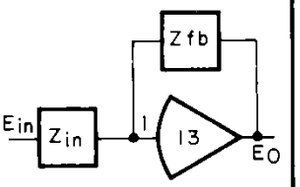
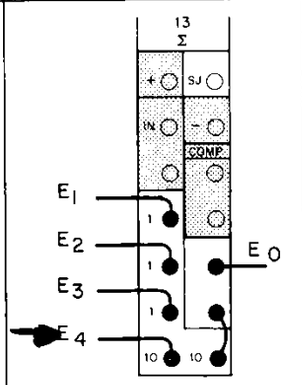
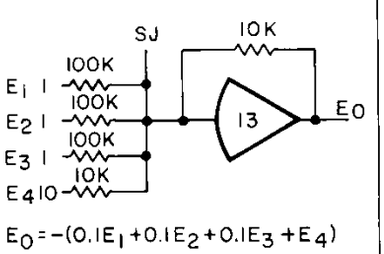
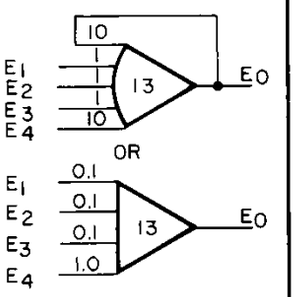
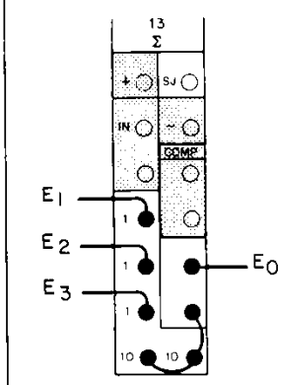
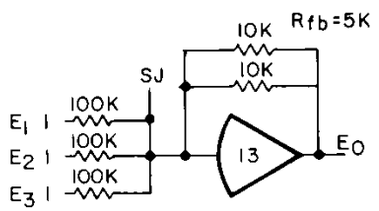
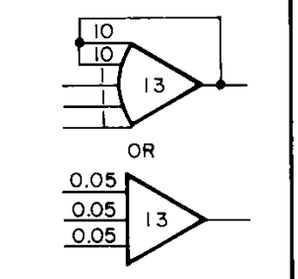
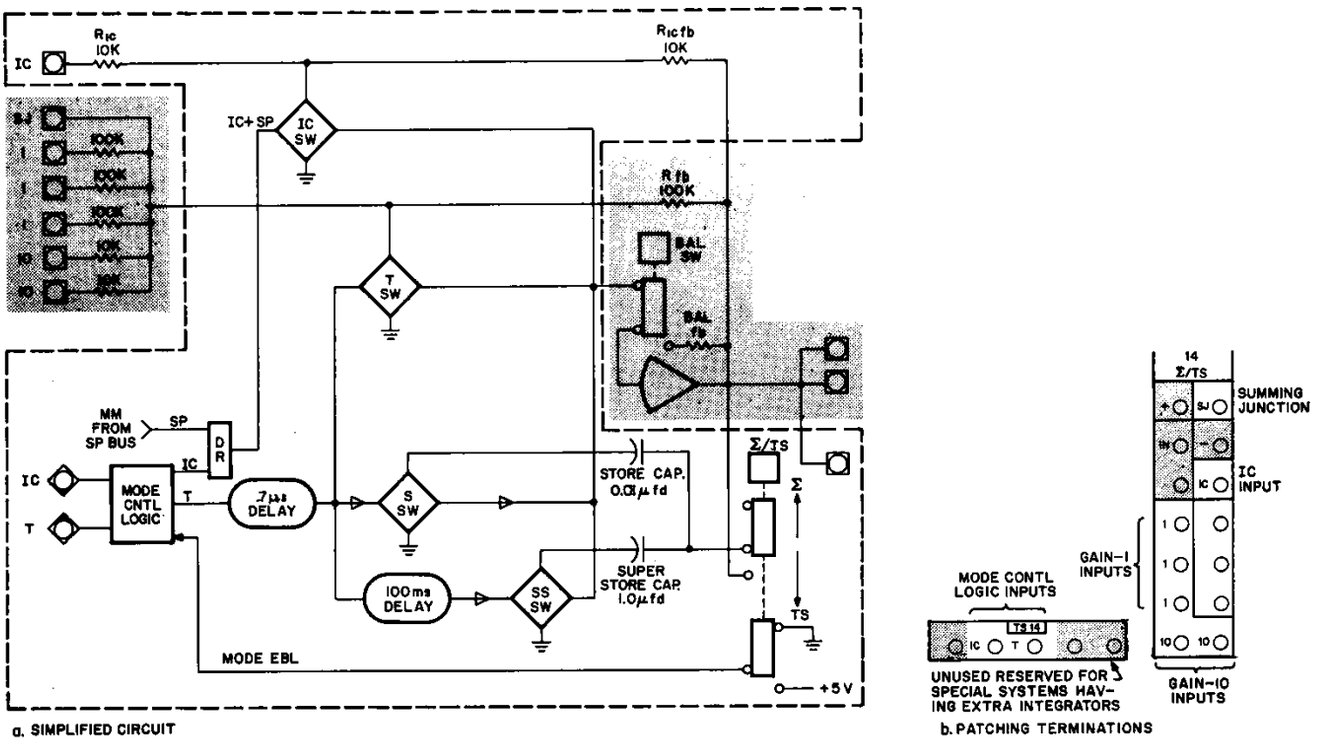
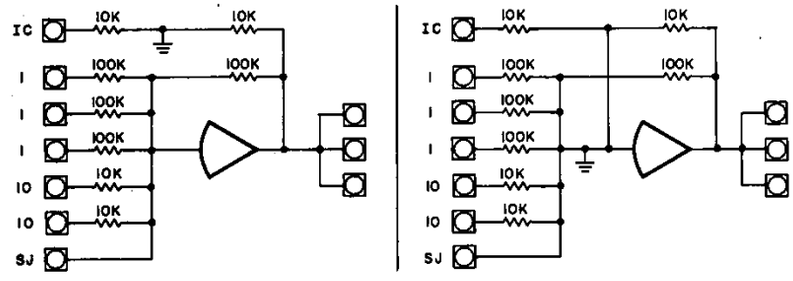
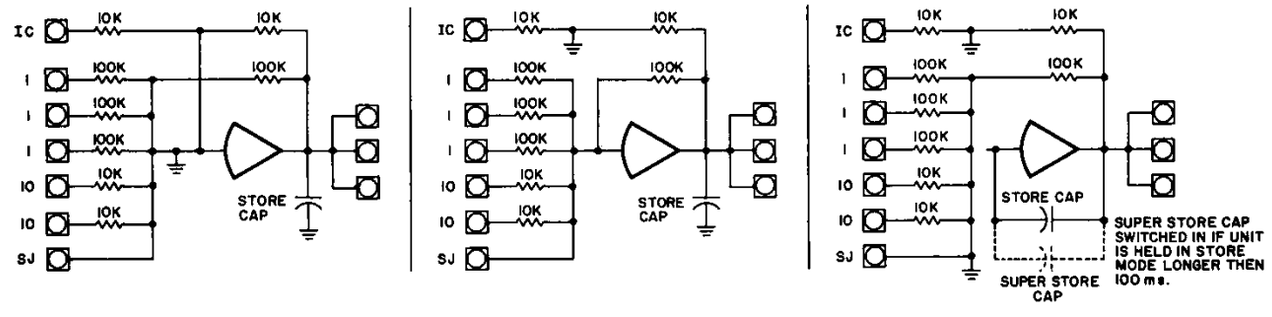
CONFIGURATION	PATCHING	EQUIVALENT CIRCUIT	PROGRAM SYMBOL
a. USED AS A HIGH GAIN AMPLIFIER		 NOTE: Z fb AND Zin CAN BE FROM ANY IMPEDANCE SOURCE SUCH AS A FUNCTION GENERATOR, OR A POT. BOTH FEEDBACK AND E in MUST ALWAYS BE PATCHED THROUGH AN INPUT RESISTOR.	
b. OBTAINING GAINS OF 0.1 AND 1			
c. OBTAINING GAINS OF 0.05			

Figure 19.3. High-Gain Patching: Σ/HG Switch Set (In)





c. SUMMER FUNCTION: EQUIVALENT CIRCUITS (Σ/TS SWITCH RELEACED)



d. TRACK STORE FUNCTION EQUIVALENT CIRCUITS (Σ/TS SWITCH SET)

Figure 20.1. Summer/Track-Store: Simplified and Equivalent Circuits

20.1 LOCATION AND IDENTIFICATION

The fully expanded MiniAC contains three summer/track-store amplifiers. These units are housed in the card file directly behind the setup switch/overload indicator panel (Figure 2.7). Each track-store amplifier (and its location) is identified by the numeric designation (14, 24, 34) visible at the top of the card file when the front panel is lowered. These numbers correspond to the patch panel module address described in Chapter 2. One summer/track-store amplifier (Σ/TS) is terminated in each field of the patch panel (Figure 2.2) These terminations are identified by the module address (14, 24, 34) and by the patch panel Σ/TS symbol. Each Σ/TS is also terminated in the patch panel control field. These terminations provide individual logic control of the Σ/TS modes (Chapter 14).

20.2 THE T-S AMPLIFIER

20.2.1 INTRODUCTION

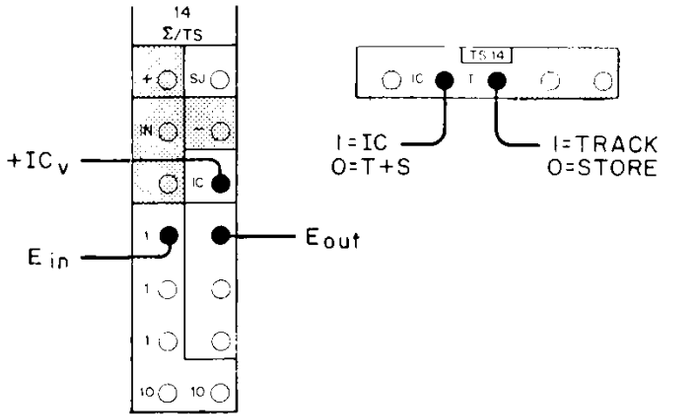
Basically, the track-store amplifier (T-S) is a standard summer with additional features that permit the device to; start with an output at an initial condition (IC) value, provide an input that follows the negative sum of the inputs (Track), and hold the output value obtained at any given point in time (Store).

Figure 20.1 is a simplified circuit of the T-S amplifier and also illustrates the equivalent circuit for each method and mode of operation. This unit may be used as a standard five-input summer, or as a track-store device. The T-S amplifier is very similar to the summing amplifier described in Chapter 19. Note that the summing portion of the amplifier circuit (in the shaded area) is nearly identical to the summer previously described. The major difference between the two amplifiers is that the T-S amplifier includes mode control logic, electronic switches (for mode control), storage capacitors, and also has provisions for an IC input. Another significant difference is that the T-S amplifier does not have the "high-gain" capability of the standard summer. The unit is equipped with a setup switch (Σ/TS) and a balance switch (BAL).

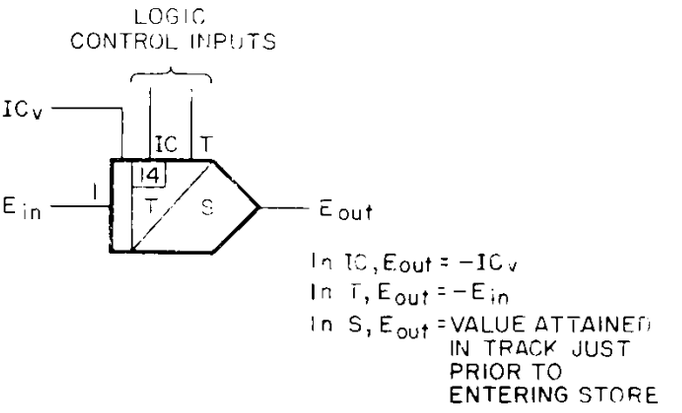
When the setup switch is released (extended), the unit operates as a summer (Σ) with a standard 100k feedback resistor (Figure 20.2). When set (depressed), the storage capacitors are connected in the feedback path whenever the Store mode is selected. The balance switch (BAL) is a momentary pushbutton that removes all inputs from the amplifier during amplifier balancing procedures. See Chapter 3.

20.2.2 MODE CONTROL

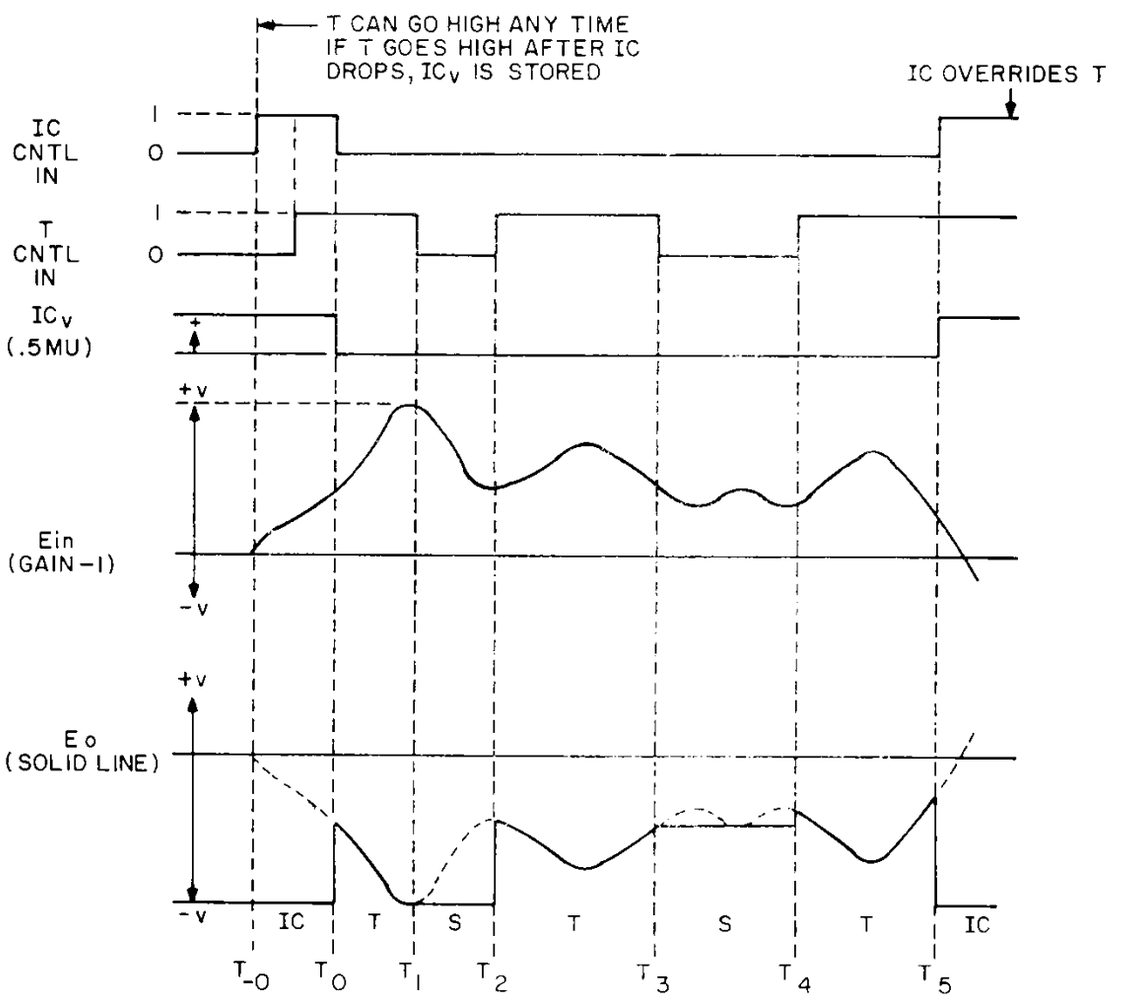
As previously stated, the T-S amplifier has three operating modes (Track-T, Store-S, and Initial Condition-IC). The electronic switches designated IC, T, and S respond to logic inputs at the patch panel to select the corresponding mode of the T-S unit. When the IC switch is conducting the 10k initial condition resistors (R_{ic} and R_{icfb}) are switched into the amplifier circuit and the amplifier output is the inverse of the analog value patched to the IC input termination. When the T switch is conducting, the amplifier follows (tracks) the patched analog inputs (acts as a summer). When the store switch is conducting, the 0.01 microfarad store capacitor is connected to the amplifier feedback path and all other inputs are disconnected from the amplifier summing junction. Therefore, when store is selected, the amplifier output is equal to minus the sum of the inputs (and the store cap holds that value). A fourth switch (SS) is provided that places the 1.0 microfarad super store cap in the amplifier feedback path if the store mode is selected for 100 milliseconds or more. The larger value capacitance provides better drift characteristics and ensures a more accurate problem solution when a value must be stored for a relatively long period. The smaller value capacitor permits rapid tracking of the analog inputs.



a. PATCHING



b. SYMBOL



c. VOLTAGE WAVEFORM

Figure 20.2. Typical Track-Store Operation

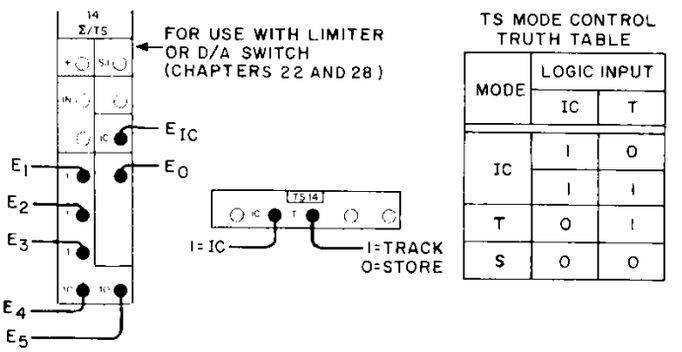
The mode control logic decodes the inputs patched at the IC and T terminations. When the Σ/TS switch is released (Σ function) this logic is inhibited and the unit is a summer and cannot be controlled by patching. Setting the Σ/TS switch (T-S function) enables the control logic and logic signals patched to IC and T to take control. The IC termination is normally low and must be patched high to enter IC. When high, IC overrides the T input. The T termination controls the track and Store Modes. T is normally high and when unpatched maintains the unit in the track mode. When T is patched low, the unit enters the store mode. Note that the store mode signal to the S and SS switches is the complement of the track mode signal. When switching from track to store, actual switching is delayed about 0.7 microsecond. This delay is generated to permit T-S units to be cascaded or connected in a bucket brigade configuration.

20.3 USED AS A SUMMER

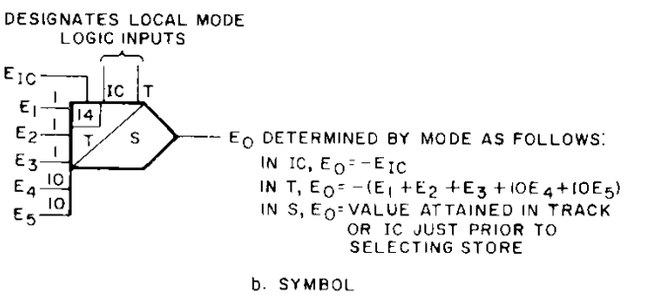
When the Σ/TS switch is released, the T-S amplifier can be used as a multiple input summer. When used in the summer configuration, the track switch (T) is conducting and all patching and operation (with the exception of high-gain use) is identical to the summer amplifier described in Chapter 19.

20.4 USED AS A TRACK–STORE UNIT

The behavior of a typical track-store operation and the related patching is illustrated in Figure 20.2. For clarity, only a single gain-one input is used. In actual practice, multiple inputs can be summed. Figure 20.3 illustrates the basic patching requirements for general use of the T-S Unit. Remember, in all track-store operations, the Σ/TS switch *must* be set (depressed). Some applications may require that two or more T-S units be cascaded in "bucket brigade" fashion. This is made possible by the built-in mode switching delay previously described. The application and use of cascaded T-S units is described in the Users Guide portion of this handbook, specifically in Chapters 11 and 12. Additional mode control information is given in Chapter 14.



a. PATCHING



b. SYMBOL

Figure 20.3. Basic Track-Store Patching

Figure 21.1. Summer/Integrator: Simplified and Equivalent Circuits

21.1 LOCATION AND IDENTIFICATION

The fully expanded MiniAC contains six summer/integrator amplifiers. These units are housed in the card file directly behind the setup switch/overload indicator panel (Figure 2.7). Each unit is identified by the numeric designations (11, 12, 21, 22, 31, 32), visible at the top of the card file when the panel is lowered. These numerals correspond to the patch panel module address described in Chapter 2. Two summer/integrators are terminated in each analog field of the patch panel (Figure 2.2). These terminations are identified by the module address (11, 12, etc.) and by the patch panel $\Sigma/\int$ symbol. Each integrator is also terminated in the patch panel control field. These terminations (designated $\int$ 11, $\int$ 12, etc.), provide individual logic control of integrator mode and time scale. The methods of controlling analog mode and time scale of integrators is described in Chapter 14.

21.2 THE SUMMER/INTEGRATOR

The summer/integrator (Figure 21.1) is provided with resistive and capacitive feedback elements to permit its use as a five input summer, as a three mode integrator, or even as a track-store amplifier. Note that when used as a track-store amplifier, tracking speed is slower than that of a standard T-S unit. Operation of the $\Sigma/\int$ setup switch determines the function of the unit. When the $\Sigma/\int$ switch is released (Σ position), a one megohm resistor provides feedback and the unit can be used as a five-input summer. When the switch is set (depressed), the unit is provided with capacitive feedback and will perform integration with respect to time. Two selectable capacitors are provided with each unit. Capacitor selection (and therefore the rate of integration) is controlled by a relay that responds to logic inputs to select either of two time scales. Time scale selection is performed at the master and local levels of control as described in Chapter 14.

When used as an integrator ($\Sigma/\int$ switch set), the mode control logic is enabled and the two switches OP/H and IC respond to logic inputs to control the integrator mode of operation. When used as a summer ($\Sigma/\int$ switch released), the logic control inputs are overridden to inhibit the IC switch and turn on the OP/H switch. This is necessary to permit the summer to function during computer modes other than operate (OP). When used as a summer, the unit is in the OP mode for all computer mode selections except SP. When Set Pot is selected the unit essentially enters the IC mode and permits setting any patched pots under normal load conditions.

Note that there are three one megohm (gain-1) and two 100k ohm (gain-10) inputs. The gain-10 inputs are switchable and can be controlled by logic patching in the patch panel control field. When unpatched, the control terminations (designated 10SW) are normally high and the gain-10 resistors are connected to the amplifier input. Patching the left or right 10SW control input low disconnects the corresponding gain-10 resistors from the amplifier circuit until the 10SW termination is allowed to go high. This permits switching a gain-ten input into the problem solution at any desired point in time, or when a given analog value has been attained.

21.3 USED AS A SUMMER

When operated as a summer, the amplifier has a one megohm feedback resistor and the amplifier output is minus the sum of all patched inputs. An unpatched input has no affect on the problem solution. To use this unit as a summer, simply release the $\Sigma/\int$ switch and patch the unit as required by the overall program. Figure 21.2 illustrates some of the more common patching configurations. The equivalent circuit and program symbol is also shown for each configuration. Notice that none of these configurations show patching to the OJ or 10SW terminations. The OJ termination is normally used in conjunction with limiter functions (Chapter 22). Refer to Paragraph 21.6 for the methods of using the 10SW terminations.

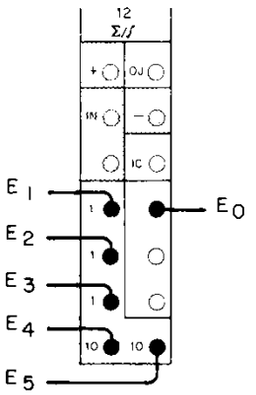
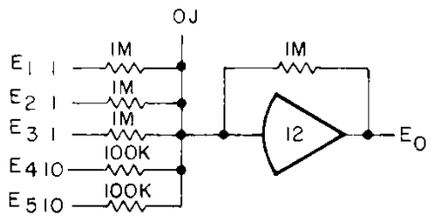
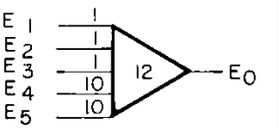
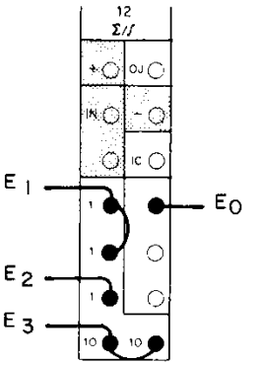
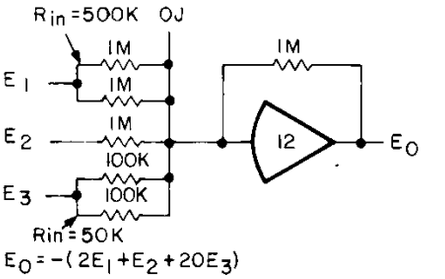
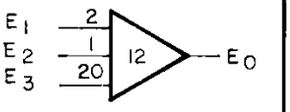
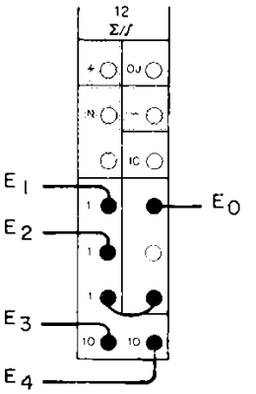
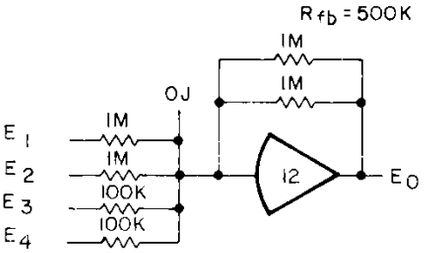
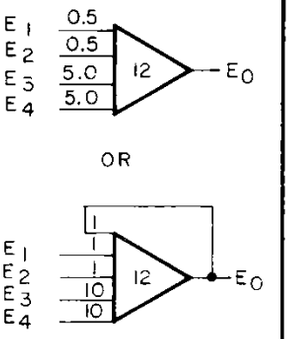
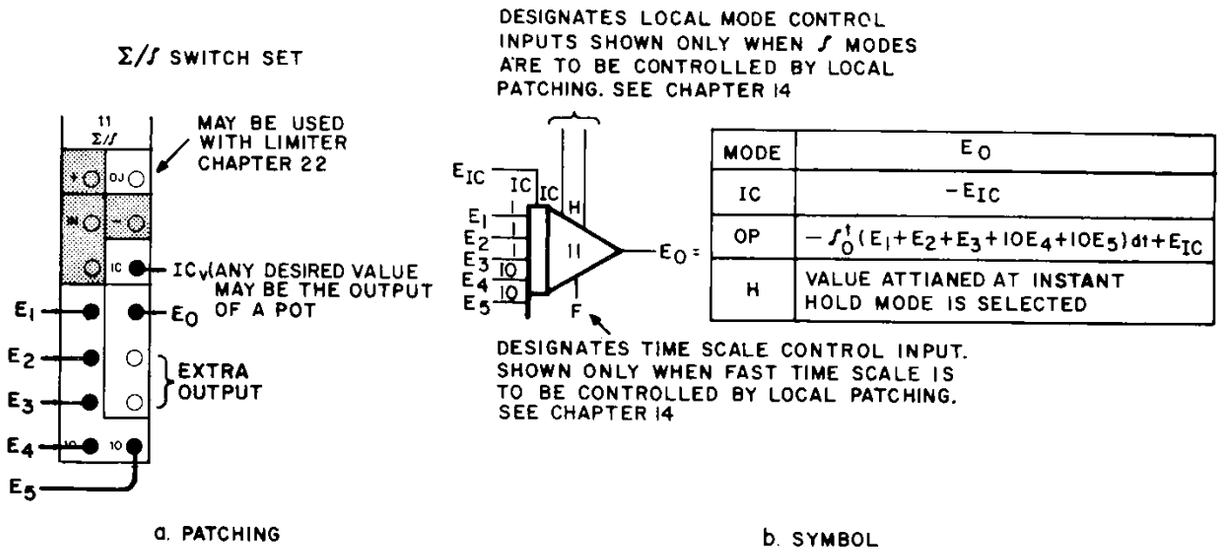
CONFIGURATION	PATCHING	EQUIVALENT CIRCURT	PROGRAM SYMBOL
a. STANDARD FIVE INPUT SUMMER (GAINS OF 1 AND 10)		 $E_O = -(E_1 + E_2 + E_3 + 10E_4 + 10E_5)$ NOTE: OJ CAN BE USED WITH A D/A SWITCH TO OBTAIN AN ADDITIONAL INPUT WITH A GAIN OF 100.	
b. OBTAINING GAINS OF 1, 2 AND 20		 $E_O = -(2E_1 + E_2 + 20E_3)$ NOTE: OJ CAN BE USED WITH A D/A SWITCH TO OBTAIN AN ADDITIONAL INPUT WITH A GAIN OF 100.	
c. OBTAINING GAINS OF 0.5 AND 5.0		 $E_O = -(0.5E_1 + 0.5E_2 + 5E_3 + 5E_4)$ NOTE: OJ CAN BE USED WITH A D/A SWITCH TO OBTAIN AN ADDITIONAL INPUT WITH A GAIN OF 50.	

Figure 21.2. Summer Patching: Σ/∫ Switch Released (Out)

21.4 USED AS AN INTEGRATOR

When operated as an integrator, the $\Sigma/\int$ produces the integral of the sum of a number of input variables. To use this unit as an integrator, simply set the $\Sigma/\int$ switch and patch the unit as required by the overall program. Figure 21.3 shows the basic patching requirements and program symbol for integration. Note that time scale selection is determined by the overall requirements of the program and is an important consideration when setting-up and patching the integrator. Refer to Chapter 14. If the integrator is to start (enter the Operate mode) at some predetermined value, an initial condition value (E_{IC}) must be patched to the analog IC input. The integrator will then go to the IC value ($E_0=-E_{IC}$) when the IC mode is selected. Then when OP is selected, integration will start at that value.



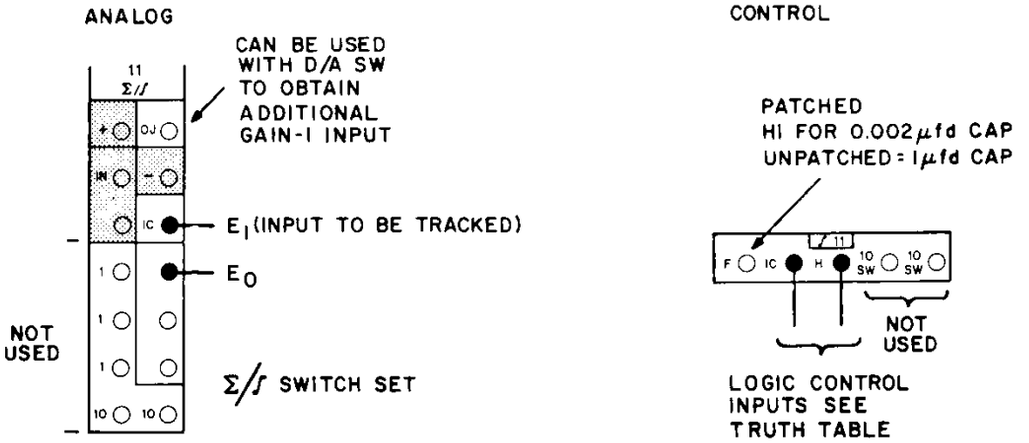
MODE	E_O
IC	$-E_{IC}$
OP	$-\int_0^t (E_1+E_2+E_3+10E_4+10E_5)dt+E_{IC}$
H	VALUE ATTIANED AT INSTANT HOLD MODE IS SELECTED

Figure 21.3. Integrator Patching: $\Sigma/\int$ Switch Set (In)

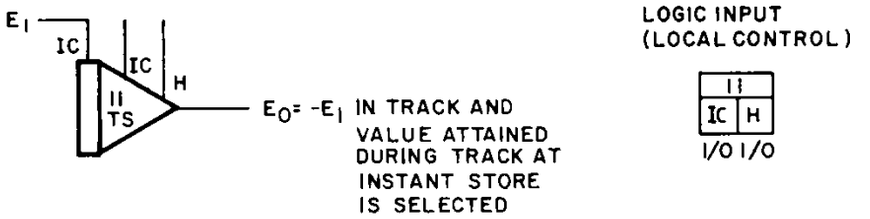
21.5 USED AS A TRACK/STORE AMPLIFIER

Since the modes of operation of an integrator can be individually controlled by logic patching, the $\Sigma/\int$ can be used as a two mode track-store amplifier. The Hold mode of an integrator corresponds to the Store mode of a track-store amplifier. Unlike the track-store unit described in Chapter 20, the $\Sigma/\int$ (when used as a T-S amplifier) cannot be initialized, and only a single gain-1 input for tracking is available. Additionally, the unit is limited to tracking a signal frequency of 5 Hz when using the normal hold capacitor, and 1 kHz when using the fast hold capacitor.

When used as a track-store amplifier the analog IC input must be patched to the signal to be tracked (Figure 21.4). Additionally, the unit must be cycled through the IC and H computer modes to attain the track/store function. This is accomplished by local control patching and by setting the $\Sigma/\int$ switch to enable the mode control logic. With the patching shown in Figure 21.4, the integrator IC mode corresponds to track, while the Hold mode corresponds to Store. The logic inputs can be provided from any desired logic source at the patch panel, such as the output of a comparator or any other device that will provide the desired switching. The truth table (Figure 21.4c) shows the control logic requirements. Using a combination of logic inputs other than those shown (such as simultaneous ones or zeros) will cause an erroneous solution and should not be permitted.



a. PATCHING



b. PROGRAM SYMBOL

LOGIC INPUT		MODE
IC	H	
1	0	TRACK (IC)
0	1	STORE (H)

c. TRUTH TABLE

Figure 21.4. Patching the Σ/∫ as a Track-Store Amplifier: Σ/∫ Switch Set (In)

21.6 USING THE 10SW CONTROL INPUTS

The Σ/∫ gain-10 inputs can be controlled in a similar manner as the general purpose electronic switches described in Chapter 28. These inputs can be program controlled by logic patching to connect or disconnect a gain-of-ten input to the amplifier at any desired point in the problem solution. The basic patching requirements, and the switch program symbols are illustrated in Figure 21.5. Note that these switches are an integral part of the Σ/∫ unit. Therefore, the diamond shaped symbol for an electronic switch is shown touching the input edge of the Σ/∫ program symbols. The logic signals used to control the switches can be any desired logic source at the patch panel, such as the output of a comparator, or any other deivce that will provide the required logic signal. If it is not desired to control the gain-10 inputs, merely leave the 10SW logic inputs unpatched. Then, whenever the Σ/∫ is in the Operate mode, the gain-10 inputs are connected to the amplifier input.

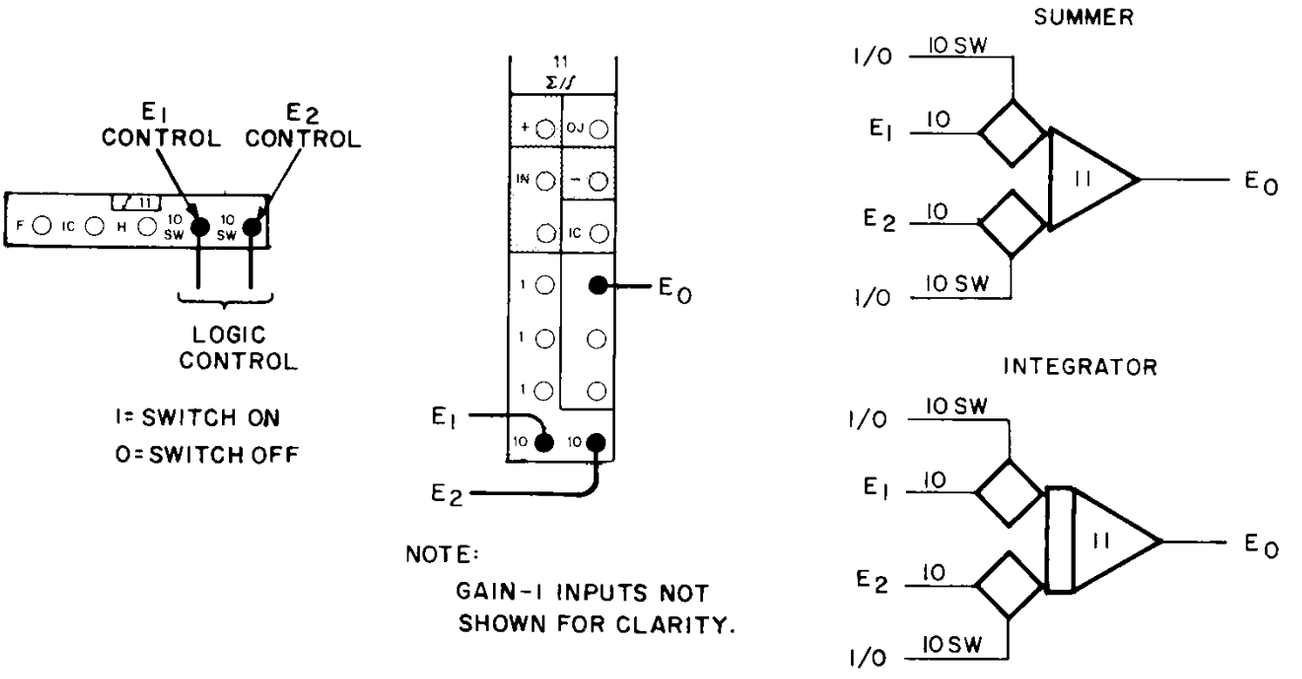
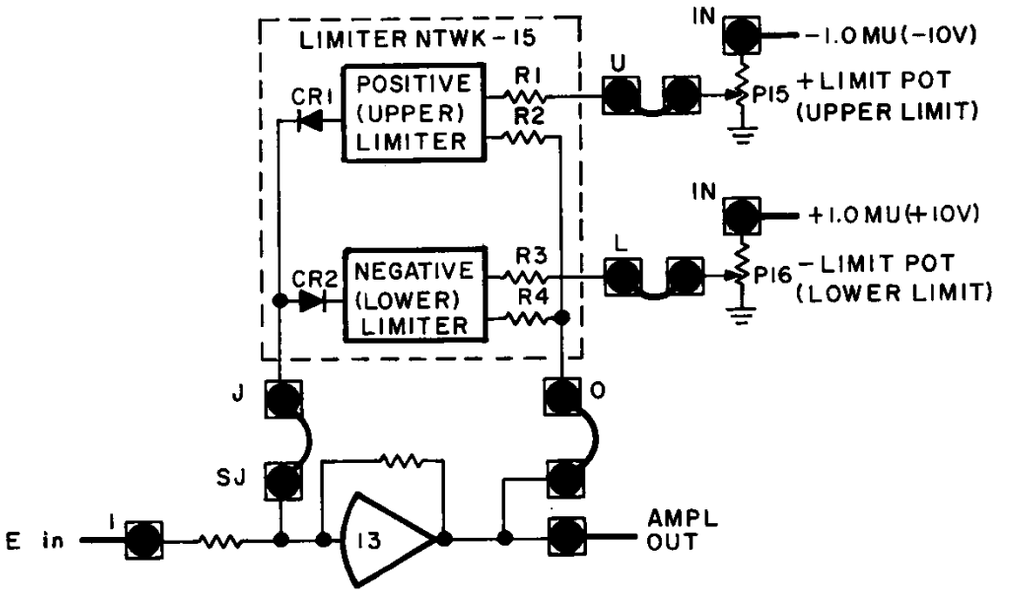
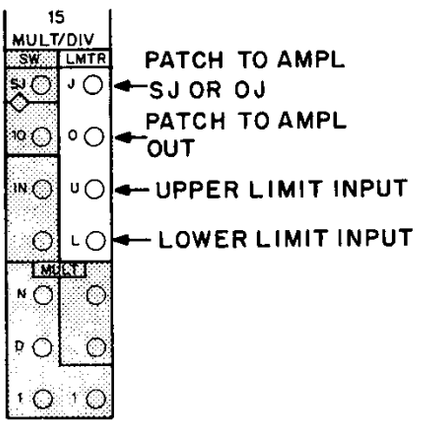


Figure 21.5. Gain –10 Switching



a. SIMPLIFIED CIRCUIT



b. PATCHING TERMINATIONS

Figure 22.1. Simplified Limiter Circuit: Used with Typical Amplifier

22.1 LOCATION AND IDENTIFICATION

There are three bi-polar feedback limiters in a fully expanded MiniAC. These limiters are terminated in patching modules 15, 25, and 35 (Figure 2.2). For ease of identification, the limiter terminations are color coded white and are identified by the designation LMTR on each patching module.

22.2 THE FEEDBACK LIMITER

The feedback limiter consists of positive and negative sections that may be used to limit (or constrain) either or both the positive and negative excursions of an analog signal. Each section is individually controlled using externally patched potentiometers. Figure 22.1 shows the basic patching and simplified circuit of a limiter connected to a typical amplifier. To understand the operation of a feedback limiter, assume that the +LIMIT potentiometer is adjusted to provide a reverse bias of −5.0 volts (.5 Machine Unit) to the positive limiter circuit. The positive limiter remains reversed biased (not conducting) until the amplifier output reaches +5.0 volts. At this time, the inputs to the positive limiter (summed through R1 and R2) equals zero. As the amplifier output tends to increase in the positive direction, the limiter circuit provides a feedback current to the amplifier summing junction (through diode CR1) that offsets the effect of the increasing input current. The negative limiter operates in the same manner to limit the negative output excursion of the amplifier, except that diode CR2 provides the current path that offsets input current.

22.3 USING THE LIMITER

22.3.1 PATCHING

The feedback limiter can be used with any amplifier in the MiniAC with its summing junction available at the patch panel. This is restricted to limiter use with Σ/HG, Σ/TS and $\Sigma/\int$ units. The limiter O (output) termination is patched to the output of the amplifier to be limited, and the J (junction) termination is patched to the corresponding amplifier summing junction (SJ for Σ/HG and Σ/TS units, OJ for $\Sigma/\int$ units). In addition, the U (Upper limit) and L (Lower limit) inputs must be patched to a potentiometer or to ± reference voltage as required by the limiting function to be performed. Figure 22.2 illustrates three common uses of the limiter and shows the patching requirements for each.

NOTE

Limits below ±0.1 Machine Unit (1.0 volt) are not reliable on all units.

22.3.2 LIMITER SETUP PROCEDURE

The two limit points (positive and negative) may be set independently. The limits are controlled by externally patching potentiometers (Chapter 18). The following procedure is recommended for setting the feedback limiter to the desired limit values.

1. Patch the setup circuit shown in Figure 22.3a.
2. Place the computer in SP and set the + and − LIMIT pots to the desired coefficient as readout on the DVM.
3. Place the computer in IC and using SELECTOR-1, read the amplifier output on the DVM.
4. Adjust the +LIMIT pot for the precise value desired.
5. Patch the amplifier with a positive reference input (Figure 22.3b) and adjust the −LIMIT pot to the desired value.

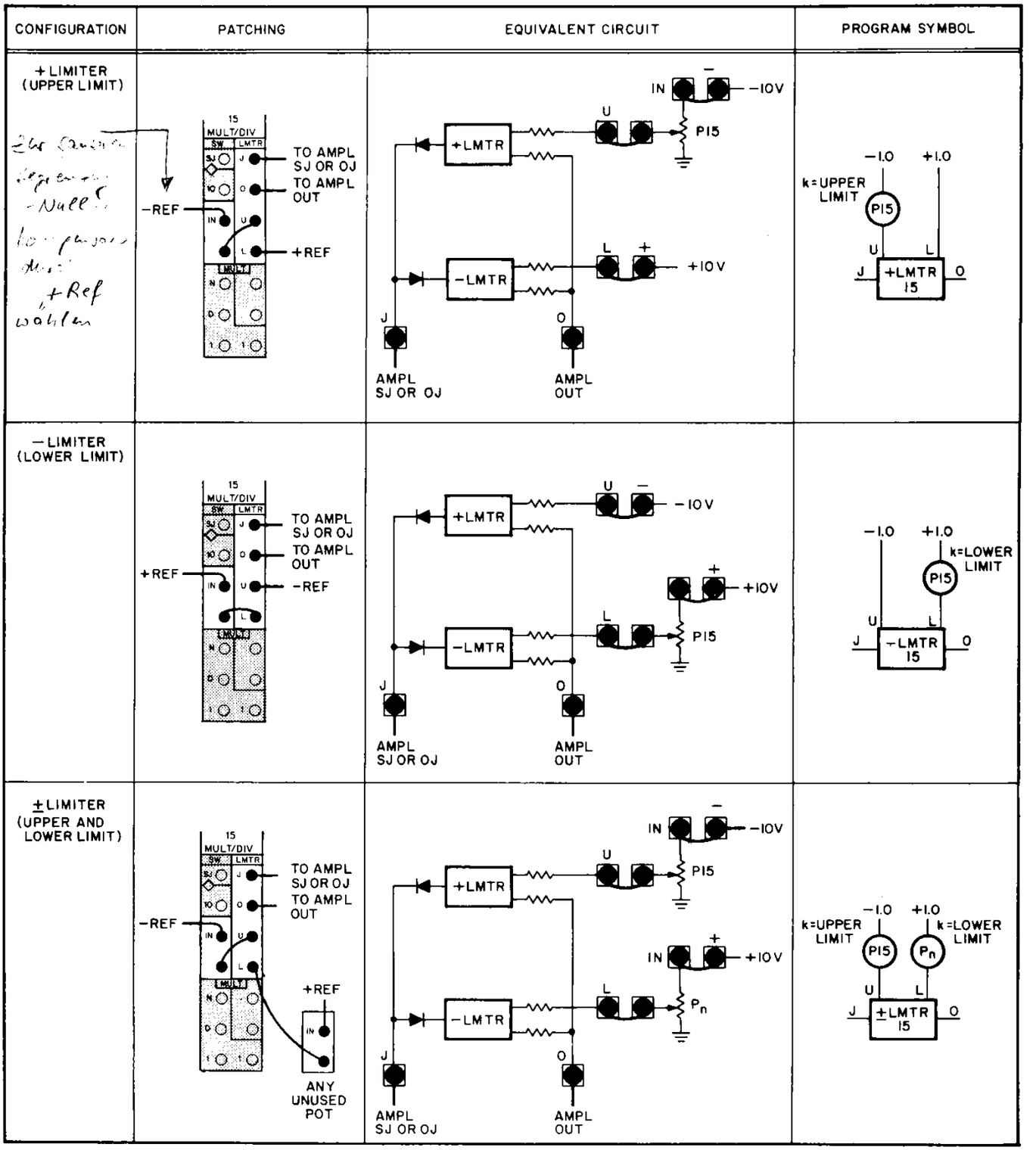
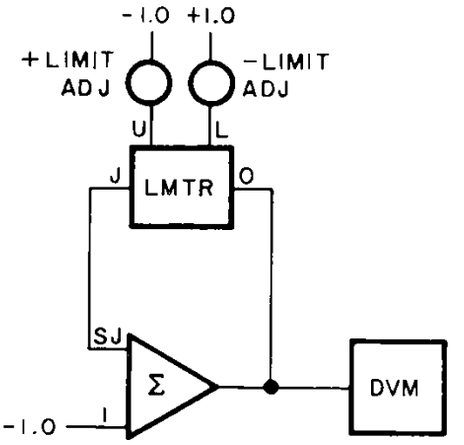
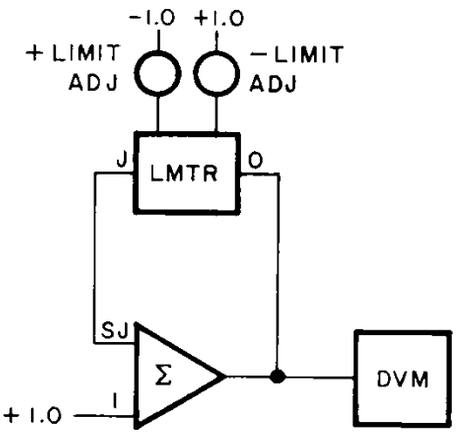


Figure 22.2. Limiter: Patching and Equivalent Circuit

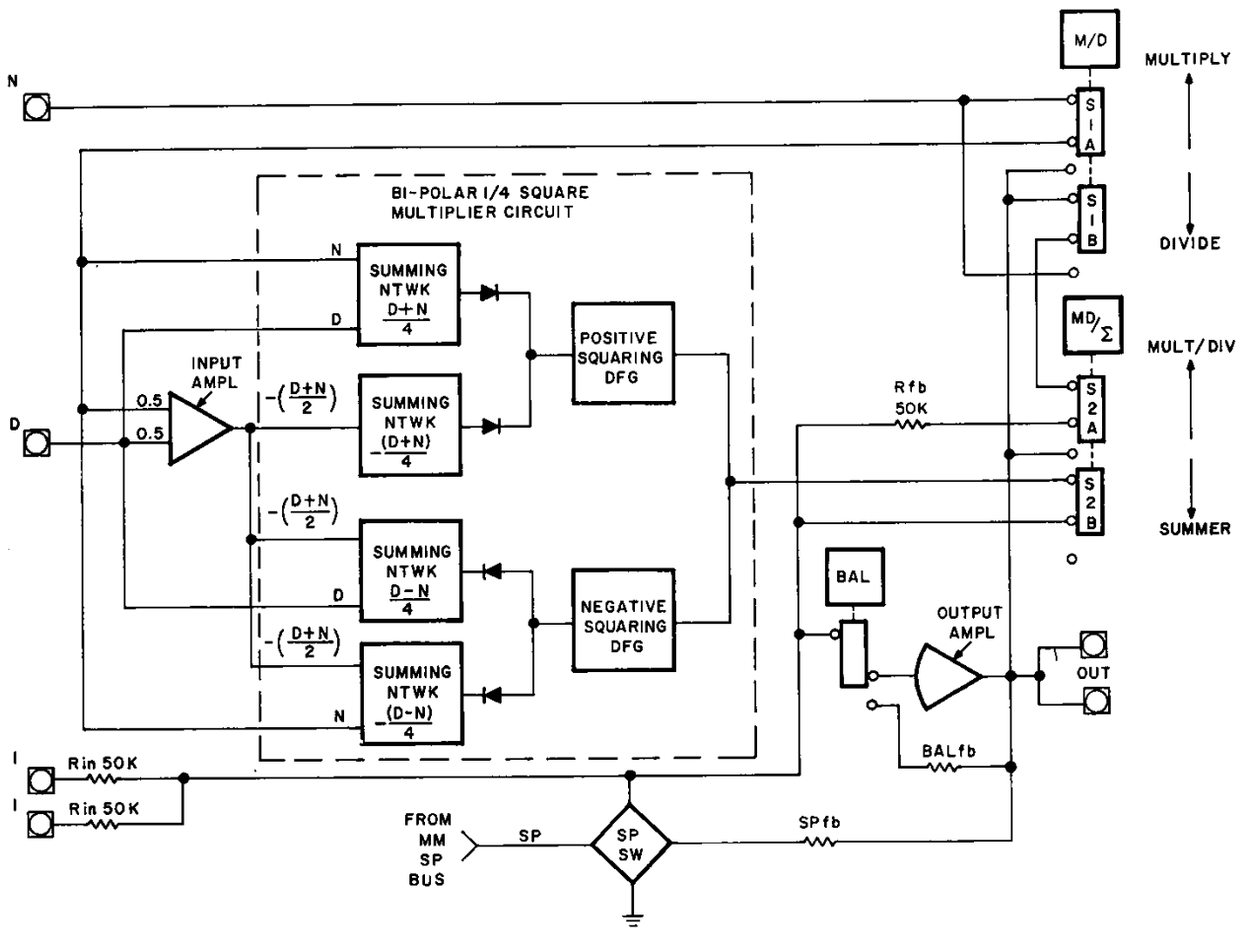


a. PLUS LIMIT ADJUSTMENT

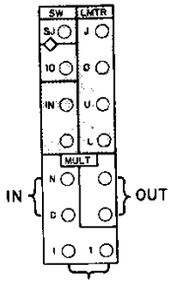


b. MINUS LIMIT ADJUSTMENT

Figure 22.3. Limiter Setup



a. SIMPLIED CIRCUIT



GAIN-1 INPUTS

b. PATCHING TERMINATIONS

Figure 23.1. Multiply/Divide Unit: Simplified Circuit

23.1 LOCATION AND IDENTIFICATION

There are three Multiply/Divide (Multiplier) Units in a fully expanded MiniAC. These units are housed in the card file directly behind the setup switch/overload indicator panel (Figure 2.7). Each multiplier location is identified by the numeric designation (15, 25, 35) visable at the top of card file when the front panel is lowered. These numbers correspond to the patch panel module address described in Chapter 2. One multiplier is terminated in each analog field of the patch panel (Figure 2.2). These terminations are identified by the module address (15, 25, 35) and by the patch panel MULT/DIV and MULT designations.

23.2 THE MULTIPLIER

23.2.1 INTRODUCTION

The MiniAC Multipliers (Figure 23.1) are of the bi-polar quarter square type and utilize an input ratio network, two squaring DFG's, a committed input amplifier, and an output amplifier that (under certain circumstances) can be used as a summer. Setup switches (M/D and MD/Σ), and individual patching configurations permit the unit to perform functions of multiplication, division, squaring, square root extraction, and summation.

23.2.2 MULTIPLICATION, SQUARING, AND SQUARE ROOT

When the M/D and MD/Σ switches are released (condition shown in Figure 23.1), the unit will respond to patched inputs at N and D to generate an output equal to: $-ND$; $-N^2$ (where N is equal to D); or $-\sqrt{N}$, (where N is equal to D) as determined by the input/output patching configuration and the nature of the input signals. When both switches are released, the N (numerator) and D (denominator) terminations are connected to the input amplifier. The output amplifier has a 50k ohm feedback resistor, and the amplifier summing junction is connected to the bi-polar quarter square multiplier summing junction.

23.2.3 DIVISION

When the M/D switch is set (and the MD/Σ switch is released), the unit will respond to patched inputs at N and D to generate an output equal to $-N/D$. Switch selection of the divide function places the bi-polar 1/4 square multiplier circuit into the feedback path of the output amplifier. With this arrangement, the D termination is connected to the input amplifier, and the N termination is connected to the summing junction of the output amplifier through a 50k ohm input resistor (formerly the output amplifier feedback resistor). With the M/D switch set, the value to be divided is patched to N (numerator), and the denominator is patched to D.

23.2.4 SUMMATION

Two gain-one inputs are provided, and except during amplifier balancing procedures, or when the computer SP mode is selected, are always connected to the summing junction of the output amplifier. This feature permits any compatible analog signals to be summed with the output of the DFG's when the MD/Σ switch is released. If the MD/Σ switch is set, the entire bi-polar 1/4 square multiplier circuit is isolated from the output amplifier. The amplifier is provided with two 50k ohm inputs and a 50k ohm feedback and can be used as a free component two-input summer or as an inverter. When used strictly as a summer or inverter, the MD/Σ switch should be set to ensure that the static error of the multiplier is eliminated.

23.3 USING THE MULTIPLIER

Figure 23.2 illustrates the patching configurations and setup switch requirements for each function that the multiplier can perform. This figure also illustrates the equivalent circuit and program symbol, and lists the various operating considerations for each function.

NOTE

When used for square root extraction, a-diode capacitor network must be patched from the external amplifer SJ to its output. See Figure 23.2-Square Root Extraction. This network can be manufactured locally and installed in series with a standard patch cord. This network consists of any standard diode and a 100 picofarad capacitor.

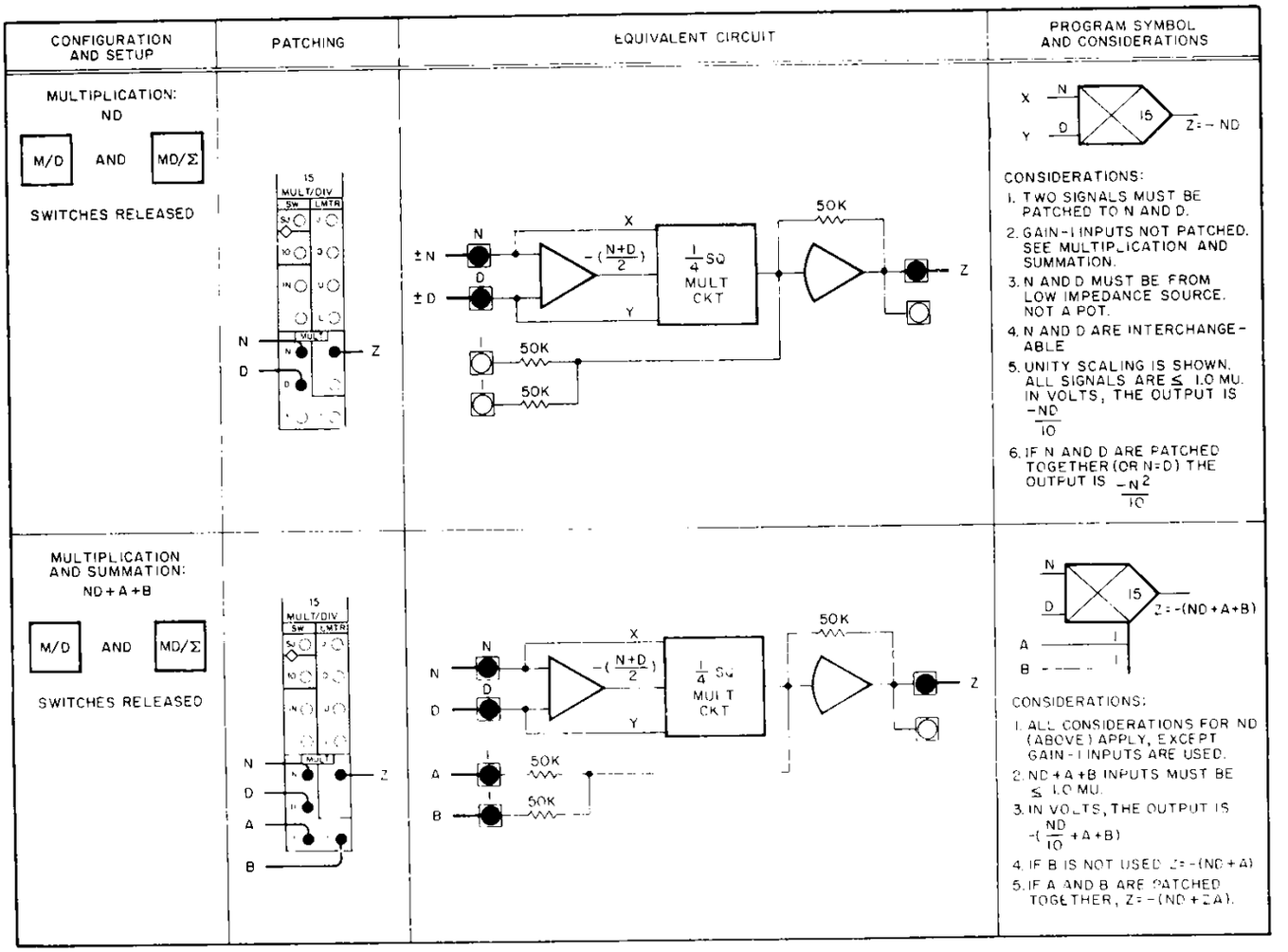


Figure 23.2. Multiply/Divide Unit: Setup and Patching

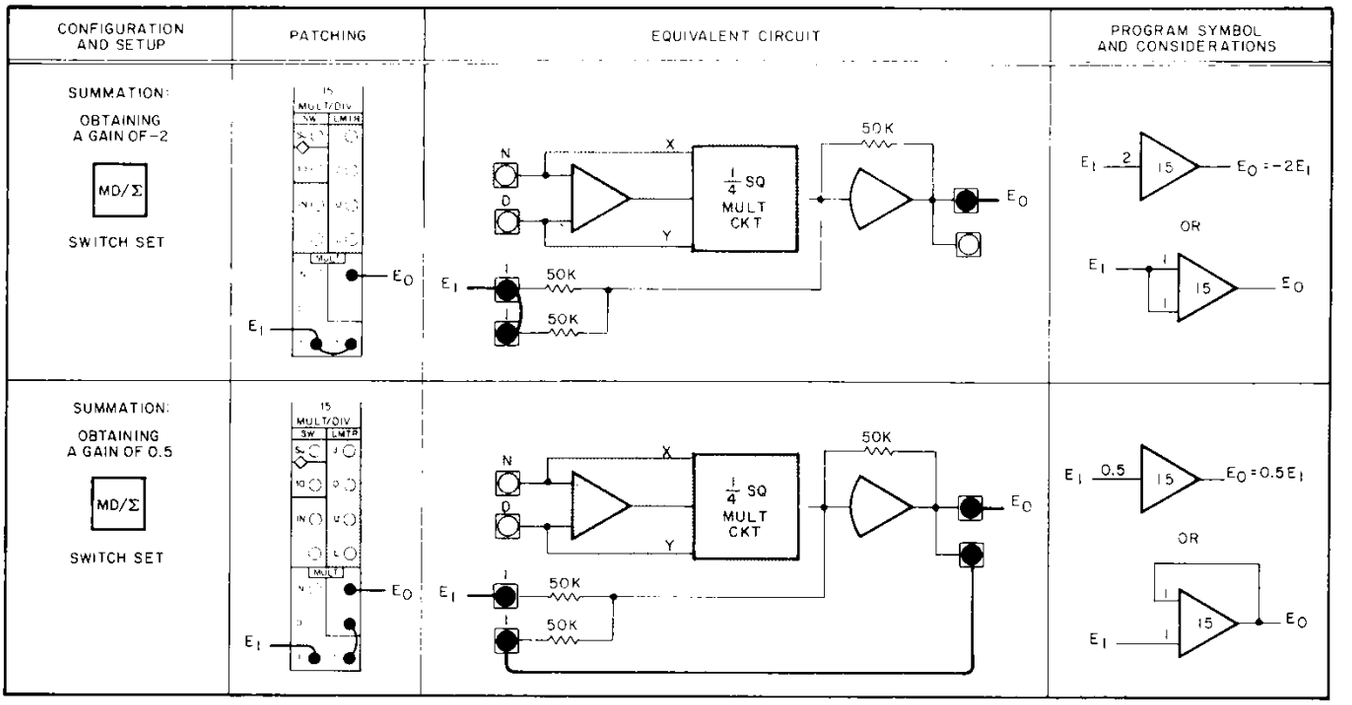


Figure 23.2. Multiply/Divide Unit: Setup and Patching (Cont)

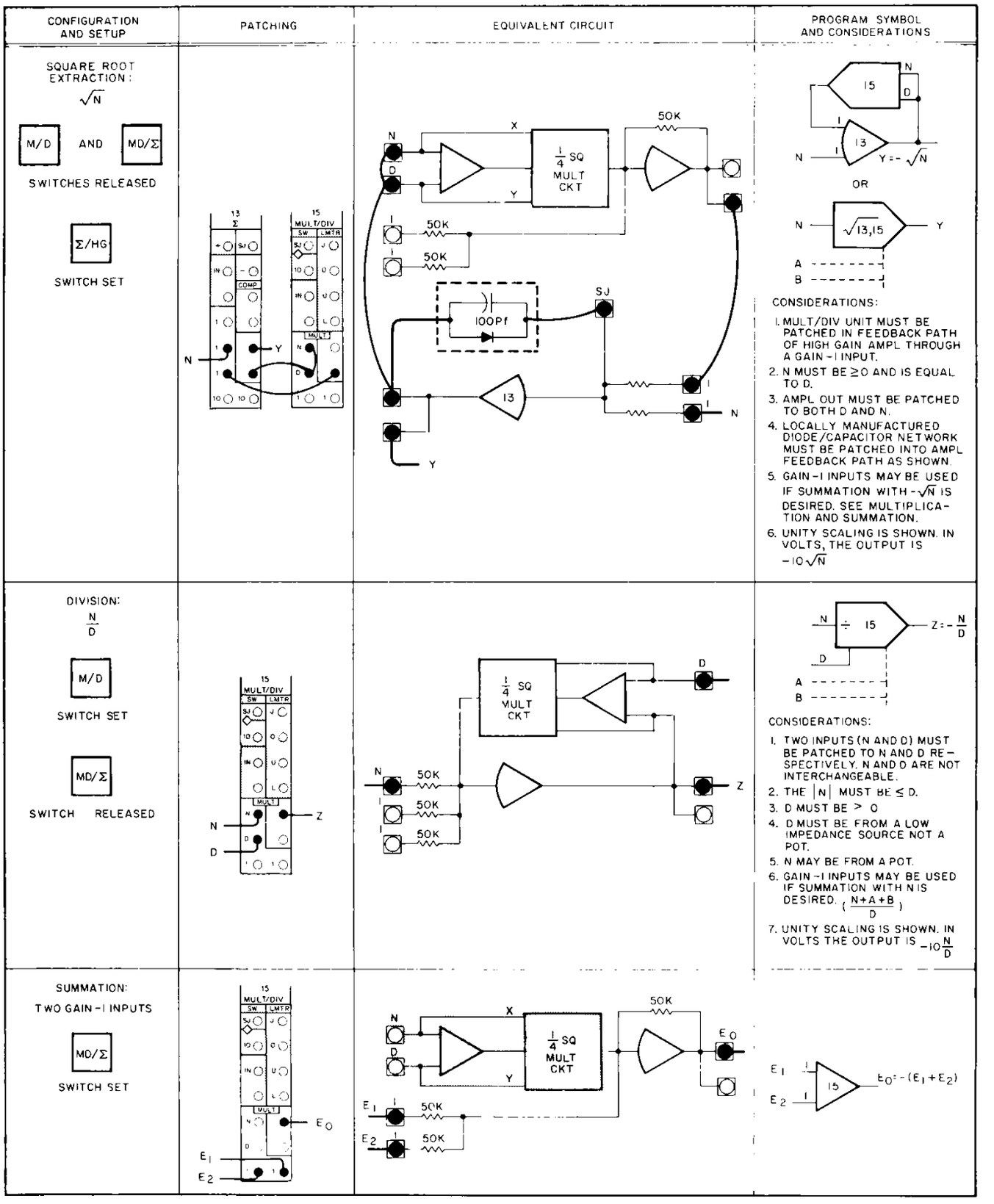
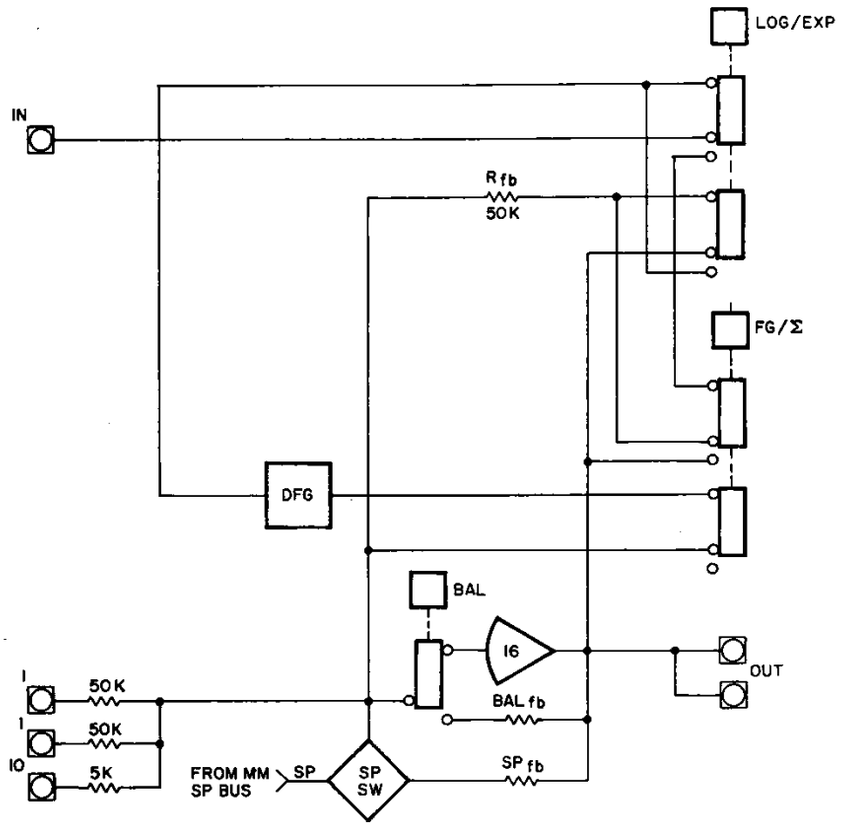
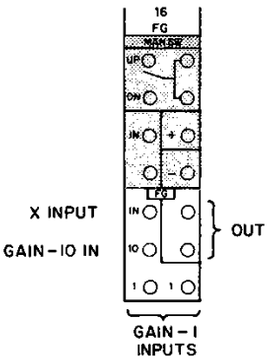


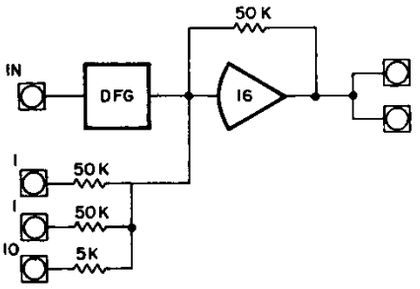
Figure 23.2. Multiply/Divide Unit: Setup and Patching (Cont)



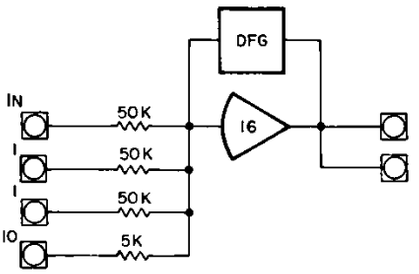
a. SIMPLIFIED CIRCUIT



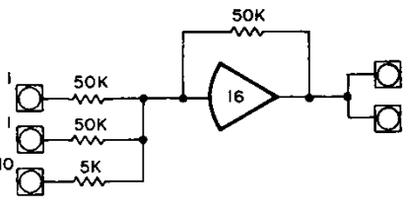
b. PATCHING TERMINATIONS



1. LOG X FUNCTION GENERATOR



2. EXPONENTIAL FUNCTION GENERATOR



3. THREE INPUT SUMMER

c. EQUIVALENT CIRCUIT

Figure 24.1. Log/Exponent Function Generator: Simplified and Equivalent Circuits

24.1 LOCATION AND IDENTIFICATION

There are two log/exponential function generators in a fully expanded MiniAc. These units are housed in the card file directly behind the overload indicator and setup panel (Figure 2.7). The log/exponential function generators are identified by the numeric designation (16 and 26) visable at the top of the card file when the front panel is lowered. These numbers correspond to the patch panel module address described in Chapter 2. These function generators are terminated in patch panel analog fields one and two. The terminations are identified by the module address (16 and 26) and by the patch panel FG (function generator) designations.

24.2 THE FUNCTION GENERATOR

24.2.1 INTRODUCTION

The MiniAc function generators (Figure 24.1) include a DFG (diode function generator), and an output amplifier that (under certain circumstances) can be used as a free component. The setup switches (designated LOG/EXP and FG/Σ) and individual patching configurations permit the unit to produce a natural logarithm function, an exponential function, or to simply act as a three input summer.

24.2.2 LOGARITHMS

When the LOG/EXP and FG/Σ switches are released (condition shown in Figure 24.1), the unit will respond to an input (X) to produce the function 0.1 lnx as X varies from 0.0001 to 1.0 machine units (0.001 to 10.0 volts) as shown in Figure 24.2

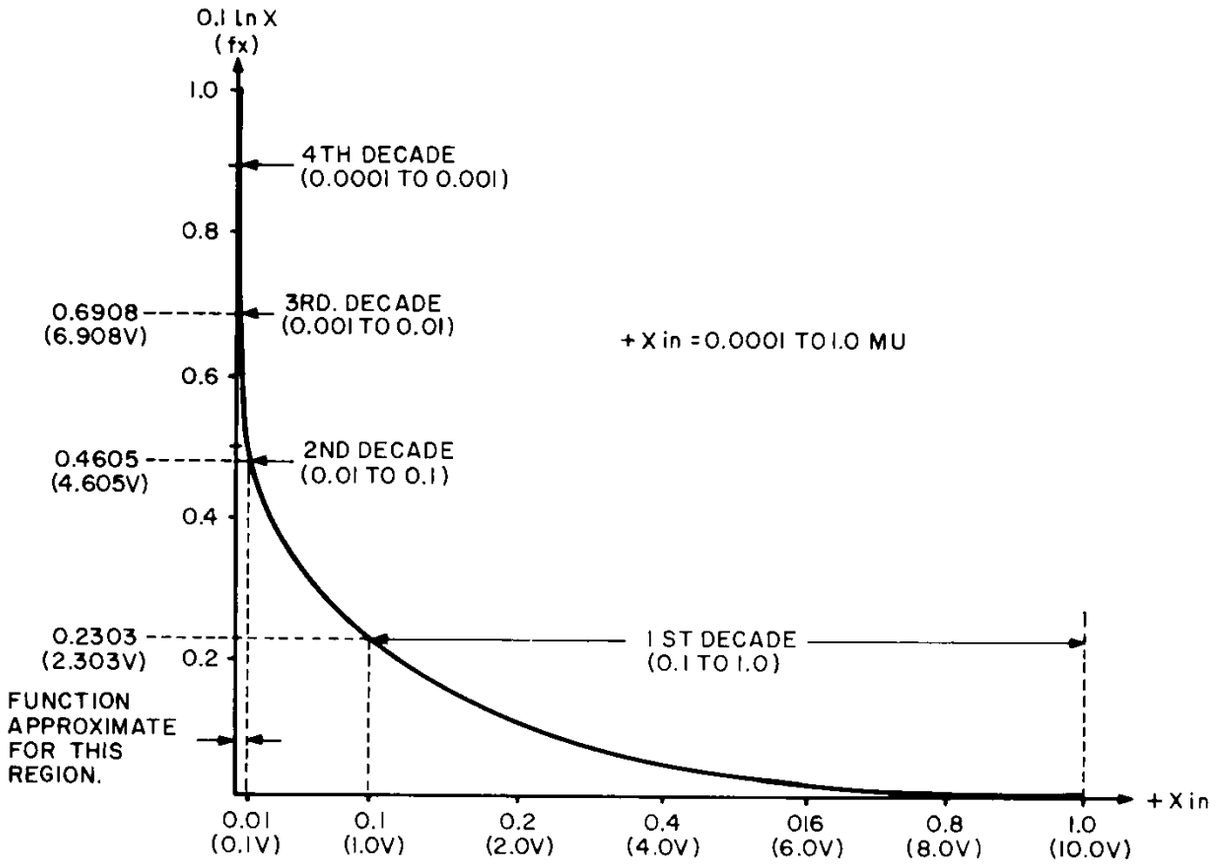


Figure 24.2. The Logarithmic Curve (0.1 lnx)

When both switches are released, the IN termination is connected to the input amplifier and the DFG. The output amplifier is provided with a 50k ohm feedback resistor, and the amplifier summing junction is connected to the DFG summing junction. With this arrangement, the output amplifier is committed to the unit and may be used to sum other inputs with the log output.

If the X input increases linearily from 0.0001 machine unit to 1.0 machine unit; the function generator will yield the output waveform function shown in Figure 24.2.

24.2.3 EXPONENTIAL FUNCTIONS (ANTILOGARITHMS)

When the LOG/EXP switch is set (and the FG/Σ switch is released), the unit will respond to the X input to produce the function e^{-10X} (Figure 24.3) as X varies from 1.000 to zero machine unit. Switch selection of this exponential function places the DFG network (Figure 24.1) in the feedback path of the output amplifier. With this arrangement, the IN termination is connected to the input of the amplifier through 50k ohm input resistor (formerly the output amplifier feedback resistor).

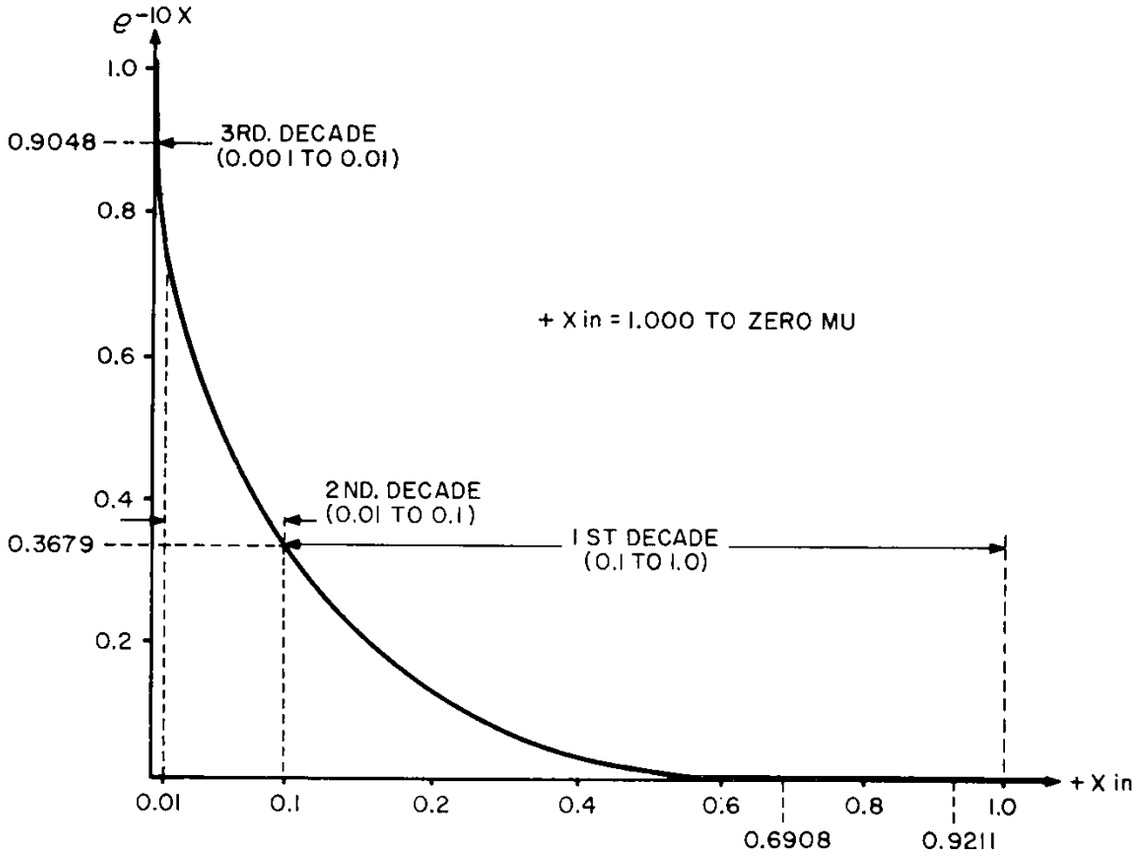


Figure 24.3. The Exponential (Antilog) Curve (e^{-10X})

24.2.4 SUMMATION

Two gain-one inputs and a gain ten input are provided, and except during amplifier balancing procedures, are always connected to the summing junction of the output amplifier. If the FG/Σ switch is set, the entire function generator network is isolated from the output amplifier. The amplifier is then provided with two 50k ohm inputs, a 5k ohm input, and a 50k ohm feedback and can be used as a free component three-input summer.

24.3 USING THE FUNCTION GENERATOR

Figure 24.2 illustrates the patching configurations and setup switch requirements for each use of the function generator. This figure also illustrates the equivalent circuit and program symbol, and lists the various operating considerations for each function. Two function generators can be interconnected to generate X^n as shown in Figure 24.5.

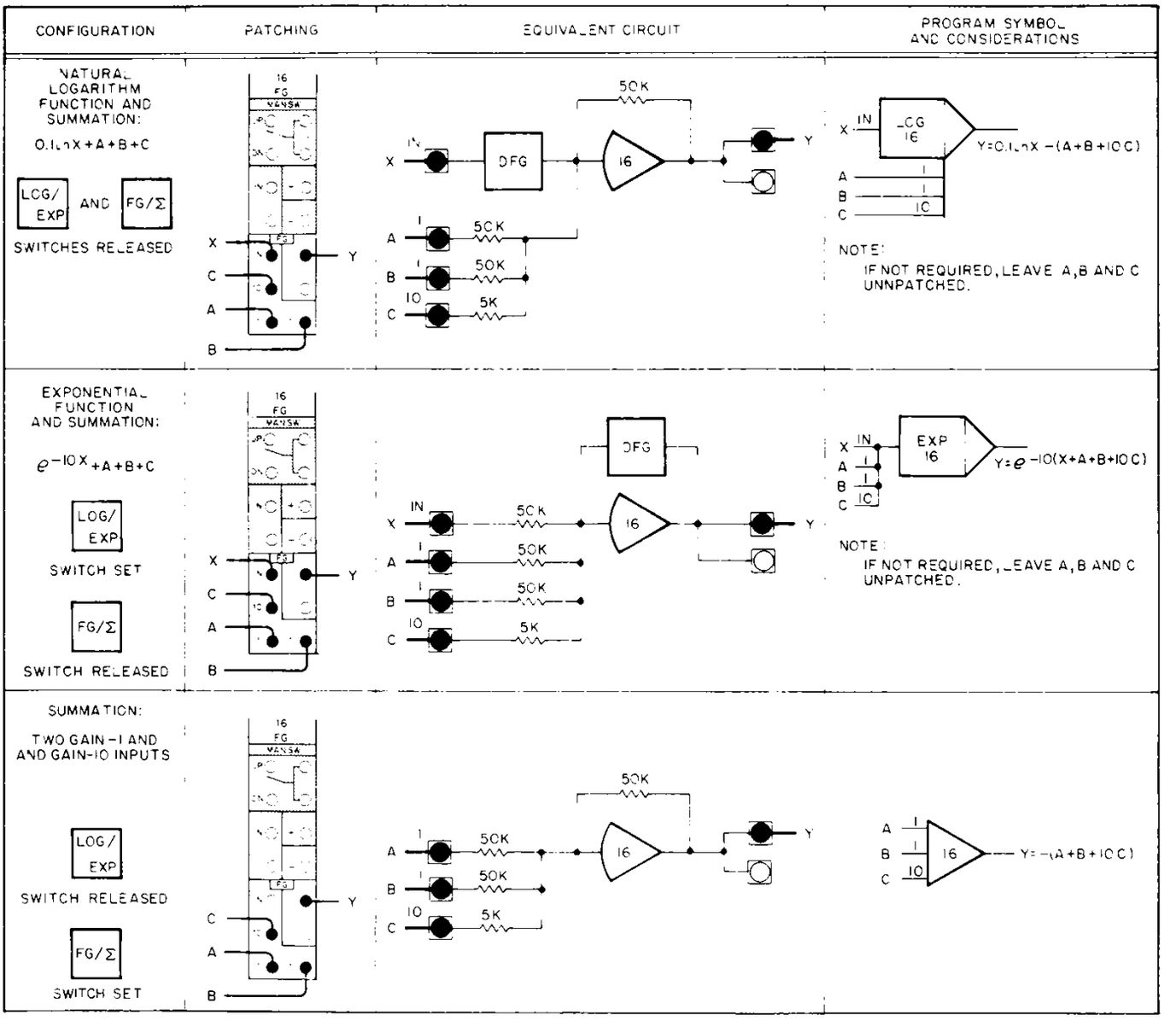


Figure 24.4. Log/Exp Function Generator: Setup and Patching

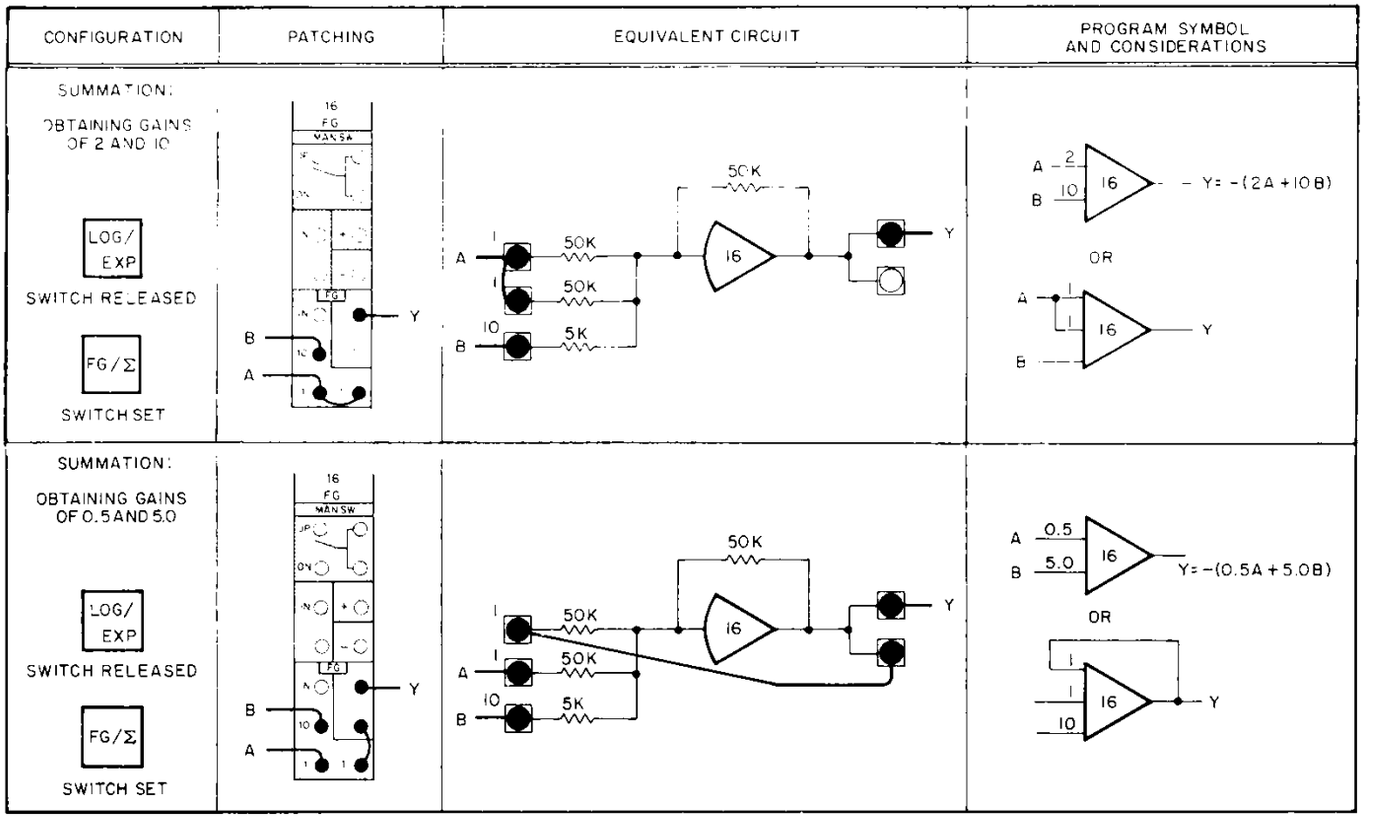
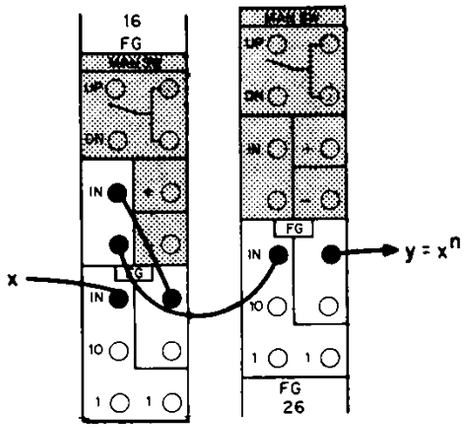
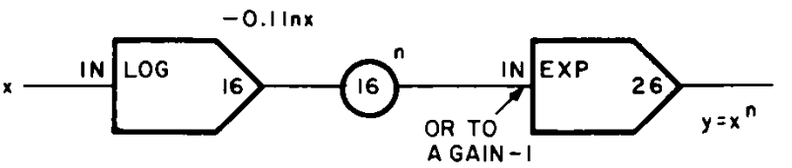


Figure 24.4. Log/Exp Function Generator: Setup and Patching (Cont)

PATCHING

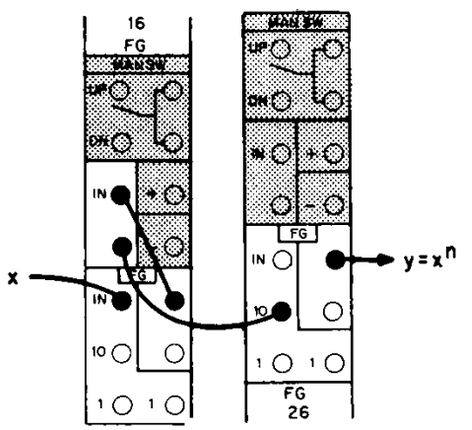


SYMBOL

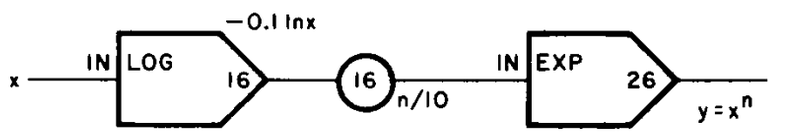


a. x^n: WHERE $0 \leq x \leq 1$, AND $0 \leq n \leq 1$.

PATCHING



SYMBOL



b. x^n: WHERE $0 \leq x \leq 1$, AND $0 \leq n \leq 10$.

Figure 24.5. Generating X^n

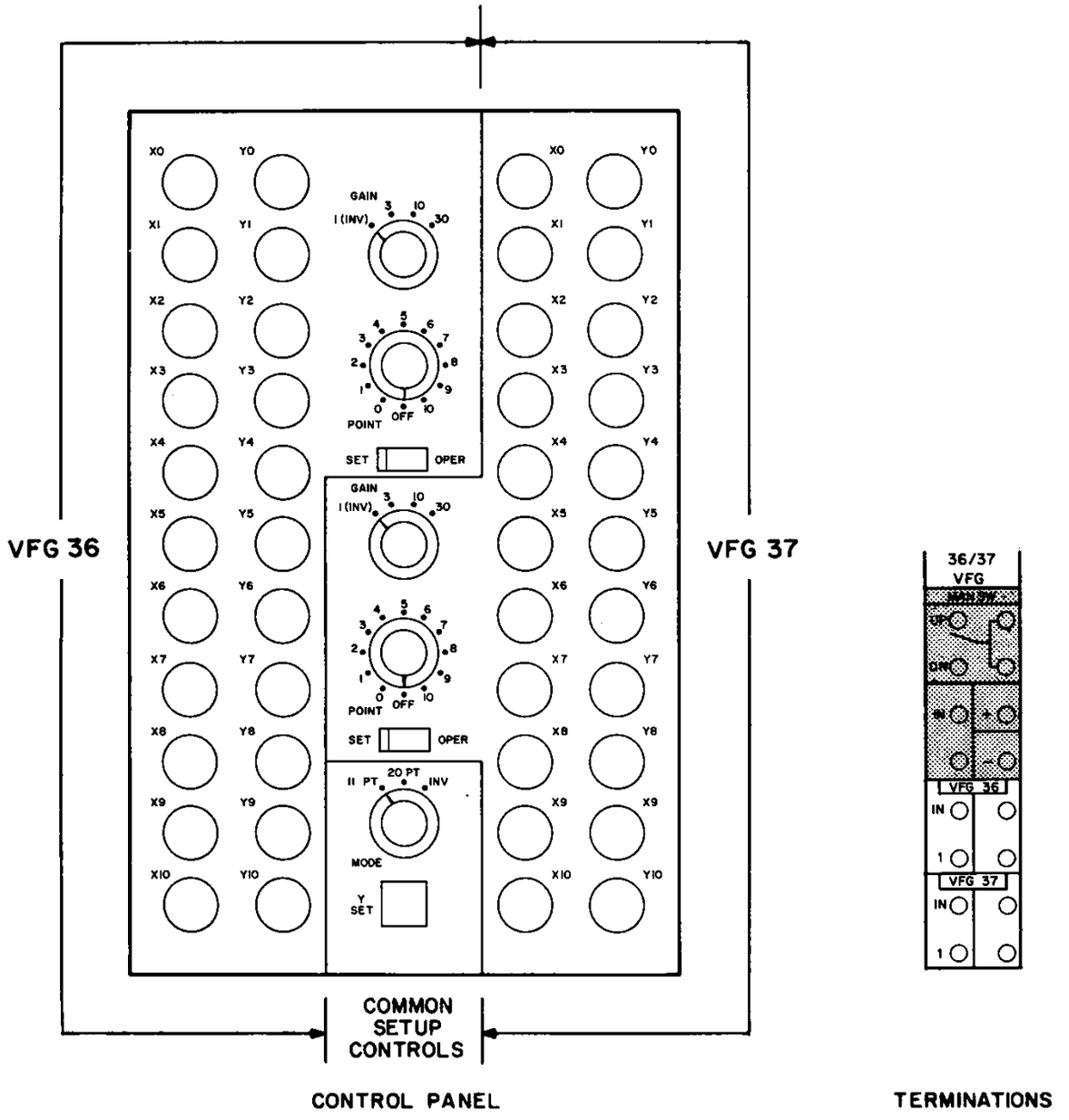


Figure 25.1. VFG: Controls and Patching Area

VARIABLE FUNCTION GENERATOR

25.1 LOCATION AND IDENTIFICATION

The fully expanded MiniAC has provisions for generating arbitrary analytic functions using a dual variable function generator (VFG). This device (Figure 25.1) can be operated as two separate and independent 10 segment VFG's, as a combined VFG where more than 10 segments are required, or (under certain conditions) as inverters. The dual VFG is physically located directly behind the removable VFG cover panel (Figure 2.1), and is terminated in the last patching module (VFG-36/37) of analog field three (Figure 2.2). The terminations for each ten segment section of the VFG are identified by the patch panel and readout address designations VFG36 and VFG37.

25.2 THE VFG

25.2.1 GENERAL OPERATION

The VFG uses straight-line segments to generate an arbitrary analytic function. The VFG is a variable gain device, in which gain varies as a function of the input signal. The input level at which gain changes occur is determined by biased diodes. The magnitude and direction of gain change is determined by the segment contribution or sinking of current applied to the summing junction of an output amplifier.

A typical variable gain diode circuit for the MiniAC VFG is shown in Figure 25.2. The voltage (Eb) at the arms of the Xn pot determines the reverse bias (or breakpoint voltage) applied to the diodes. As the X input voltage goes more positive, it substracts from Eb. When Xin is greater than Eb, the diode conducts. This places the segment network (consisting of the diode, the Yn + 1 pot and the input amplifier) in series with Rin and the output amplifier. This network forms the input impedance to the output amplifier and controls the overall circuit gain. The setting of Yn + 1 determines the magnitude and direction of current applied to the output amplifier summing junction. Assuming a fixed feedback resistor (Rfb) around the output amplifier, the overall gain or slope of the entire circuit depends on the adding or subtracting of current at the output amplifier summing junction. If the yn + 1 pot is adjusted to the right of mid-point (clockwise) the output (fx) will increase in the negative direction (negative slope). If adjusted to the left of mid position, the output increases in the positive direction (positive slope). Several such segment groups (biased to conduct at different levels) are connected in parallel to form a complete function. Each segment contributes its own slope, either adding to or substracting from the slopes of other conducting networks, so that the slope at any point is the algebraic sum of the slopes generated by all conducting segments.

Maximum versatility of the unit is achieved by providing an initial value of X and initial value of Y that permit starting a function at any desired values of X and Y (fx) and in any quadrant. The only requirement is that the starting point (origin) of X must be to the extreme left of the function. This will always be the most negative (or least positive) value of X. The Y value. for any given X point can be either positive or negative as long as the function is single valued.

NOTE

When the VFG is unused or is operated in the INV Mode, false overloads may be indicated. If a valid function is stored in the VFG, the possibility of false overloads may be eliminated by patching the IN terminations to ground. If an invalid function is stored, false overloads may be indicated even with the IN terminations grounded. If this occurs, set the X0 and Y0 pots at mid position and turn all other pots fully clockwise.

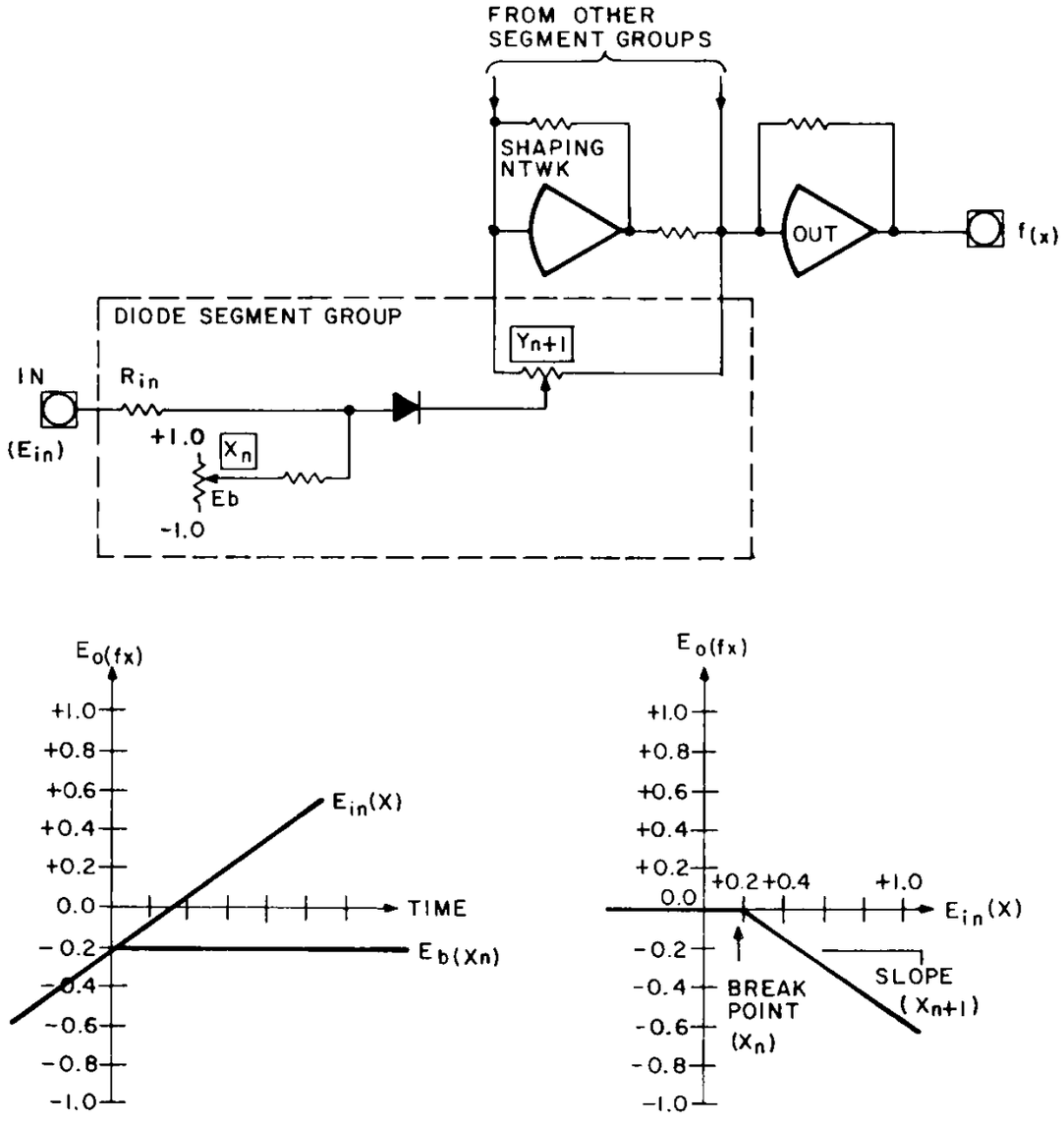


Figure 25.2. Typical Diode Segment

25.2.2 MODES OF OPERATION

The dual VFG is equipped with switching circuits that permit manual selection for operation as two independent 11 point or 10-segment VFG's as a combined 20 point (19 segment VFG with a free inverter) or two independent inverters. Figure 25.3a illustrates the equivalent circuit of the VFG when the 11-point mode is selected. Note that each ten segment section (VFG36 and VFG37) has its own X input (IN) as well as independent outputs.

When the 20-point mode is selected (Figure 25.3b) both 10 segment DFG sections are paralleled. Note that the DFG associated with VFG37 is no longer connected to its own output amplifier. Therefore, the module address of VFG36 must be used when selecting the VFG for readout. In the 20-point mode, the output amplifier for VFG37 becomes a free component and may be used as an inverter. When placed in the INV mode (Figure 25.3c), both output amplifiers become free components and may be used as inverters.

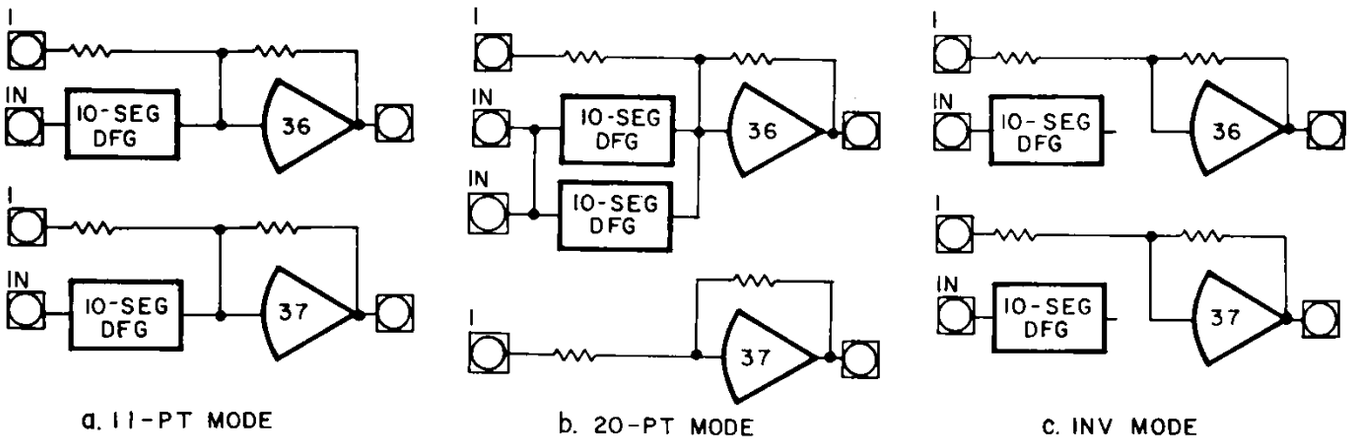


Figure 25.3. VFG Modes and Equivalent Circuits

25.2.3 THE VFG CONTROLS

As illustrated in Figure 25.1, the dual VFG control panel is divided into three distinct and separate areas. The segment and setup controls for VFG36 are to the left of the panel, and those for VFG37 are to the right. The area at the bottom of the panel contains controls that are common to both VFG sections.

X0-X10 POTS:

The vertical row of potentiometers (designated X0 to X10) at the outer edge of each VFG section are used to set the values of X required to setup the function. The X0 and X10 pots select the initial and final values of X. The controls designated X1 through X9 select the breakpoints.

Y0-Y10 POTS:

The vertical row of potentiometers designated Y0 through Y10 on each VFG section are used to set the Y output or $f_{(x)}$. The Y0 and Y10 pots correspond to the initial value of $f_{(x)}$ and the final values, respectively. The Y1 through Y9 pots select the value of $f_{(x)}$ for the corresponding value of X.

GAIN SELECTOR:

The GAIN selector (uppermost switch) on each VFG section controls the change in slope (gain) of the output amplifier. There are four gains (1, 3, 10 and 30) available.

POINT SELECTOR:

The POINT selector (middle switch) in each VFG section is used during VFG setup to select the point being set. Positions 0 - 10 correspond to X0-X10 and Y0-Y10 controls. The point switch must be placed in the OFF position after the function is setup and prior to placing the VFG in service.

SET/OPER SWITCH:

The SET/OPER switch (bottommost switch) on each VFG section is placed at SET during the VFG setup procedure. The VFG cannot be placed in service until the switch is set at the OPER position.

MODE SELECTOR:

The Mode Selector is located in the central area between the VFG sections and is common to both. When set at 11PT each VFG section can be as independent 11 point (10 segment) VFG's. When at the 20PT position, the VFG sections are combined to provide a single 20 point VFG. The combined output is taken from VFG36 and the output amplifier of VFG37 becomes a free inverter. When the MODE selector is set at INV, both output amplifiers become free inverters.

Y-SET PUSHBUTTON:

This momentary pushbutton switch is used during VFG setup while setting the Y0-Y10 potentiometers. Depressing this switch displays the value of Y on the DVM.

25.3 HOW TO SET UP THE VFG

25.3.1 TABULATING VALUES OF X AND $F_{(x)}$

Since the variable function generator approximates a given curve by a series of straight-line segments, the first step in the setup of any VFG is to determine the location of the breakpoints ("corners") in order to fit the curve as smoothly as possible. Usually the desired curve is given in graphical form, and the programmer must determine (by inspection of the graph) how best to separate the function into a series of straight-line segments, and thereby generate a table of values of x and $f_{(x)}$ for the set up. Sometimes the function is given in tabular form; usually the result of experimental measurements, or sometimes the result of a series of calculations.

If the function is given in tabular form it is tempting to simply set up the DFG to the values of x and $f_{(x)}$ from the table, especially if the number of data points in the table happens to coincide with the number of segments available in the DFG. The difficulty with this approach is that the distribution of the data points that define the function will probably not be the best distribution of breakpoints for straight-line approximation. For example, the original data may have been obtained for equally-spaced values of the input variable x, whereas it is generally *not* a good idea to use equally-spaced breakpoints for segment approximation. It is better to plot the data points, pass a smooth curve through them, and determine a good breakpoint location from this smooth graph. In any case, it is desirable to know what the function looks like before trying to set it up, which means it should be manually plotted before set up.

Therefore, the setup procedure in this chapter assumes that the function is defined graphically. The procedure consists of determining good breakpoint locations, tabulating the values of x and $f_{(x)}$ at these points, and setting up the function from this table of values. The dual VFG's have common setup controls and individual segment controls that permit direct set up of values of x and $f_{(x)}$ and both the input x and output $f_{(x)}$ may be read directly on the DVM during set up. However, it is a good idea to obtain a continuous plot of $f_{(x)}$ versus x to make sure that the function has been set up correctly. Such a plot should be a part of the problem documentation, along with the circuit diagram, listings of pot-settings, amplifier assignments, and so on.

25.3.2 DETERMINING BREAKPOINT LOCATION

Given a smooth curve, where should the breakpoints be located to approximate it with minimum error? With a bit of experience, a good programmer can come very close to the optimim breakpoint location simply by inspection of the curve. The following general rules serve as a rough guide to this technique.

Keep in mind the total number of breakpoints available on the VFG. Most functions of practical interest may be adequately represented with 10 segments or less; a few may require more. The MiniAC ten-segment VFG's are capable of being paired (20PT mode) to handle this.

Start out by locating the areas where the function is nearly straight; the individual segments in such areas may be relatively long. In between these areas will be the areas of rapid slope-change; the breakpoints should be concentrated here.

As a general rule, it does not pay to locate two breakpoints closer together than about 4% of full scale (0.04 unit, or 0.4 volt on a ten-volt computer). If two breakpoints are spaced more closely than this, they tend to blend into one because of the characteristics of diodes, which are not perfect switches. This effect, which rounds off the corners of the function, gives a smoother output and is beneficial in determining breakpoint locations. In case of very sharply curved functions, it may be necessary to space breakpoints slightly closer – say, as close as 2% of full scale (0.02 unit).

As an example, consider Figure 25.4. This curve represents an arbitrary function, scaled on a unit basis, so that input x varies from zero to unity. The procedure starts by noting the two areas where the function is almost straight, namely $0 \leqslant x \leqslant 0.2$ and $0.58 \leqslant x \leqslant 0.74$. There would be little point in putting any breakpoints here, so the process starts by drawing two fairly long segments to approximate the function over these intervals.

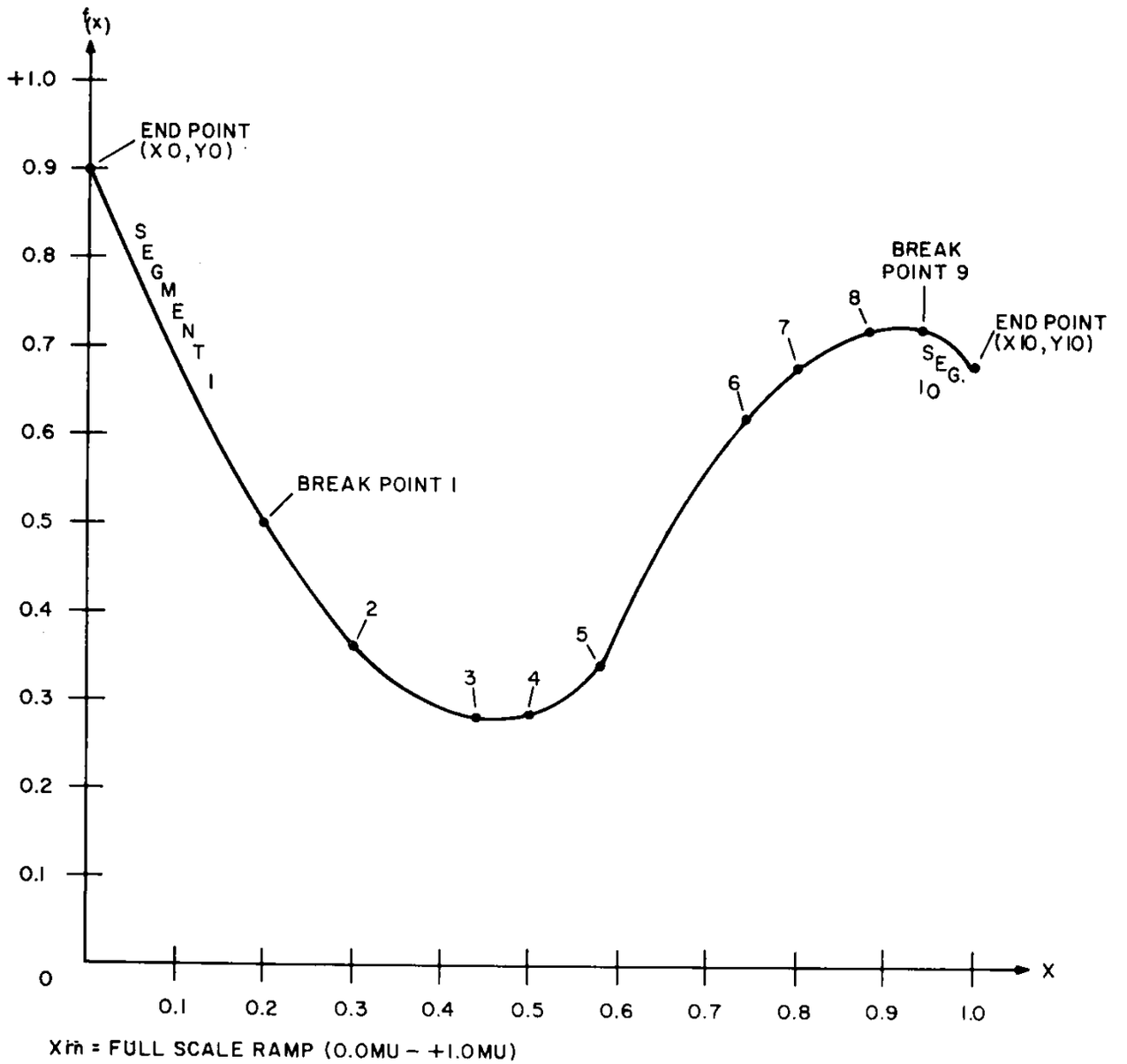


Figure 25.4. Sample Ten Segment Function

The intervals from 0.2 to 0.58 and from 0.74 to 1.0 are the intervals where the function curves noticeably. Since this is a relatively "mild" function, it may be easily approximated by ten segments. Hence, there are *nine* breakpoints to be determined. Note that the number of breakpoints is one less than the number of segments. For example, a two-segment function has one breakpoint – where the two segments join; a three-segment function has two breakpoints, and so on. The *endpoints* of the interval are essentially fixed, and are not counted as breakpoints.

We have nine breakpoints to divide between the two intervals of rapid slope-change. Since the first interval is longer (and the slope changes somewhat more), more breakpoints should be put in this interval. A five-four split was arbitrarily selected.

Based on the five-four split, the breakpoint location in Figure 25.4 was determined. No claim is made that is the absolute optimim, but is a fairly good fit, and easily arrived at. In using the visual technique, there is no substitute for experience; the above rules and example are intended only as a rough guide.

25.3.3 VFG SETUP THEORY

Each breakpoint, or "corner" in a VFG curve represents a diode that is changing state; as the input voltage (x) gives more positive, or less negative, more and more diodes start to conduct. Each diode (when it starts to conduct) increases or decreases the slope of the curve. The change in slope for a given segment is determined by adjusting a potentiometer (the Y pot). For example, Figure 25.5 shows the effect of changing the Y pot for segment 6 on a partially setup function. The various lines extending from X5 represent the output slope for several settings of the Y pot.

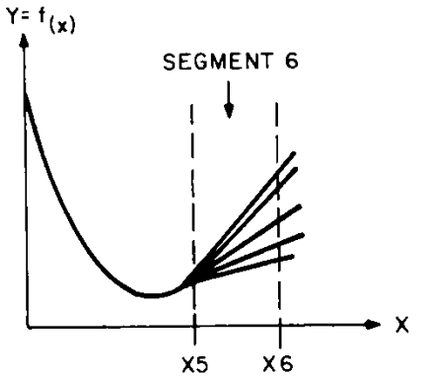


Figure 25.5. Effect of Changing Slope Pot Y6 on VFG Output

The VFG setup procedure is arranged so that the operator does not observe slopes, but rather values of X and Y ($Y=f_{(x)}$). In otherwords, the value of X6 is set first using the X6 pot then the slope for segment 6 (or the value that the function will assume at X6) is set using the Y6 pot.

In practice, the Y pot for a given segment has an effect on the previous segment. This is due to the fact that the diode is not a perfect switch. The "corner" at X5 in Figure 25.5 is somewhat rounded, and a slight shift in X5 will occur when setting X6. In most cases, this shift will cause the VFG output to more closely approximate the actual curve.

25.4 SETTING 11-POINT FUNCTION

25.4.1 GENERAL

Once the values of x and y are tabulated, the function setup is quite straight forward. The VFG has potentiometers for adjusting the X and Y coordinates at the starting point and the positive end of each segment. This arrangement permits direct setting of a function from a table of values; first adjust x (which may be read directly on the DVM), and then press a switch (Y-SET) and set Y.

As an example, consider the function in Table 25.1. This is the same function that was given graphically in Figure 25.4. Note that since 10 segments are to be used, the function is tabulated at 11 points, including the 2 endpoints and 9 breakpoints. For this example, the endpoints are set at 0 and 1.

Table 25.1. 11-Point Table of Values

X	POINT Selector	$Y = F_{(x)}$	X	POINT Selector	$Y = F_{(x)}$
0.00	0	0.90	0.58	6	0.34
0.20	1	0.50	0.80	7	0.66
0.30	2	0.36	0.88	8	0.72
0.38	3	0.30	0.94	9	0.72
0.44	4	0.28	1.00	10	0.68
0.50	5	0.28			

In setting such a function some programmers find it easier to set all x values (breakpoints) first, and then the Y or function values. However, it is recommended that you set the first breakpoint X, then set the corresponding value of Y and continue alternating in this manner. The setup procedure makes use of the DVM and signal selector, and the VFG control panel.

25.4.2 PRELIMINARY SETUP PROCEDURES

1. Select the computer IC mode.
2. Set VFG MODE switch at 11PT.
3. Place SET-OPER switch at SET.
4. Set GAIN switch at 1.
5. Set POINT switch at 0.
6. Rotate all X pots (X0 through X10) fully clockwise.
7. Using SELECTOR-1 on the analog control panel, address the VFG (36 or 37) that is to be set up.
8. Proceed to set the desired values of X and Y as described in the following paragraph.

25.4.3 SETTING X AND Y

After all preliminary steps (Paragraph 25.4.2) are completed, setup the values of X and Y. The following list outlines the basic order of procedure.

1. Adjust X0 for the initial value of X, as displayed on the DVM.
2. Depress and hold the Y-SET pushbutton and set the Y0 pot until the desired initial value of Y is displayed on the DVM. Release the Y-SET pushbutton.

NOTE

The adjustments in Steps 1 and 2 select the initial value of X and Y to define the origin of the function.

3. Select POINT-1 and then adjust X1 for the first breakpoint.
4. Depress Y set and set Y1. This forms the first segment.
5. Using the POINT Select and Y-SET switches alternately set the X and Y pots for segments 2 through 10 as in Steps 3 and 4.
6. After setting the last X,Y point it may be necessary to "touch up" the function by repeating Steps 3 through 5.
7. After segment 10 (POINT-10) is set, place the POINT switch at OFF the SET/OPER switch at OPER, and plot the function using the procedure in Paragraph 25.6.

To aid the operator in setting up a function, a sample 11 point (ten segment) setup procedure is given in Table 25.2. This table defines the order of procedure and specifies which switch and pot to operate at any given time. Table 25.2 applies to the ten segment function illustrated in Figure 25.4 and tabulated in Table 25.1; and is provided to permit the operator to set up a sample function and thereby become familiar with the setup operations.

Table 25.2. Sample Order of Procedure (Ten-Segment Function)

Step	1 (Note 1)	2	3	4	5	6 (Notes 2 and 3)
	Set POINT Selector To:	Set Value of X Using This Pot	Observe Value of X on DVM (0.XYZ-1)	Depress and Hold This Switch to Read Y	Set Y Using This Pot	Observe Value of Y on DVM (0.XYZ-1)
a.	0	X0	0.000	Y SET	Y0	0.900 (Origin of Y or 1st endpoint)
b.	1	X1	0.200	Y SET	Y1	0.500 (First Segment)
c.	2	X2	0.300	Y SET	Y2	0.360 (2nd Segment)
d.	3	X3	0.440	Y SET	Y3	0.280
e.	4	X4	0.500	Y SET	Y4	0.280
f.	5	X5	0.580	Y SET	Y5	0.340
g.	6	X6	0.740	Y SET	Y6	0.620
h.	7	X7	0.800	Y SET	Y7	0.680
i.	8	X8	0.880	Y SET	Y8	0.720
j.	9	X9	0.940	Y SET	Y9	0.720 (Segment 9)
k	10	X10	1.000	Y SET	Y10	0.680 (10th Segment - final value of Y or second endpoint)

Notes:

1. SET OPER switch must be at SET throughout procedure and must be at OPER prior to starting a program run.
2. Release Y SET pushbutton before advancing to next POINT selector position.
3. GAIN selector should be preset to 1. If insufficient slope is encountered (desired value of Y cannot be obtained), switch to next higher gain position, rotate all X pots fully clockwise and repeat all previous adjustments. See Paragraph 25.4.4.

25.4.4 INSUFFICIENT SLOPE

During the above procedure, it is possible that a particular function value cannot be obtained. If the Y pot is rotated to the end of its travel and the function value still has not been reached, the first thing to do is make sure you are setting the right pot. Also make sure the POINT selector is in the right position. Assuming that no such error has been made, the problem is probably one of insufficient slope.

For each segment there is an upper limit to the amount of slope *change* that is can introduce; this is the amount produced with the slope pot fully clockwise. With the GAIN switch in the 1 position a maximum slope change of about 1.4 may be obtained with one segment. The 30 position of this switch provides for slope changes as great as 42.

The GAIN switch increases the feedback resistance on the output amplifier, thus providing increasing gain when at the 3, 10 and 30 positions. For very steep or sharply curving functions, the GAIN 30 position may be necessary.

The effective resolution of the Y pots decreases with higher gain; at the high settings a small motion of the pot produces a large change in the output, making accurate set up more difficult. Also, electrical characteristics such as bandwidth are degraded with high gain settings. For this reason, it is not desirable to use more gain than necessary in generating a particular function.

If at any time during the setup procedure, an increase gain is necessary, you must return to Y0 reset all X pots (X0 through X10) fully clockwise, and reset the entire function. When using GAIN-30, avoid prolonged use of the Y-SET pushbutton. If setting a Y value takes more than a few seconds, release the Y-SET switch, and then depress it again. This is necessary to eliminate the effects of drift incurred by the internal setup T/S Amplifier.

25.4.5 TRIMMING ADJUSTMENTS

If a check of the Y values is made after the entire function is set, a shift of about 0.010 to 0.020 unit (100 to 200 millivolts), and rarely greater than 0.050 may be noticed. This is because the value at a given breakpoint (X) is set by changing the slope of the previously set breakpoint, and causes a small shift in the previously set function value (Y). A rounding of the function will occur around the breakpoint which will make the output appear as a curve rather than a series of straight lines. A series of trimming adjustments may be made where a more accurate output is desirable. These adjustments should be made in the same order as the original setup — starting at the origin and working outward. The trimming process goes much faster than the original setup, since the X values do not have to be set again, and only small changes in Y values are required.

An alternate method of trimming is to plot or display the function (Paragraph 25.6) and make trimming adjustments as desired to more closely approximate the actual curve. Remember that the adjustments must be made starting at the most negative (or least positive) value.

25.5 SETTING A 20-POINT FUNCTION

Provision is made for operating the dual VFG as a single unit with up to nineteen segments. This method of operation provides increased accuracy through the use of more segments. Basically, the function is set in exactly the same manner as functions having ten segments or less, except as follows: the MODE switch must be set at 20PT; the X10 control on VFG36 is not used; and the X0, Y0, Y1 and GAIN controls on VFG37 are not used. The following list outlines the basic order of procedure for setting function of more than 10 segments:

1. Perform the following preliminaries:
   a. Select computer IC mode.
   b. Set VFG MODE switch at 20PT.
   c. Set GAIN-36 and GAIN-37 switches at 1.

NOTE

GAIN-37 not used during 20PT mode. Must be set at 1 if VFG37 is used as an inverter.

   d. Set both SET/OPER switches to SET.
   e. Set POINT selector-36 to 0 and POINT selector-37 to OFF and rotate all X pots fully clockwise.
2. Set initial value of X using X0-36.
3. Depress Y-SET and set initial value of Y using Y0-36.
4. Set POINT selector-36 to 1 and set X for the first breakpoint using X1-36.
5. Depress Y-SET and set the value of Y using Y1-36.
6. Continue alternating X and Y settings for the corresponding POINT, selector positions (1 through 9) on VFG36 as in Steps 4 and 5.

NOTE

If insufficient slope is encountered at any point, increase the GAIN-36, rotate all X pots fully clockwise, and reset the entire function.

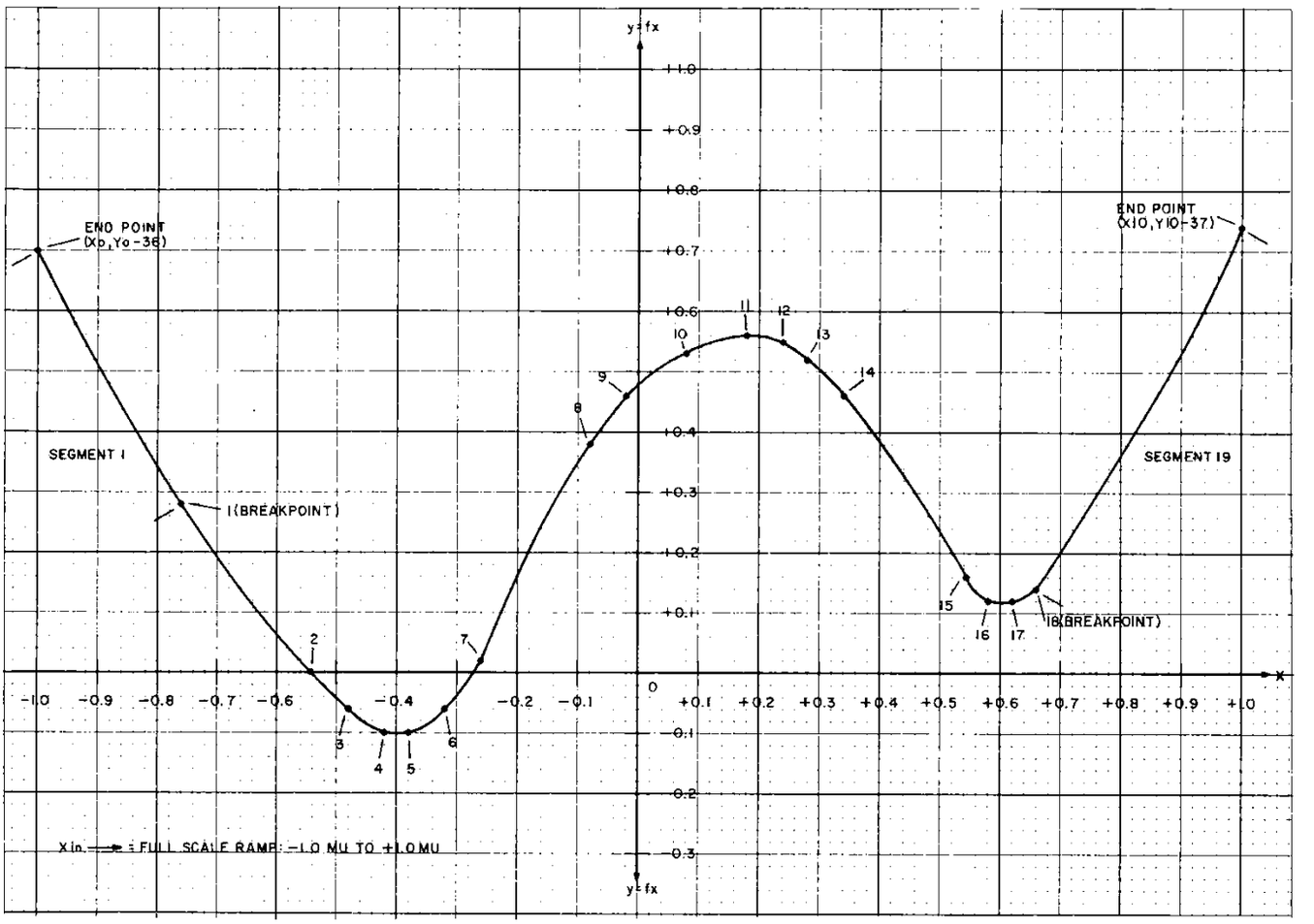
7. After completing the Y9 setting set POINT selector-36 at OFF and POINT selector-37 at 1.
8. Set as value of X for the tenth breakpoint using X1-37.
9. Depress Y-SET and set Y using Y10-36.
10. Continue to alternately set X and Y for the corresponding POINT selector positions 2-10 on VFG37 and the corresponding X and Y Pots.

NOTE

If you have numbered the breakpoints the POINT switch setting and the X and Y pot numbers will be nine less than the point actually being set. Breakpoint 11=VFG37 POINT-2, X2 and Y2.

11. After setting Y10, set POINT-37 at OFF set the SET/OPER switch at OPER, and plot the function (Paragraph 25.6).

A sample 20-point function is illustrated in Figure 25.6. This function has three regions where it curves rather sharply, separated by regions where it is relatively straight. The first curved regions lies in quadrant three of the graph, the second curved region lies in quadrant two, and the third lies mostly in quadrant one. Note that more breakpoints are required in quadrant one than in the other quadrants. The table in Figure 25.6 shows the table of values of X and Y for the illustrated function and shows the relationship of the breakpoints to the POINT selector and the X, Y pots. This data is provided to permit the operator to set this sample function for familiarization.



BREAK POINT	POINT SEL	X	Y		
FIRST END POINT	0	−1.00	+0.70	0	STARTING AT X 0−36 AND Y0−36, ALTERNATELY SET THE X VALUE AND THEN THE Y VALUE FOR THE CORRESPONDING POINT SEL −36 POSITIONS (0 − 9 IN ORDER)
1	1	−0.76	+0.28	1	
2	2	−0.54	0.00	2	
3	3	−0.48	−0.06	3	
4	4	−0.42	−0.10	4	
5	5	−0.38	−0.10	5	
6	6	−0.32	−0.06	6	
7	7	−0.26	+0.02	7	
8	8	−0.08	+0.38	8	
9	9	−0.02	+0.46	9	AFTER SETTING Y9 − 36, SET POINT SEL −36 AT OFF. POSITION 10 NOT USED
VFG 36	OFF				VFG 36
VFG 37			+0.53	10	AFTER SETTING POINT SEL −37 AT 1, SET X VALUE WITH X1−37. THEN SET Y VALUE WITH Y10−36. Y0 AND Y0 NOT USED. VFG 37
SEE NOTE 2				0	
10	1	+0.08		1	
11	2	+0.18	+0.56	2	CONTINUE TO ALTERNATELY SET VALUES OF X AND Y FOR THE CORRESPONDING POINT SEL −37 POSITIONS (2−10 IN ORDER).
12	3	+0.24	+0.55	3	
13	4	+0.28	+0.52	4	
14	5	+0.34	+0.46	5	
15	6	+0.54	+0.16	6	
16	7	+0.58	+0.12	7	
17	8	+0.62	+0.12	8	
18	9	+0.66	+0.14	9	
FINAL END POINT	10	+1.00	+0.74	10	AFTER SETTING Y10−37, SET POINT SEL AT OFF.
	OFF				

SEE NOTE 1

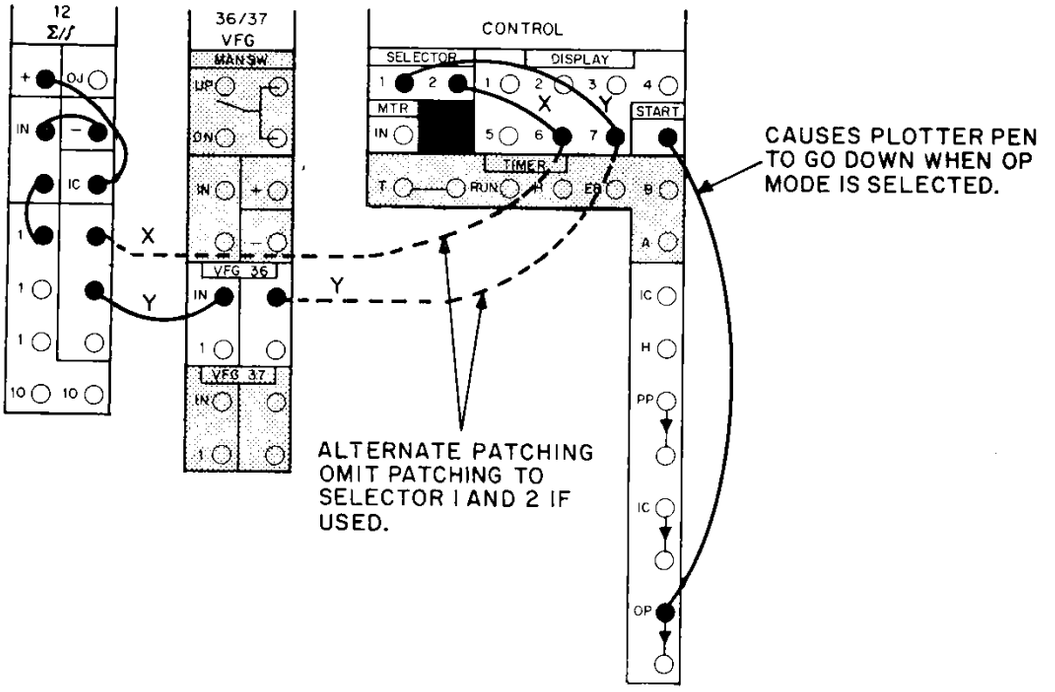
NOTES:
1. NUMBERS 0−10 REFER TO POT NUMBERS.
2. THE X1 POT ON VFG 37 MUST BE SET BEFORE THE Y10 POT ON VFG 36.

Figure 25.6. Sample 19 Segment Function

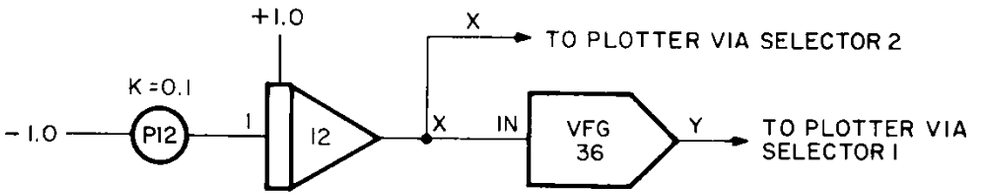
25.6 PLOTTING THE FUNCTION

Once the function is set, a continuous plot of output versus input should be made. This serves to verify the function and provides a record for problem documentation.

Figure 25.7 illustrates the typical circuit arrangement and patching requirements for plotting the function. The setup shown will provide a linear ramp input (±X) to the VFG that starts at −1.0 machine unit and integrates up to +1.0 machine unit. Using a ramp of this nature permits a continuous plot of any function over the entire range of the VFG. If the function does not require a −X value (X starts at zero and goes only in the positive direction), it is not necessary to integrate from −1.0 to +1.0 volts. In such a case, the IC input (+REF) is not required. Removing the initial condition voltage will provide a ramp that goes from 0.0 machine unit up to +1.0 machine unit. Note that the input potentiometer is set at 0.1 unit (1 volt). With a ten volt input and a normal time scale selected, the integration rate is 0.1 machine unit per second. Integration at a relatively slow rate is recommended to permit the operator to observe and inspect the function as it is plotted.



a. PATCHING



b. PROGRAM SYMBOL

NOTES:

1. ADDRESS INTEGRATOR USING SELECTOR −1 AND VFG WITH SELECTOR −2. IF ALTERNATE PATCHING IS USED, UNITS NEED NOT BE ADDRESSED.
2. IF THE FUNCTION DOES NOT REQUIRE −X VALUE, REMOVE IC INPUT FROM INTEGRATOR.

Figure 25.7. Typical Setup for Plotting A Function

Plot the function as follows:

1. Place the console in the Set Pot (SP) mode and set the coefficient pot.
2. Patch the VFG to the X-Y Recorder as illustrated in Figure 25.7 and prepare the recorder for the plot.
3. Ensure that the POINT selector(s) is at OFF and the SET/OPER switch(es) is at OPER. Do not disturb any other controls.
4. Select the normal time scale and place the computer in IC and then in OP.
5. Observe the plotter and place the computer in IC when the plot is completed.
6. Compare the plot to desired function and note any irregularities. If necessary, replot the function and make final trimming adjustments as required.

25.7 PATCHING THE VFG INTO A PROBLEM

Figure 25.8 illustrates the typical input/output patching requirements of the VFG for each mode of operation.

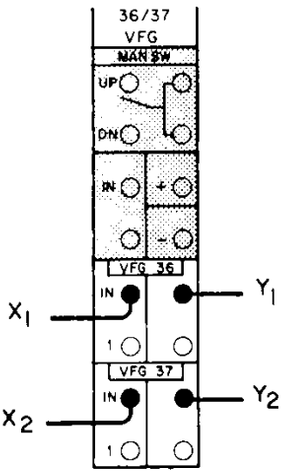
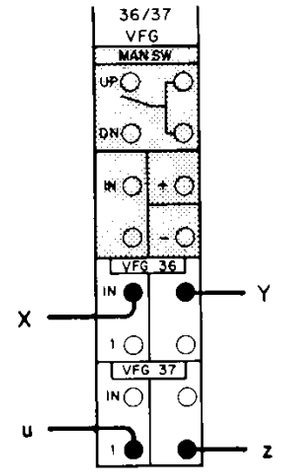
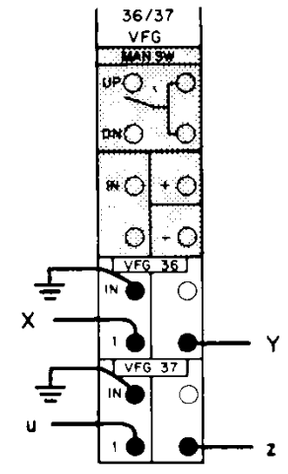
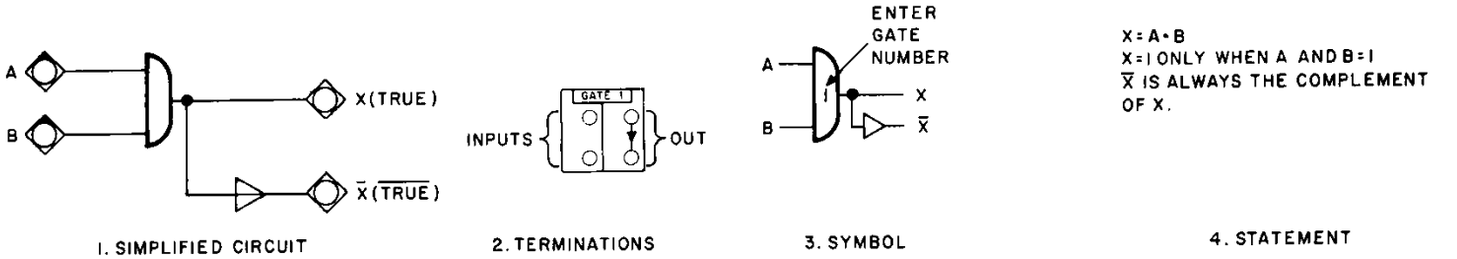
MODE	PROGRAM SYMBOL	PATCHING	OPERATIONAL SWITCH SETTINGS
11–PT	X_1 IN VFG 36 $Y_1 = fx_1$ X_2 IN VFG 37 $Y_2 = fx_2$		MODE: 11 PT SET–OPER (BOTH): OPER. POINT (BOTH): OFF GAIN (BOTH): A/R
20–PT	X IN VFG 36 Y=fx u I 37 z=–u		MODE: 20 PT SET–OPER (BOTH): OPER. POINT (BOTH): OFF GAIN: VFG 36 – A/R VFG 37 – 1 IF USED AS INVERTER
INV	X I 36 Y=–X u I 37 z=–u		MODE: INV SET–OPER (BOTH): OPER. POINT (BOTH): OFF GAIN (BOTH): 1 NOTE: VFG 36,37 IN PATCHED TO GROUND TO AVOID FALSE OVERLOAD

Figure 25.8. VFG: Basic Patching



a. TWO INPUT GATES

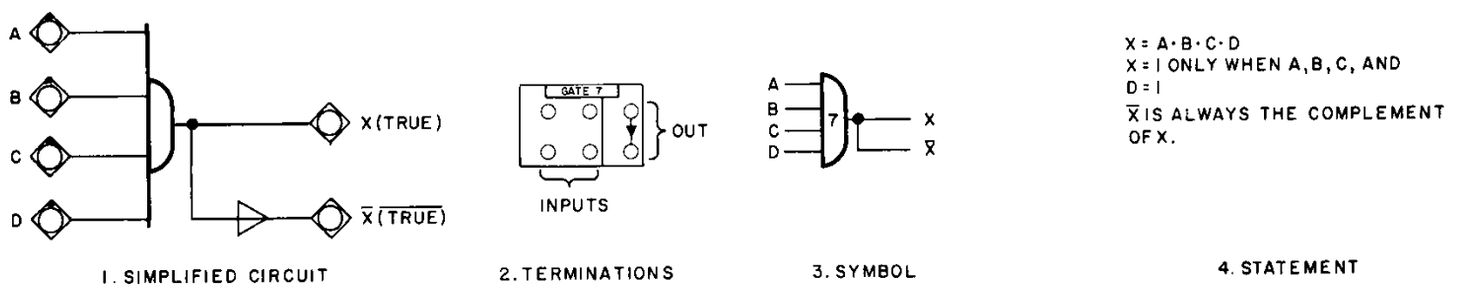


Figure 26.1. Gates: Simplified Circuits

26.1 LOCATION AND IDENTIFICATION

MiniAC logic expansion contains ten AND gates. Of these, six are two-input and four are four-input These gates are terminated at the top of the logic field as shown in Figure 2.2. The gates are identified by the panel markings GATE 1 through GATE 10. Gates designated GATE 1 through GATE 6 are two input gates. The remaining (GATE 7 – GATE 10), are four input gates. For patching ease and identification, the AND gate terminations are color coded. Input terminations are green and the output terminations (true and false) are orange. Each gate has a corresponding indicator on the logic control panel. When the true output of a gate is high, the corresponding indicator illuminates.

26.2 THE AND GATE

An AND gate accepts two or more logic inputs and produces a high output (logic one) when all inputs are simultaneously high. Figure 26.1 is a simplified diagram of the two and four input gates and shows the patching terminations, program symbol, and truth table for each. All gates in the MiniAC have built-in inverters and both the normal (true) and inverted (false) output are terminated at the patch panel. The gate input logic is arranged so that unused inputs are high (logic one). Therefore, an unpatched input enables the corresponding gate input. This feature permits a gate to be used with less than the maximum number of inputs. For example, a four input gate may be enabled using only two or three inputs. Extraneous patching is not required to ensure that unpatched inputs are high.

26.3 USING THE AND GATE

In addition to the ANDing function, the AND gate may be used in OR functions, as a logic inverter, or simply to obtain a constant logic level. Figure 26.2 illustrates the patching and programming symbol for each of these functions.

The output of a single gate can be used as a control input for several other gates as shown in Figure 26.3. Up to five gates can be patched in this manner. If a fan-out of greater than five is used, the logic level output of the control gate may not be sufficient to trigger or enable the controlled gates.

Gates can be patched in cascade for AND functions requiring more than four logic inputs. Figure 26.4 shows the recommended method of cascading gates.

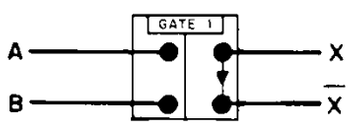
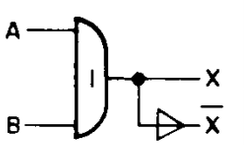
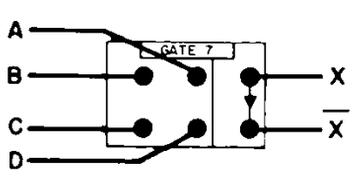
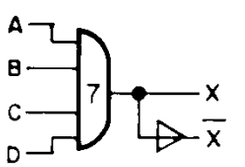
FUNCTION	PATCHING	PROGRAM SYMBOL	CONSIDERATIONS
AND			X=A·B X=1 ONLY WHEN A AND B=1 $\overline{X}$ IS ALWAYS THE COMPLEMENT OF X
			X=A·B·C·D X=1 ONLY WHEN A, B, C AND D=1 $\overline{X}$ IS ALWAYS THE COMPLEMENT OF X

Figure 26.2. Gate Patching

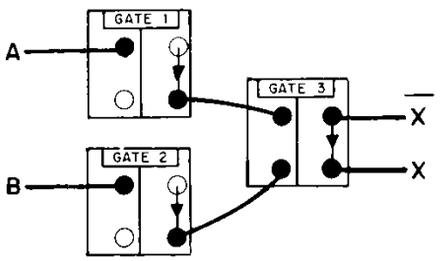
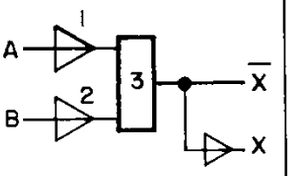
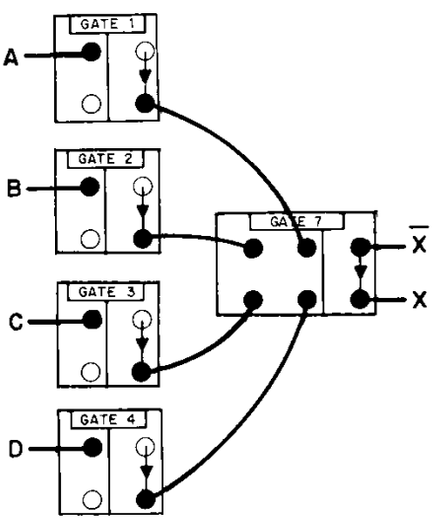
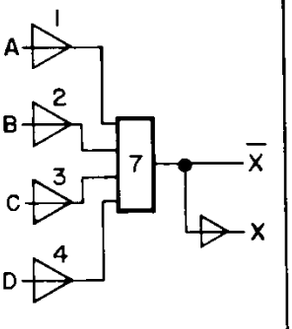
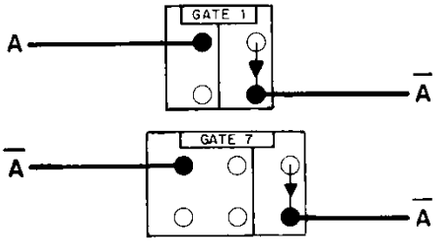
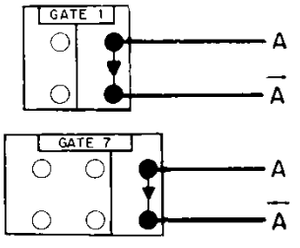
FUNCTION	PATCHING	PROGRAM SYMBOL	CONSIDERATIONS
OR			X=A+B X=1 WHEN A OR B=1 $\overline{X}$=0 WHEN A OR B=1
			X=A+B+C+D X=1 WHEN ANY INPUT=1 X=0 WHEN ANY INPUT=1
LOGIC INVERTER			$\overline{A}$=1 WHEN A=0; AND 0 WHEN A=1
CONSTANT LEVEL		N/A	A=1 AND $\overline{A}$=0 WHEN ALL INPUTS ARE UNPATCHED

Figure 26.2. Gate Patching (Cont)

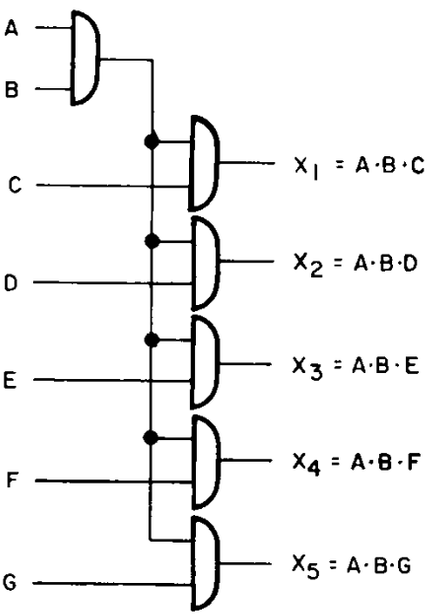


Figure 26.3. Typical AND Gate Fan-Out Circuit

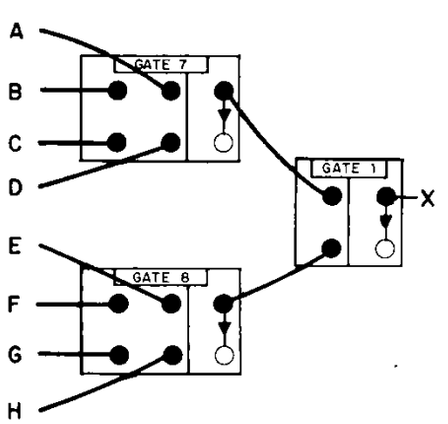
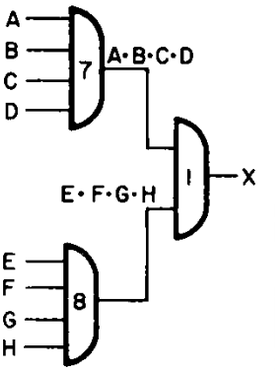
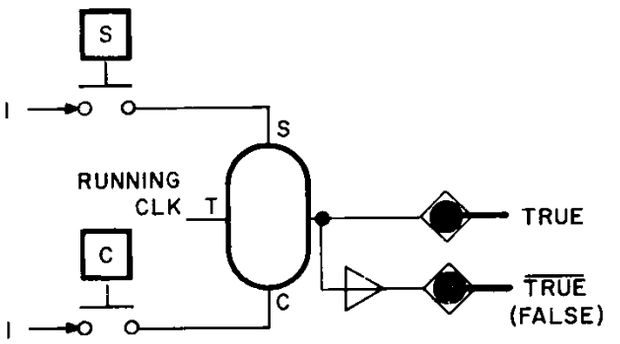
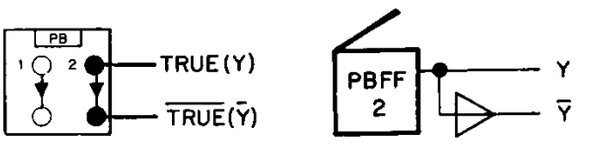
FUNCTION	PATCHING	PROGRAM SYMBOL	CONSIDERATIONS
8-INPUT AND			X=A·B·C·D·E·F·G·H X= 1 ONLY WHEN ALL INPUTS = 1

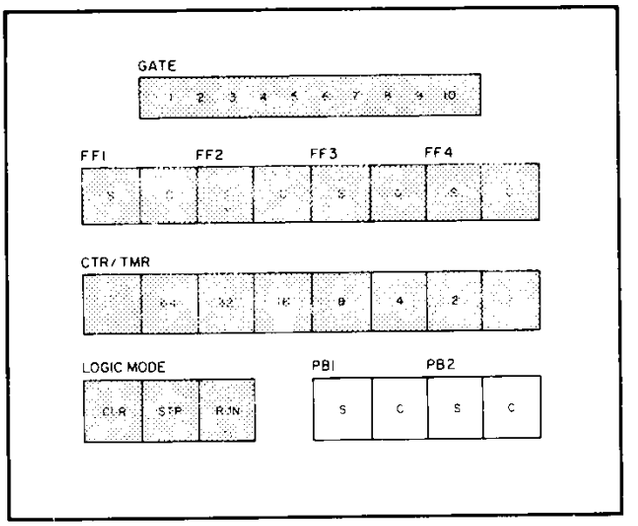
Figure 26.4. Typical Method of Cascading Gates



a. SIMPLIFIED CIRCUIT



b. PATCHING AND PROGRAM SYMBOL



LOGIC CONTROL PANEL

c. CONTROLS

Figure 27.1. The Pushbutton Flip-Flop

GENERAL PURPOSE AND PUSHBUTTON FLIP-FLOPS

27.1 LOCATION AND IDENTIFICATION

A fully expanded MiniAC contains two pushbutton and four general puspose flip-flops. The pushbutton flip-flops are controlled manually while the general purpose flip-flops can be controlled either by logic inputs (patching), or manually by pushbutton. As shown in Figure 2.2 the flip-flops are terminated in the logic field directly below the four input AND gates. Each general purpose flip-flop (GPFF) is identified by the patch panel alphanumeric designation FF1 – FF4. The pushbutton flip-flops are identified by the patch panel designation PB1 and 2. The manual controls for all flip-flops are located on the logic control panel and are identified by the panel designations FF1 – FF4 and PB1 and 2 that correspond to the terminations in the patch panel logic field.

27.2 THE PUSHBUTTON FLIP-FLOP

The pushbutton flip-flop (PBFF) is a clocked element that changes state on the first clock pulse after a manual set/reset command, and therefore produces synchronized outputs There are two pushbuttons (Figure 27.1) associated with each PBFF. Depressing the S pushbutton sets the flip-flop (true output goes high). The C pushbutton resets (clears) the flip-flop (true output goes low).

The terminations for PBFF1 and 2 are red and provide both the true and false outputs. When the flip-flop is set, the upper termination is high. In either state (S or C), the false output is the complement of the true output as designated by the inverter symbol.

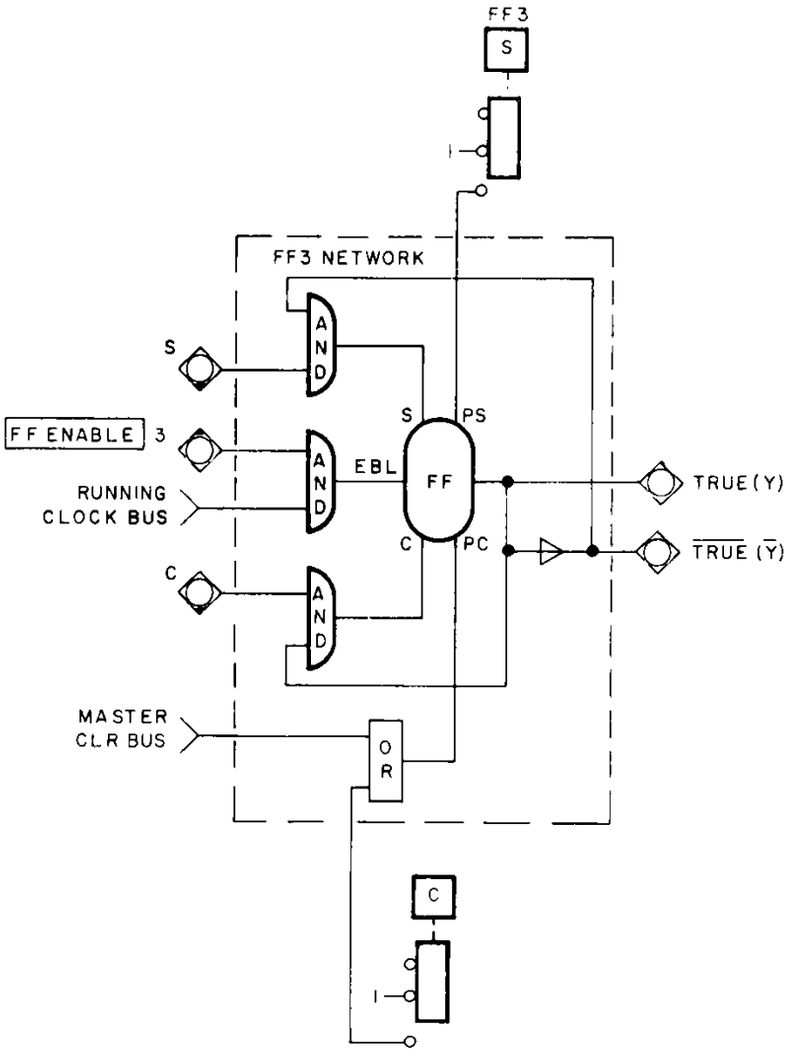
27.3 THE GENERAL PURPOSE FLIP-FLOP

27.3.1 GENERAL OPERATION

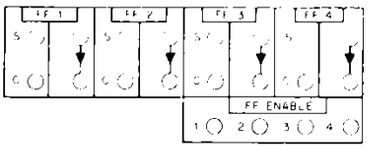
The GPFF. Figure 27.2, changes state in response to patched logic inputs, or manual inputs (pushbuttons on the logic control panel). When under patch panel control, the flip-flop changes state on the first clock pulse after the set/reset command and is therefore synchronized with the MiniAC logic clock. To obtain complete patch panel set/reset control of the GPFF, two logic inputs (set and clear) must be used. These can be obtained from separate sources, or simply by using the complement of a single source. When the flip-flop is set or reset manually, the change in state is independent of clock and is entirely asynchronous.

27.3.2 THE PUSHBUTTON CONTROLS

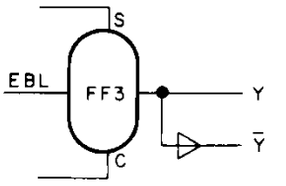
There are two momentary pushbuttons associated with each GPFF. The S pushbutton is used to manually set the GPFF, while the C pushbutton (when depressed) resets or clears the flip-flop. Note that if logic inputs are patched to the flip-flop, inadvertent operation of these pushbuttons during a program run may interfere with the problem and may cause an erroneous solution. The S pushbutton is equipped with a lamp. Whenever a GPFF is set, the corresponding S indicator lights. When reset, the lamp is extinguished.



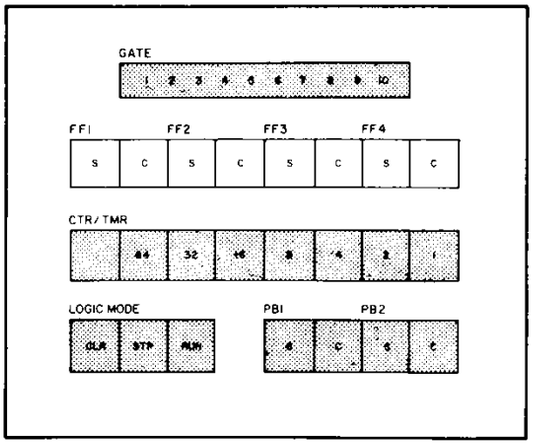
a. SIMPLIFIED CIRCUIT



b. PATCHING TERMINATIONS



c. PROGRAM SYMBOL



LOGIC CONTROL PANEL

d. MANUAL CONTROLS

Figure 27.2. The General Purpose Flip-Flop

27.3.3 THE PATCH PANEL TERMINATIONS

Each GPFF has three input and two output terminations in the patch panel logic field. The output terminations provide a true and a false output. When the flip-flop is set, the true output (upper termination) is high, and when reset it is low. In either state, the false output is the complement of the true output as designated by the logic inverter symbol.

The set and clear inputs (S and C) respond to a high to correspondingly set or clear the flip-flop. Note that if both inputs are low simultaneously, the flip-flop will remain in its previous state. With both inputs patched high, the flip-flop can be made to toggle. The four terminations designated FF ENABLE (1-4) provide the third control input to the corresponding numbered flip-flop. These terminations are normally high. Patching a low into one of these terminations prevents the corresponding flip-flop from changing state. This latching feature is especially useful for monitoring an event and flagging that event the first time it occurs, or for sampling logic inputs.

27.4 BASIC CIRCUITS USING THE GPFF

27.4.1 GENERAL USE

Basically, the GPFF is a storage element and can be used to perform numerous functions. The GPFF can be easily interconnected with other flip-flops to form a shift register and a parallel jam load register. This device can also be used to generate a delay. The following paragraphs describe some of these basic uses and provide patching and programming information. Figure 27.3 illustrates the basic patching requirements of the GPFF when used both manually (pushbutton flip-flop) and under patch panel control.

27.4.2 USED TO GENERATE A ONE CLOCK DELAY

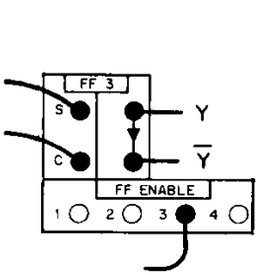
As previously stated, the general purpose flip-flop is clocked, and does not change state until the first clock pulse after the set or clear signal. Therefore, the flip-flop output switches in synchronism with the logic clock. If used with a synchronous input, the GPFF can be patched to generate a one clock delay as shown in Figure 27.4a. Note that the clear (C) input is the complement of the set (S) input. If the signal source has complementary outputs, the logic inverter.(Gate 2) is not necessary.

27.4.3 USED AS A DIFFERENTIATOR

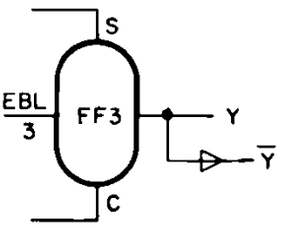
When used with a synchronous input (Figure 27.4b) the GPFF can be patched as a leading edge differentiator. In this circuit, the GPFF responds to input transitions from the 0 state to the 1 state to produce an output one clock period in duration. The only requirement for input X is that it must be synchronous and or more than one clock period in duration.

27.4.4 USING THE LATCHING FUNCTION

A low control input at FF ENABLE (EBL) prevents the flip-flop from changing state. This feature permits the GPFF to be used as a sampling device, a permanent storage device, and as a toggle flip-flop. Figure 27.4 (Parts c, d and e) illustrates three uses of the latching function. When patched as a sampling device, note that the flip-flop only changes state when the EBL input goes high.



PATCHING

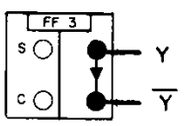


PROGRAM SYMBOL

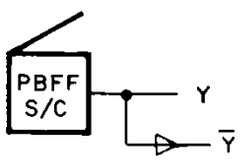
Input Conditions and Output States

Input Logic			Output State		
EBL 3	S	C	Y	$\overline{Y}$	Description
1	1	0	1	0	When S goes high, switches to set state on next clock pulse.
1	0	1	0	1	When C goes high, switches to reset state on next clock pulse.
1	1	1	If 1 goes to 0 If 0 goes to 1	If 0 goes to 1 If 1 goes to 0	When S and C go high, toggles to opposite state on next clock pulse providing S and C inputs are held high. Will continue to toggle on succeeding clocks if EBL 3 remains HIGH.
1	0	0	1 or 0	0 or 1	Flip-flop remains in previous state until opposite input (S or C) goes high.
0	X	X	1 or 0	0 or 1	Latches in last state attained. S and C inputs have no control. (X = Either 1 or 0 or don't care)

a. PATCH PANEL CONTROL



PATCHING



PROGRAM SYMBOL

Y=1 AND $\overline{Y}$=0 WHEN S IS DEPRESSED

Y=0 AND $\overline{Y}$=1 WHEN C IS DEPRESSED

b. MANUAL OPERATION

Figure 27.3. GPFF: Basic Patching and Operating Information

When patched as a permanent storage device, the flip-flop latches up on its own output and cannot change state regardless of the X input. This is useful (for example) when monitoring a change in state of a comparator. When the comparator output goes high (sum of its inputs are positive), the flip-flop sets and remains set regardless of succeeding changes in comparator output.

When patched as a toggle flip, the flip-flop toggles to the opposite state on the clock pulse after the EBL input goes low.

NOTE:
MOST OF THE FOLLOWING FUNCTIONS USE GATE 2 AS A LOGIC INVERTER FOR COMPLEMENTARY STEERING. IF THE SOURCE SIGNAL HAS COMPLEMENTARY OUTPUTS, THE LOGIC INVERTER IS NOT REQUIRED.

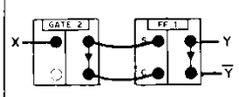
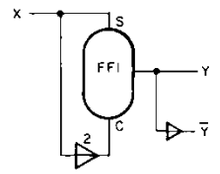
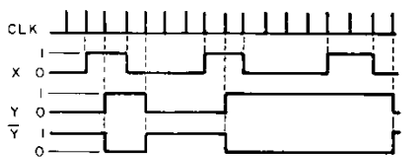
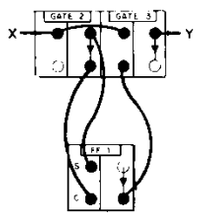
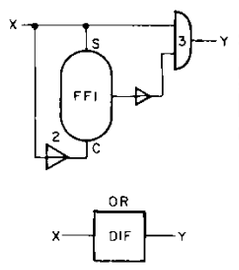
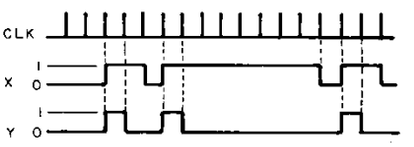
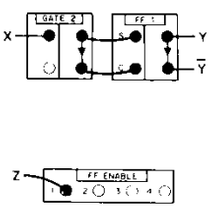
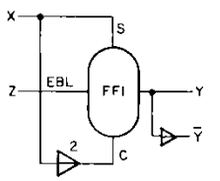
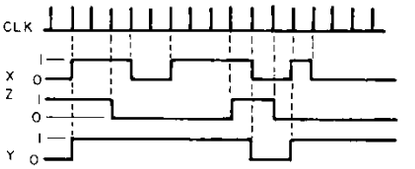
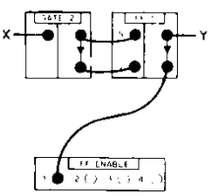
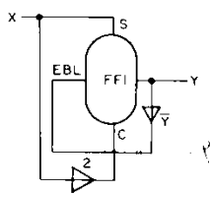
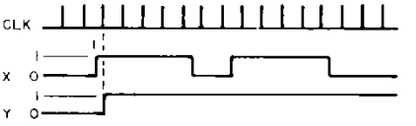
FUNCTION	PATCHING	PROGRAM SYMBOL	TIMING
a. ONE CLOCK DELAY			 X_{in} MUST BE SYNCHRONIZED WITH THE CLOCK AND BE ONE CLOCK PERIOD OR MORE. Y OUT IS DELAYED ONE CLOCK PERIOD. $\overline{Y}$ OUT IS THE COMPLEMENT OF Y
b. DIFFERENTRATOR			 X_{in} SHOULD BE SYNCHRONIZED WITH CLOCK AND IS NORMALLY LONGER THAN ONE CLOCK PERIOD.
c. SAMPLING			 REGUARDLESS OF INPUT X, FF DOES NOT CHANGE STATE UNLESS INPUT Z IS HIGH.
d. PERMANENT STORAGE			 WHEN X_{in} GOES HIGH, $\overline{Y}$ GOES LOW ON THE FOLLOWING CLOCK AND LATCHES THE FF IN THE SET STATE: Y=1.

Figure 27.4. GPFF: Patching and Timing

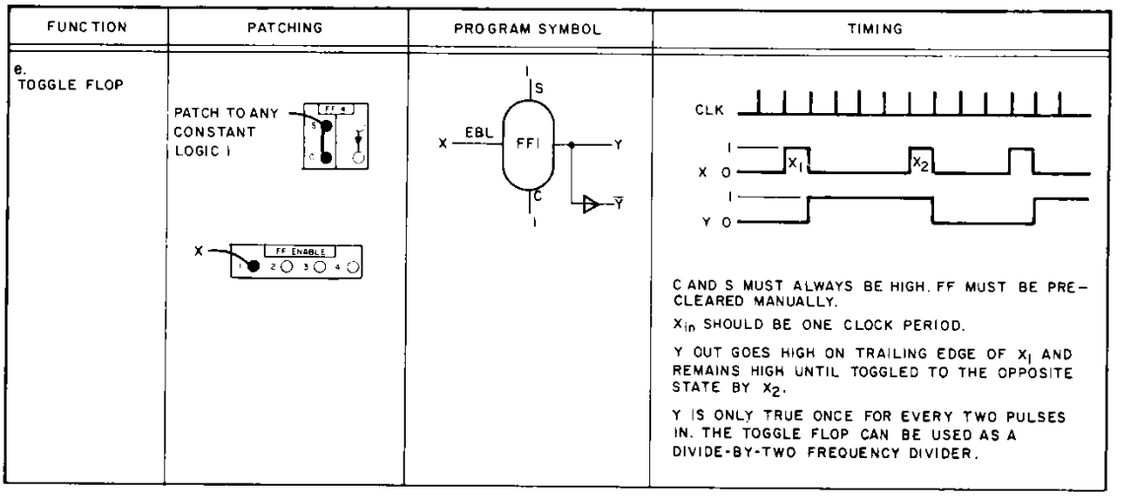


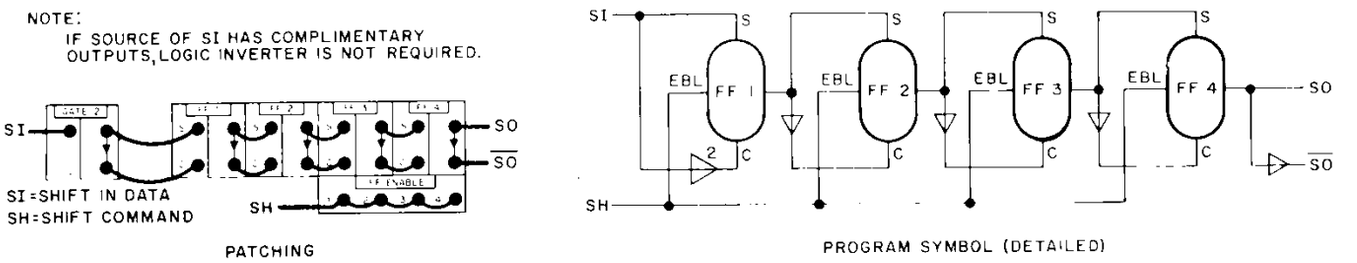
Figure 27.4. GPFF: Patching and Timing (Cont)

27.4.5 USED AS A FLIP-FLOP REGISTER

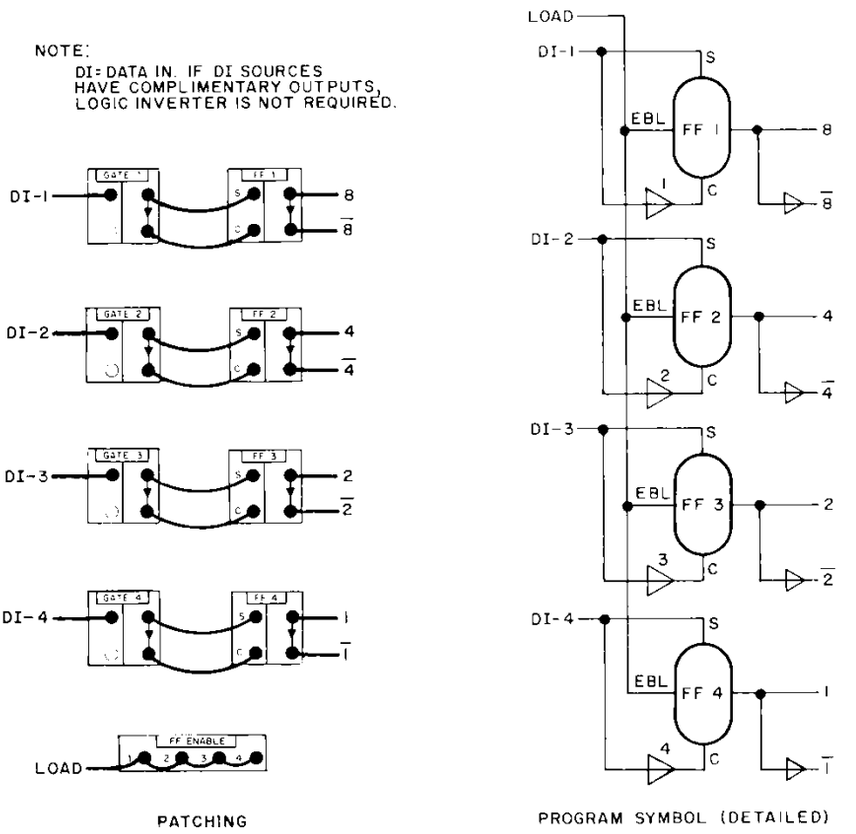
Up to four general purpose flip-flops may be patched together to form a flip-flop register. Figure 27.5 illustrates the patching configuration and detailed programming symbol for a four-bit shift and four-bit jam load register.

Basically, the shift register (Figure 27.5a) consists of four logic samplers connected in cascade. The SH input (shift) determines when the register will shift and is patched to each FF ENABLE termination (SH must be high to enable the GPFF). The SI input (shift in) is the data input and can only change the state of FF1 when SH is high. If the data source has complementary outputs, the logic inverter (GATE 2) is not necessary.

Parallel register patching is illustrated in Figure 27.5b. Note that each flip-flop has its own data input (DI-1 to DI-4). To obtain parallel data transfer, the FF ENABLE control input of each flip-flop is paralleled. When the LOAD signal is received (LOAD = 1), all flip-flops are enabled and the data present at the DI inputs are loaded simultaneously (jam loaded). If the data sources have complementary outputs, the logic inverter at each input is not required.

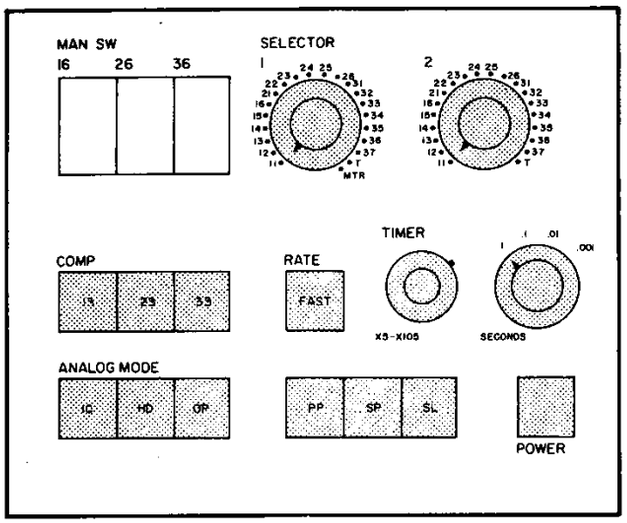


a. SHIFT REGISTER

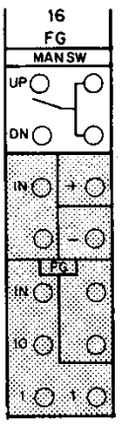


b. JAM LOAD REGISTER

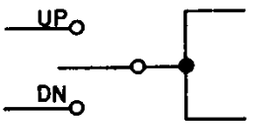
Figure 27.5. Using the GPFF to Form a Register



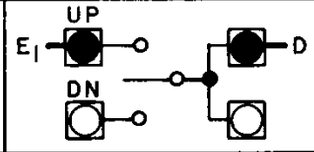
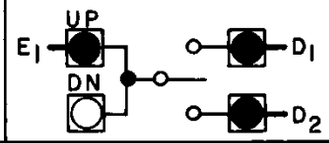
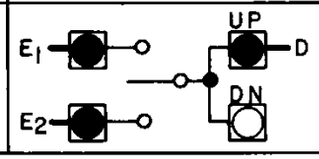
a. CONTROLS



b. TERMINATIONS



c. PROGRAM SYMBOL

FUNCTION	PATCHED CIRCUIT	DESCRIPTION		
		UP	CENTER	DN
FEED A SINGLE DESTINATION (ON-OFF-OFF)		D=E_1 (ON)	D=0 (OFF)	D=0 (OFF)
FEED A SINGLE INPUT TO ONE OF TWO DESTINATIONS (ON-OFF-ON)		DI=E_1 (DI ON)	DI AND D_2=0 (BOTH OFF)	D_2=E_1
SELECT ONE OF TWO INPUTS (ON-OFF-ON)		D=E_1	D=0 (OFF)	D=E_2

d. BASIC SWITCH CIRCUITS

Figure 28.1. Manual Switch: Patching and Operating Information

28.1 LOCATION AND IDENTIFICATION

A fully expanded MiniAC contains three manual switches, three D/A switches, and three A/D comparators. One of each of these devices is terminated in each analog field of the patch panel (Figure 2.2). Individual logic controls for each D/A switch, and the logic output of comparators are terminated in the control field. Each device is identified by its module address (Chapter 2) and by the individual panel markings as outlined below.

Device	Designation	Module Location
Manual Switch	MAN SW	16, 26, 36
D/A Switch	SW	15, 25, 35
Comparator	COMP	13, 23, 33

28.2 THE MANUAL SWITCH

The manual switch (Figure 28.1) is a single pole double throw (center off) device with two terminations at the single pole (wiper) end. Three switches can be patched into a problem and manually set to provide nearly any desired analog switching function These switches are especially useful during problem checkout. For example, when checking integrators initial values a constant can be patched to an integrator (through a MAN SW) that does not have an IC during the problem solution. Then after all initial values are checked, the test IC input can be switched from that integrator without disturbing existing patching.

Physically, the manual switches are located on the analog control panel. When the upper portion of the switch is fully depressed the switch is in the UP position as designated on the patch panel.

Fully depressing the lower portion of the switch corresponds to the DN position. When neither end of the switch is depressed (center position) the switch is off.

28.3 THE D/A SWITCH

28.3.1 GENERAL OPERATION

The D/A switch is a high speed solid state element that allows an analog input to be turned on and off in response to a logic control signal. A logic ONE at the control input turns the switch on; a logic ZERO turns it off.

Basically, the unit consists of a resistor in series with an electronic switch. The resistance value is chosen so that the total series resistance is 10k ohms when the switch is conducting. When the switch is turned off, the output end of the resistor is grounded. Therefore, the effective input resistance is 10k ohms regardless of the switch state (open or closed). This feature prevents loading errors when the analog input is fed by a high impedance source.

28.3.2 USING THE D/A SWITCH

The overall electrical behavior of the D/A switch and patching information is illustrated in Figure 28.2. For clarity, an equivalent relay circuit is shown. Note that the control input (ON) is normally high and the switch is conducting. When ON is patched low, the switch is off.

The D/A switch can only be used with amplifiers whose summing junction is available at the patch panel. The SW output is labeled SJ to indicate normal patching is to SJ of a Σ/TS, Σ/HG, or OJ of a Σ/∫. When patched to the summing junction (SJ) of a Σ/TS unit, the standard feedback resistor is 100k ohms. Therefore, the switch acts as a gain of ten input. The feedback resistance of a Σ/∫ is one megohm (when operated as a summer); therefore when the SJ of the switch is patched to the summing junction (OJ) of the Σ/∫, the switch acts as a gain 100 input. Note that the D/A switch is not normally used with a Σ/∫ as these devices have internal D/A switches.

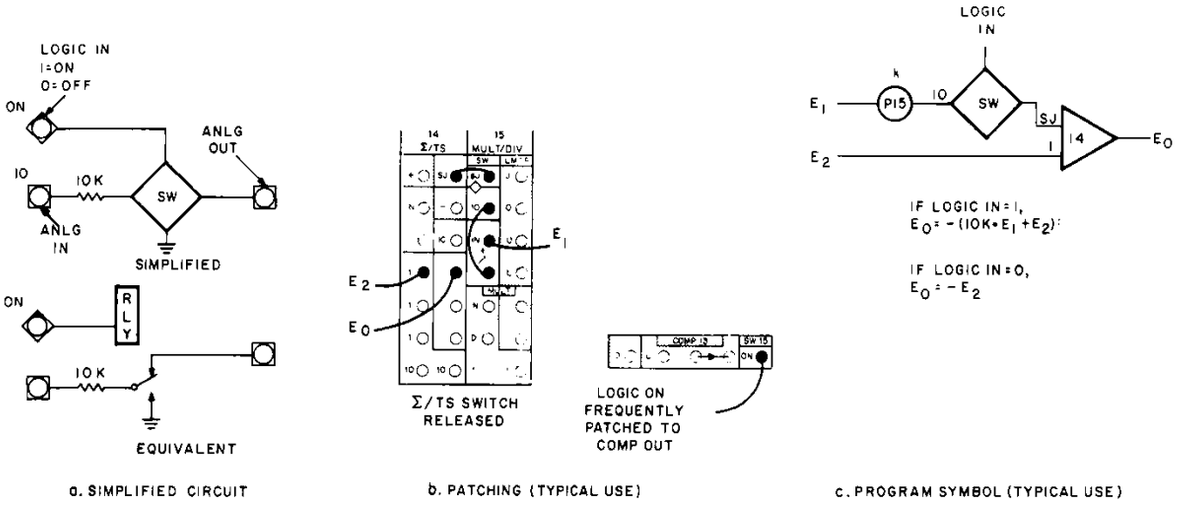


Figure 28.2. D/A Switch: Patching and Operating Information

28.4 THE A/D COMPARATOR

28.4.1 GENERAL OPERATION

The A/D comparator is the basic analog-to-digital interface component. The comparator (Figure 28.3) provides complementary logic output signals as the function of the polarity of the sum of two analog input signals. When the sum of the inputs become positive, the true output goes high. Conversely, when the sum is negative, the true output goes low. If only a single analog input is used, the comparator output will change state each time zero crossover occurs. If used in this configuration, the unused input should be grounded.

Basically, the comparator consists of an operational amplifier, an analog-to-logic translator, steering logic and a clocked output flip-flop. With this arrangement, the comparator switches state in synchronism with system clock. The steering logic is arranged so that the comparator will latch in either existing state, when a logic one (high) is applied to the L termination. Each comparator is equipped with a momentary pushbutton switch (located on the control panel). During normal operation, depressing this switch forces the comparator to its opposite state. When the switch is released, the comparator will return to the previous state. The feature is useful during problem checkout. Note that each COMP switch is also an indicator and illuminates whenever the comparator true is high.

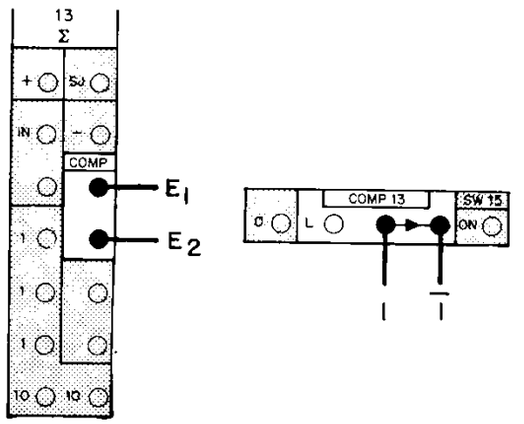
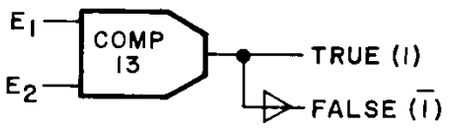
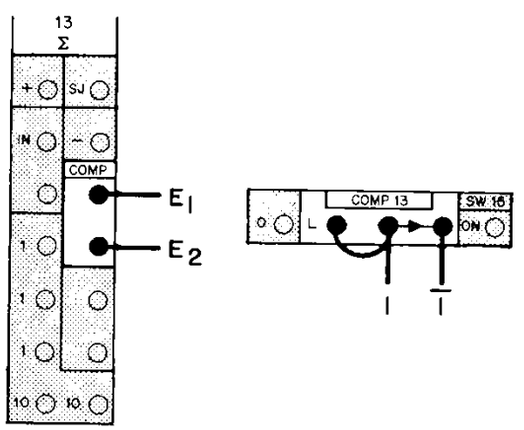
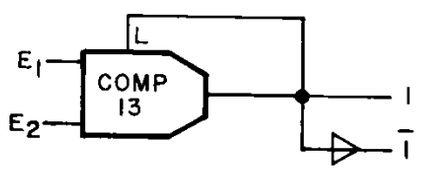
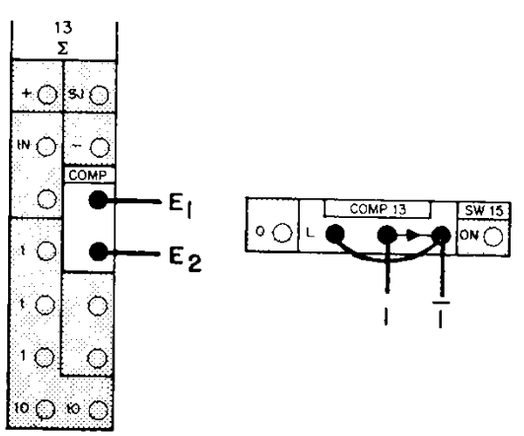
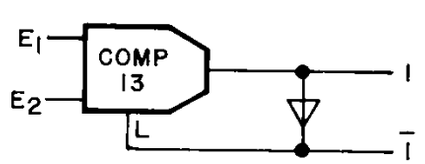
CONFIGURATION	PATCHING	PROGRAM SYMBOL
CONTINUOS OPERATION		 WHEN $E_1 + E_2$ =POSITIVE VALUE, THE TRUE OUTPUT IS A 1 WHEN $E_1 + E_2$ =A NEGATIVE VALUE, THE TRUE OUTPUT IS A 0
POSITIVE LATCH		 WHEN $E_1 + E_2$ IS A POSITIVE VALUE (TRUE OUTPUT=1), COMPARATOR LATCHES
NEGATIVE LATCH		 WHEN $E_1 + E_2$ IS A NEGATIVE VALUE (FALSE OUTPUT=1) COMPARATOR LATCHES

Figure 28.4. Comparator Patching

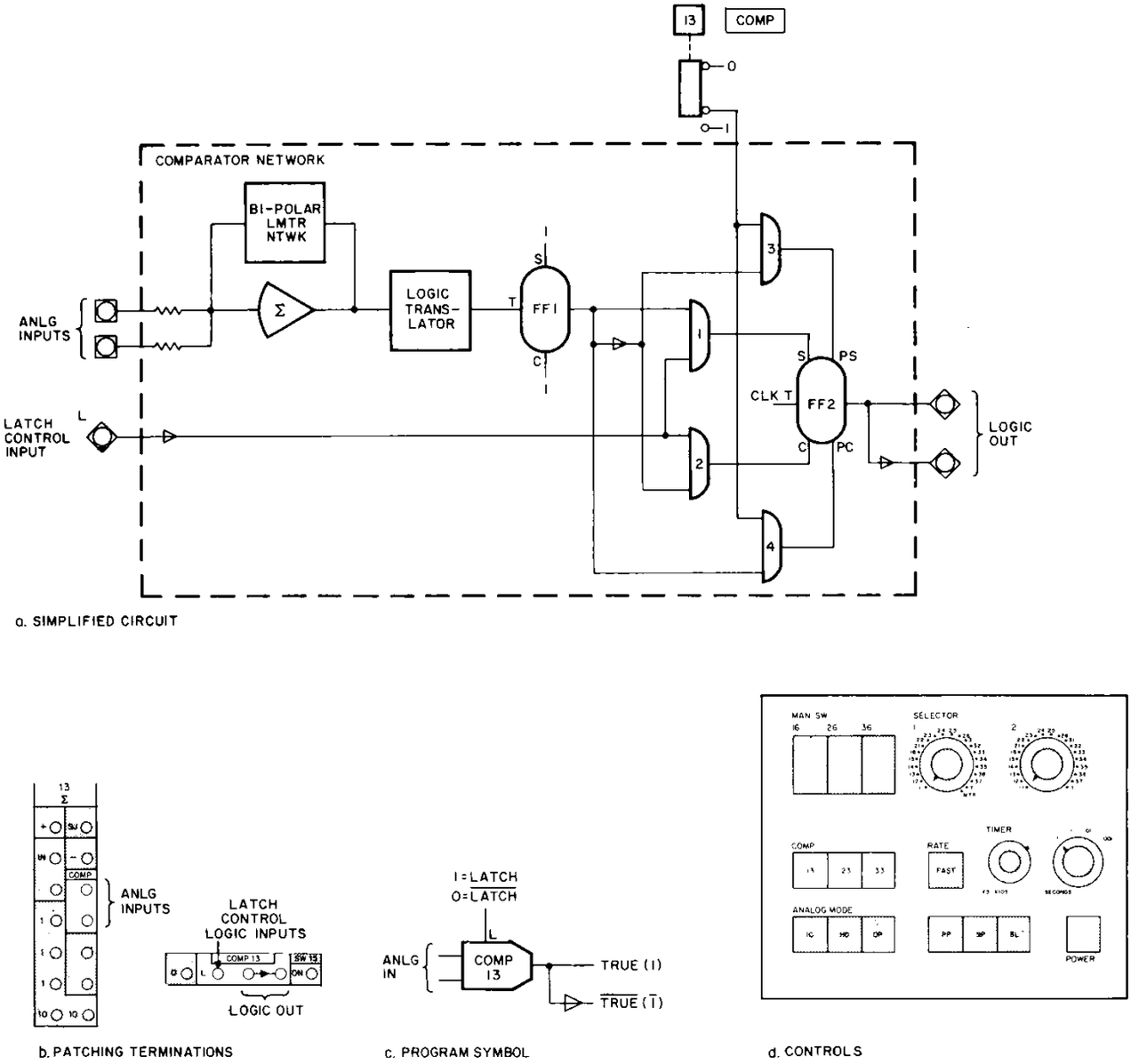
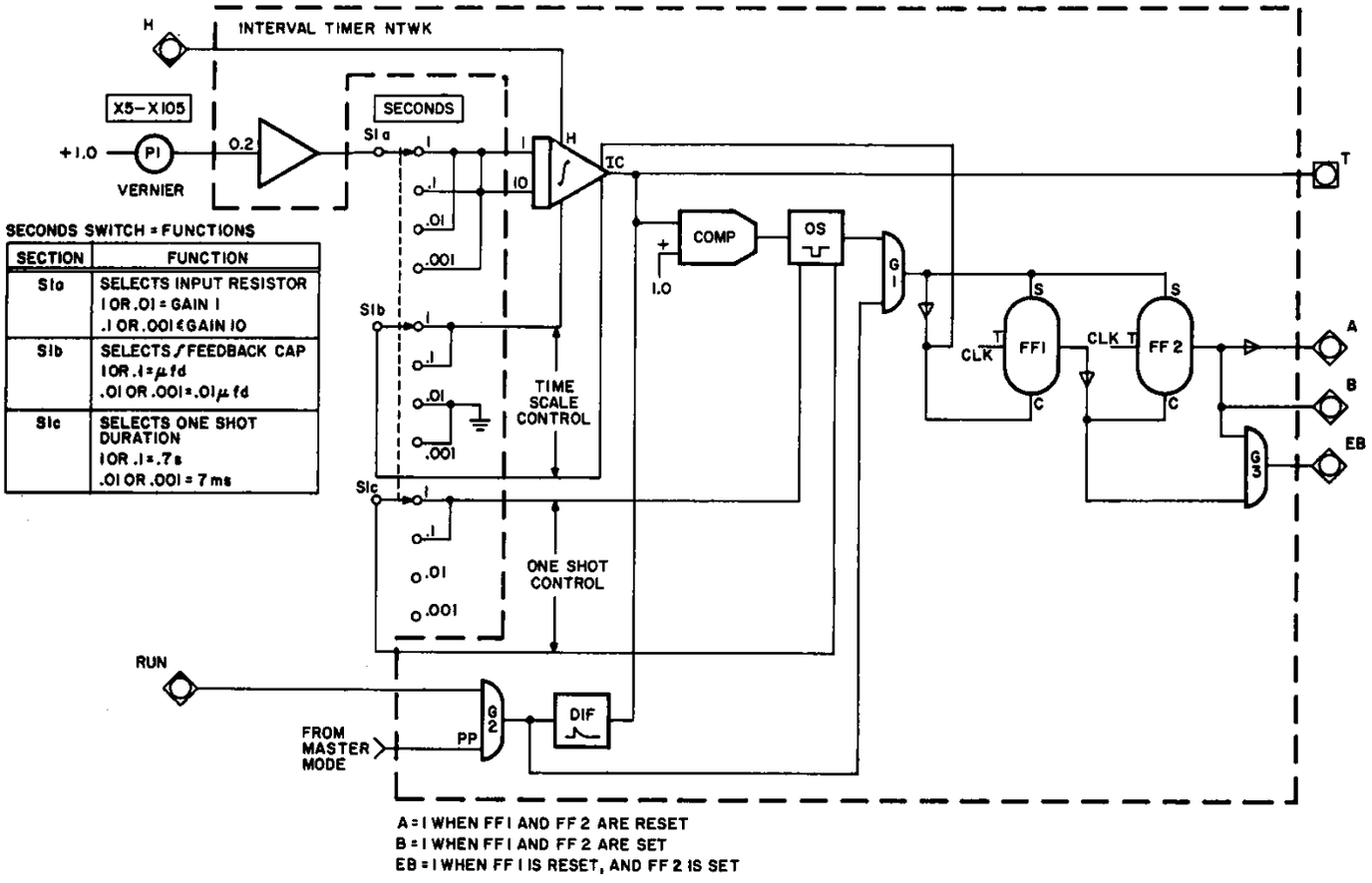


Figure 28.3. The Comparator

28.4.2 BASIC COMPARATOR PATCHING

Figure 28.4 illustrates the basic patching requirements and program symbol for general use of the comparator. Note that the comparator output is synchronous and can be used to drive any logic element, or a D/A switch as required by the program.

When either comparator output (true or false) is patched to its own L termination, the comparator will latch up whenever that output goes high, and cannot change state regardless of analog input conditions. When this self latching function is used, the comparator can be released from latch by depressing the associated COMP pushbutton. This switches the comparator to the opposite state causing the signal at the L termination to go low. However, if the sum of the inputs still provide the same or equivalent conditions to those that caused latch up, the comparator will latch again when the switch is released.

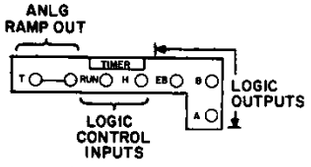


SECONDS SWITCH = FUNCTIONS

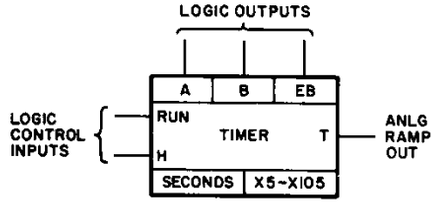
SECTION	FUNCTION
SIa	SELECTS INPUT RESISTOR 1 OR .01 = GAIN 1 .1 OR .001 = GAIN 10
SIb	SELECTS ∫ FEEDBACK CAP 1 OR .1 = μfd .01 OR .001 = .01 μfd
SIc	SELECTS ONE SHOT DURATION 1 OR .1 = .7 s .01 OR .001 = 7 ms

A = 1 WHEN FF1 AND FF2 ARE RESET
B = 1 WHEN FF1 AND FF2 ARE SET
EB = 1 WHEN FF1 IS RESET, AND FF2 IS SET

a. SIMPLIFIED CIRCUIT

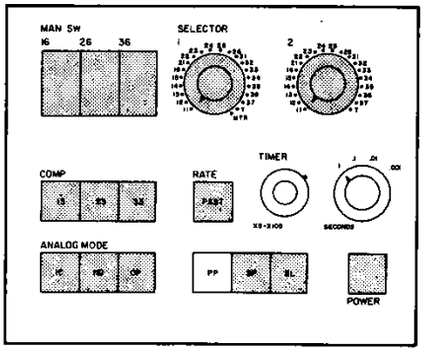


b. PATCHING TERMINATIONS



NOTE:
UNUSED INPUTS AND OUTPUTS DO NOT HAVE TO BE SHOWN ON SYMBOL.

c. PROGRAM SYMBOL



ANALOG CONTROL PANEL

d. CONTROLS

Figure 29.1. The Interval Timer

29.1 LOCATION AND IDENTIFICATION

In addition to the logic clock and the downcounted clock (PULSES – 10^5, 10^4, 10^3, and 10^2,) described in Chapter 15, the fully expanded MiniAC is equipped with three timing devices. These include an interval timer, a counter/timer, and a monostable multivibrator (one-shot).

The interval timer is primarily used for controlling the computer analog modes during repetitive operation (Chapter 14). For patching convenience, this device is terminated in the patch panel control field (Figure 2.2) in close proximity to the Master Mode and local control terminations. Setup controls for the interval timer are located on the analog control panel. The patching termination and controls for this device are identified by the TIMER designation on both panels.

The counter/timer is primarily used for counting logic events. This logic device (designated CTR/TMR) is terminated in the logic field with its manual controls on the logic panel. Since the CTR/TMR is frequently used in conjunction with the downcounted clock, the terminations (designated CTR/TMR) are located in the bottom left-hand corner of the logic field adjacent to the PULSES output terminations.

The one shot is terminated in the lower right-hand corner of the logic field. There terminations are identified by the patch panel MONO designation.

29.2 THE INTERVAL TIMER

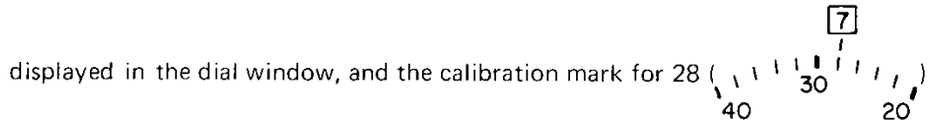
29.2.1 OVERALL OPERATION

Basically, the interval timer (Figure 29.1) is a logic device that produces two timed intervals (designated A and B) to provide automatic logic control of the computer IC and OP modes. The A output corresponds to the IC mode and the B output corresponds to the OP mode. Only one of these outputs is ever true at any one given time. The A interval is fixed at 6 milliseconds in high speed operations and 0.6 second for low speed operations. The B interval is fully adjustable from 5 milliseconds up to 105 seconds. A third logic output (designated EB) is a one microsecond pulse that corresponds to the end of interval B. This pulse appears one microsecond before interval A is dropped and disappears when interval B is raised.

As shown in Figure 29.1a, the timer consists of a constant source input amplifier, an integrator network (used as a ramp generator), a comparator, and an array of logic elements. The switch (designated SECONDS) controls the rate of integration, to select the time range of the B interval. This switch also selects either of the two time durations for the A interval. The timer circuits are arranged so that when the comparator output goes to a logic one, a one-shot is fired, the ramp generator goes to IC mode, and the A output goes high (FF1 and FF2 reset). At the end of the one shot (7 milliseconds or 0.7 second), the ramp generator is placed in operate and both flip-flops set. (A drops and B is raised.) Interval B remains high while the integrator times out. When the integrator output reaches 1.0 machine units, the comparator output goes to a logic zero, to place the integrator in IC, and resets FF1. This raises EB, and on the succeeding clock, FF2 resets. This drops B and EB, and raises A.

The timer continues cycling in this manner as long as patch panel mode control is selected and the RUN and H control inputs (Paragraph 29.2.3) are unpatched.

29.2.2 THE MANUAL CONTROLS

The duration of the timer A and B intervals are manually selected using the TIMER-SECONDS switch and the TIMER-X5-X105 potentiometer. The four position SECONDS switch selects the duration of the A interval (about 0.7 second or 7 milliseconds) and the time range of the B interval as outlined in Table 29.1. The X5-X105 control is a ten-turn precision potentiometer with a calibrated dial and locking lever. The SECONDS switch setting (1, .1, .01 or .001) multiplied by the X5-X105 pot setting determines the exact duration of interval B. For example, if the desired B interval is 0.728 seconds, simply set the SECONDS switch at 0.1. Then, release the locking lever (push to left) and set the pot so that a 7 is displayed in the dial window, and the calibration mark for 28 (

) is aligned with the dial scribe mark. When the pot is set, lock the dial in position (push lever to right) to prevent inadvertant misadjustment.

Table 29.1. Timer Ranges Versus SECONDS Switch Setting

SECONDS Switch Setting	Interval		Note
	A (Approx)	B	
1	0.7 S	5 S to 105 S	SECONDS switch setting multiplied by X5-X105 pot setting determines exact duration of interval B
.1	0.7 S	0.5 S to 10.5 S	
.01	7 ms	50 ms to 1.05 S	
.001	7 ms	5 ms to 0.105S	

29.2.3 THE TIMER PATCH PANEL TERMINATIONS

As shown in Figure 29.1, the timer has two input terminations (RUN and H) and five output terminations (A, B, EB, and two designated T) in the patch panel control field. The RUN and H inputs provide logic control of the timer. The A and B terminations provide logic outputs that correspond to the A and B intervals. The termination designated EB goes high for one microsecond just prior to the end of the B interval. T is an analog ramp output that is coincident with the B interval. The function and use of each TIMER termination is described in Table 29.2.

Table 29.2. TIMER Patch Panel Terminations

Termination	Function and Use
T	Linear Ramp Output: Corresponds to interval B. Starts at 0.0 Machine Unit and reaches 1.0 Machine Unit one microsecond before B ends: Generally used as time base for XY plotter (User's Guide, Chapter 9) or as external sweep for the scope. Can also be used as input to any analog device terminated at the patch panel. Duration of T is determined by TIMER control settings. Never patch T to a logic device.

(Cont)

Table 29.2 TIMER Patch Panel Terminations (Cont)

Termination	Function and Use
RUN	Logic Control Input: Normally high. If unpatched, timer starts when ANALOG MODE-PP pushbutton is depressed. When patched low forces timer to A interval. Timer will not start Timing out in A interval until RUN goes high. Can be used to prevent timer from starting when PP is depressed and/or to force timer into A interval after any given logic event. RUN need only be low for one microsecond to force timer to A interval.
H	Logic Control Input: Normally low. When patched, a high causes timer to hold linear ramp (T). Can be used to extend OP period for problem analysis. When H goes low, the timer resumes operation from the point in time at which it was interrupted.
EB	1 Microsecond Logic Output: EB (end of B) goes high one microsecond before B drops. Normally used to trigger MONO during automatic three-mode (IC, OP and H) operations. Can also be used for any logic timing control such as carry-in (CI) for CTR/TMR.
A & B	Timed Interval Logic Outputs: A and B go high in turn for a period determined by the TIMER controls (Table 29.1). A is high when the ramp is in the reset state. B is high when the ramp is in operate or hold. A is normally used for patch panel control of the OP and IC modes. Both A and B can be used to control any logic function.

29.2.4 USING THE TIMER

The timer is frequently used as a rep-op timer for automatic iterative and repetitive operation of the computer. When used as a rep-op timer, the goal is to simply cycle the computer between the analog IC and OP modes. To accomplish this, the A output must be patched to the Master Mode IC input termination and the ANALOG MODE-PP pushbutton must be depressed as described in Chapter 14. The general procedure to be followed when using the timer for mode control is as follows:

1. Patch the timer to Master Mode terminations as required by the overall program (Chapter 14 and Figure 29.2).
2. Select the desired computer time scale (Chapter 14) Normal if SECONDS switch is at position 1 or .1. FAST if switch is at .01 or .001.
3. Depress ANALOG MODE-PP pushbutton. This transfers mode control to the patch panel, and if RUN is unpatched, starts the timer.

In the preceding procedure, consideration was not given to the use of the RUN and H control inputs. The basic patching and timing requirements for typical use of the RUN and H control are illustrated in Figures 29.3 and 29.4 respectively. With additional patching, the timer may be used in conjunction with the MONO (Paragrapn 29.4) to control the three analog operating modes (IC, OP and H). These patching requirements are described in Chapter 14. All other operations are the same as described in the preceding paragraph.

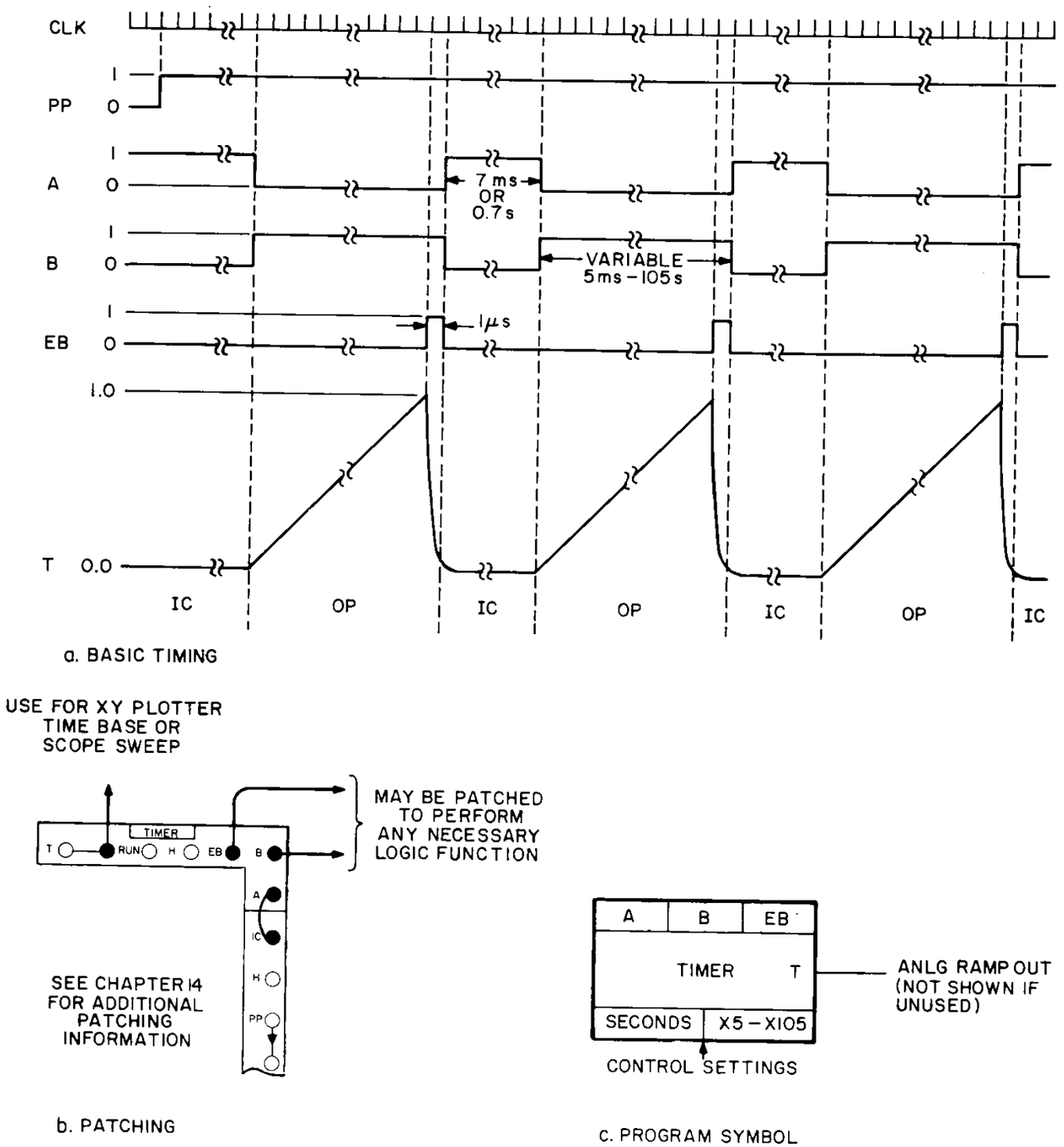
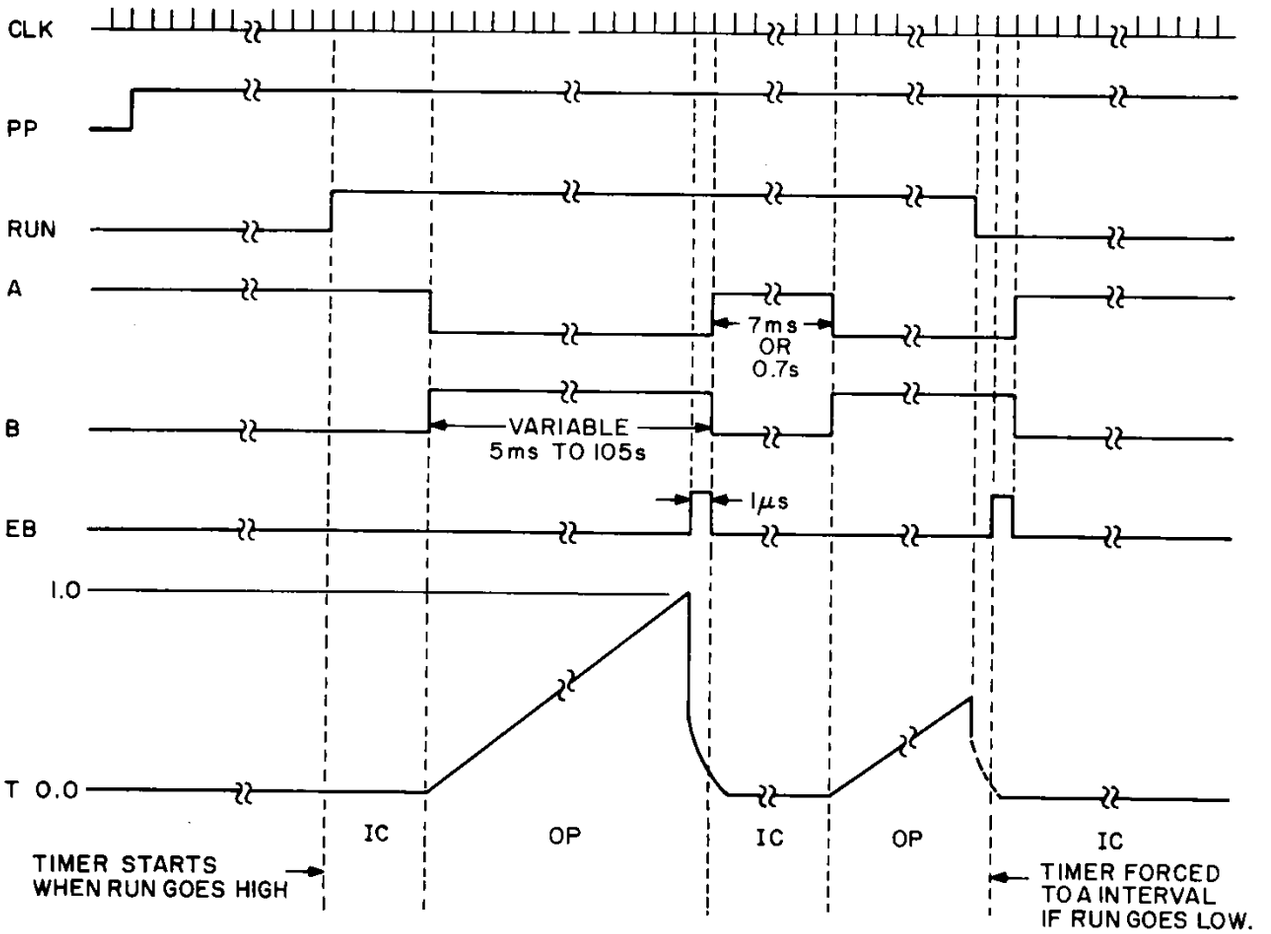
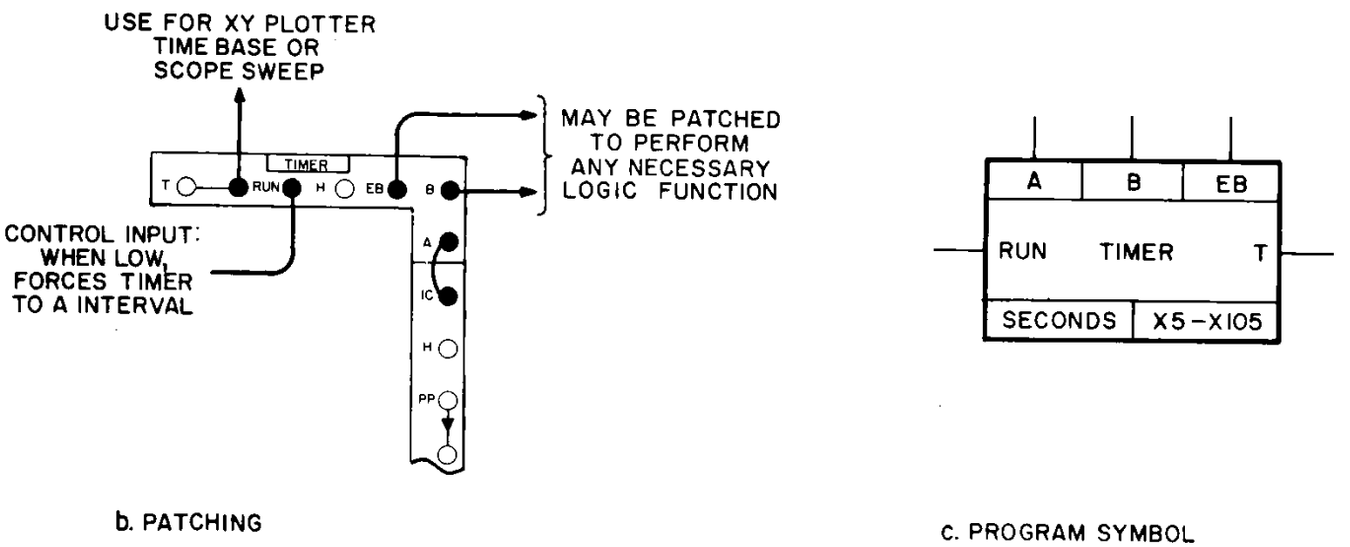


Figure 29.2. Typical Use (RUN and H Unpatched): Timer Starts When PP is Depressed.



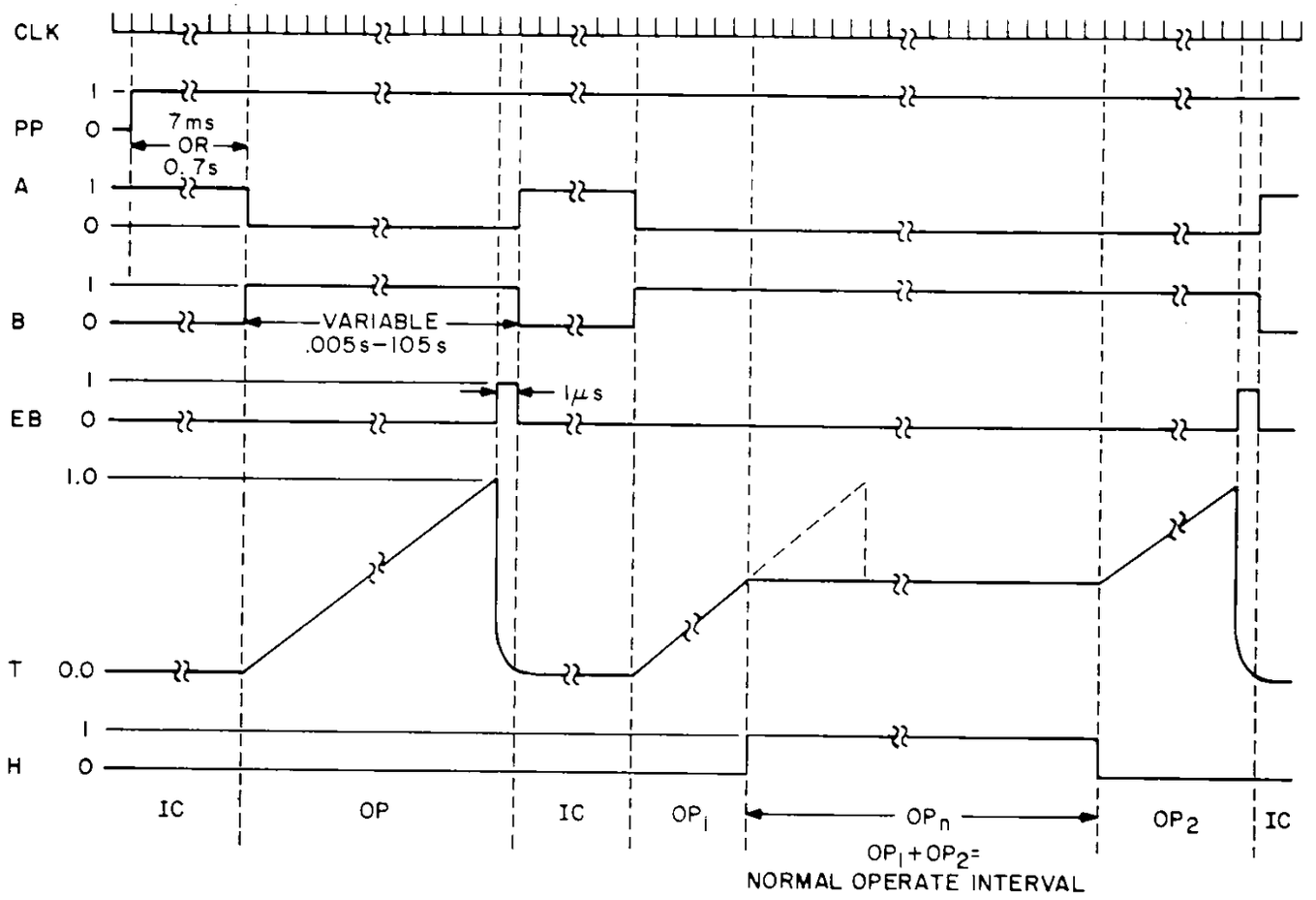
a. BASIC TIMING



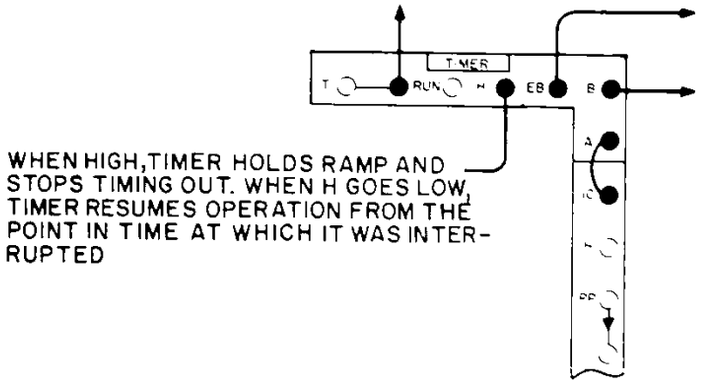
b. PATCHING

c. PROGRAM SYMBOL

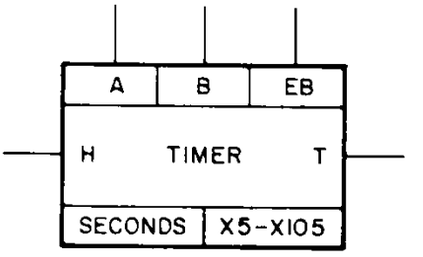
Figure 29.3. Typical Use of RUN: Delayed Start



a. BASIC TIMING



b. PATCHING



c. PROGRAM SYMBOL

Figure 29.4. Typical Use of H: Extended B (OP) Interval

29.3 THE COUNTER/TIMER

29.3.1 GENERAL OPERATION

The counter/timer can be used as; an event counter, a simple event monitor, a monostable multivibrator (one-shot), and a timer. When used as an event counter or as a one shot, the device cycles through a pre-selected count and stops. When used as a timing device, the counter/timer cycles through a pre-selected count continuously.

Basically, this device is an upcounter with a counting capacity of $127_{(10)}$. The desired count is pre-selected manually by pushbuttons. The counter/timer (Figure 29.5) consists of; a carry-in flip-flop, a count-out flip-flop (with complementary outputs) a 7 bit toggle flop register, input control logic, decoding gates, and manual switches for pre-selecting the count to be decoded and program checkout purposes. This device responds to input pulses to provide a logic output that goes high and remains high until a predetermined count is attained. The counter/timer operates in such a manner that output is high for the pre-selected count; (as entered on pushbuttons) plus a count of one. Therefore, the pushbutton entry must be one less then the desired count.

29.3.2 THE MANUAL CONTROLS

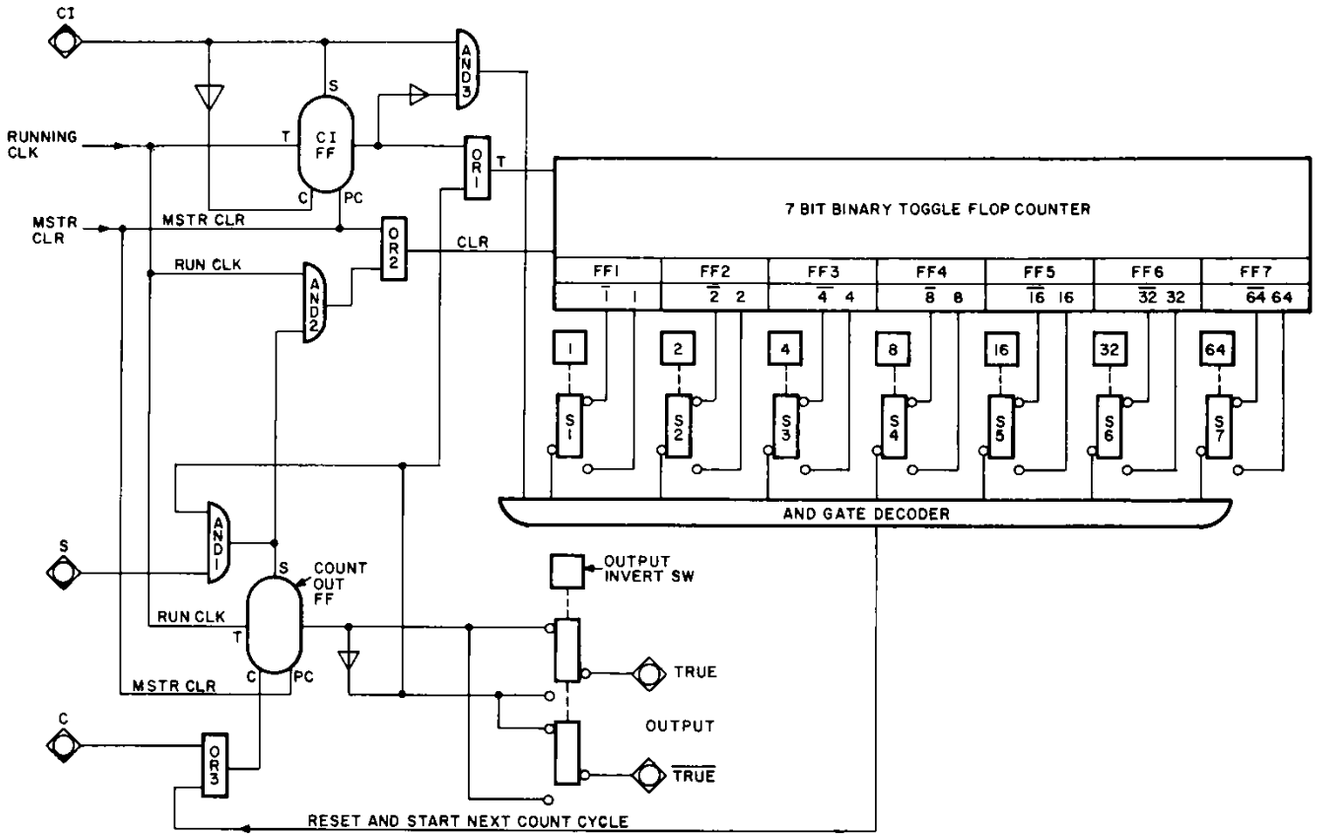
There are eight pushbutton switches on the logic control panel (Figure 29.5) associated with operation of the counter/timer. The numbered pushbuttons (CTR/TMR-64,32,16, etc.) are used to pre-select the desired count. The undesignated (blank) pushbutton is provided for program checkout, and when depressed, reverses the state of the CTR/TMR true and false outputs at the patch panel. This pushbutton is momentary and when released, the outputs return to their former state. The blank pushbutton also contains an indicator lamp that is illuminated whenever the CTR/TMR true patch panel output is high.

The seven numbered pushbutton switches each correspond to a flip-flop in the toggle register and are of the latching type. The numeric designation is the binary coded decimal (BCD) weight of the associated flip-flop. Depressing any one, or any combination of these pushbuttons pre-selects the number of counts through which the counter/timer will cycle. When the pre-selected count (plus one) is reached, the flip-flop outputs are gated to the clear side of the count out flip-flop and the counter resets (true output goes low). The desired count is pre-selected by setting those pushbuttons whose total BCD weight (as determined by the numeric designations) is equal to the desired count minus one. For example, if an actual count of 27 is desired, depress the 16, 8, and 2 pushbuttons. This totals 26, one less than the desired count.

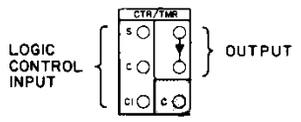
Indicator lamps behind each pushbutton light whenever the corresponding flip-flop sets. Therefore, when a given count is completed, these pushbuttons indicate the number of counts actually performed. Under given operating conditions, the counter timer can be used as an event monitor (Paragraph 29.3.6) and the number of counts will remain displayed on the indicators. Remember that, the actual count is one more than the total of the pre-selected count. Therefore the counter will display the total of the set pushbuttons, plus one. Using the previous example, 26 (16, 8, 2 pushbuttons set) was entered to obtain a count of 27. When the count is complete, 16, 8, 2, 1, will be displayed.

29.3.3 THE PATCH PANEL TERMINATIONS

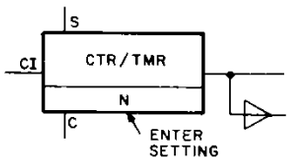
The counter/timer is terminated in the patch panel logic field. Two red output terminations provide a true and false logic output. When the count-out flip-flop is set, the true output (upper termination) is high. When the flip-flop is reset, this output (referred to as CO) is low. In either state, the lower termination is the false output (referred to as $\overline{CO}$), and is the complement of the true output as indicated by the logic inverter symbol.



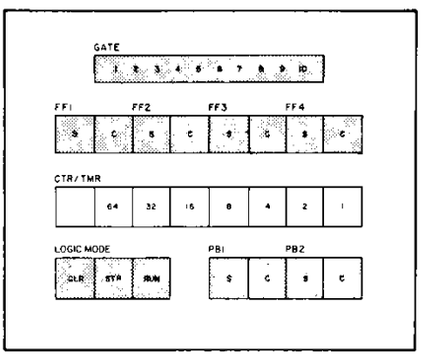
a. SIMPLIFIED CIRCUIT



b. PATCHING TERMINATIONS



c. PROGRAM SYMBOL



LOGIC CONTROL PANEL

d. CONTROLS

Figure 29.5. The Counter/Timer

Three inputs (designated CI, S and C) provide control of counter/timer operation. A series of positive pulses applied to the CI (carry-in) termination increments the counter once for each carry-in until the pre-selected count (plus one) is reached. The S (set) input termination is normally low. This termination controls the count-out flip-flop and is used to initialize the counter/timer. If pulsed high, the counter/timer will go through one counting operation. If S is held high, the counter will cycle continuously.

The C (clear) input termination is normally low. A high applied to this termination clears the count-out flip-flop, causing the true output to go low. This does not clear the toggle flop counter. If C goes high while CI is low, the counter will increment one and the indicated output may be invalid. Should S and C both be high simultaneously, the output flip-flop will toggle on clock. This would provide incorrect timing or counting data and should not be permitted during normal operation. If both the S and C inputs are used, C should be the complement of S.

29.3.4 GENERAL USE

As previously mentioned, the counter/timer can be used to perform several counting and/or timing functions. In any of these operations, the pushbutton must be preset to select the desired count or time interval. The following steps outline the general order of procedure for using the counter/timer.

1. Patch the counter/timer into the problem in accordance with the program requirements.
2. Select the Logic CLR mode.
3. Depress the numbered pushbuttons so that they total the desired count minus one.
4. Select the logic RUN mode.
5. If it is desired to stop the counter at any point in the program for analysis purposes, merely select the logic STOP mode. The timer will stop at the count attained at the time STOP is selected. To restart the counter, select the logic RUN mode.

29.3.5 USED AS AN EVENT COUNTER

The counter/timer may be used to count a given number of random or timed events and provide an output to control any given logic function. When used as a counter and control device, the set input (S) must be patched to a logic one, the CI input must be patched to the device generating the event to be counted, and the outputs (CO and/or $\overline{CO}$) must be patched to the device to be controlled. Then the desired count (minus one) must be pre-selected using the TMR/CTR pushbuttons. Figure 29.6 illustrates the basic timing and patching requirements for this use.

29.3.6 USED AS AN EVENT MONITOR

The counter/timer can be used to merely display the number of times that a logic event occurs. This is accomplished by patching the true logic output of the event to the CI input. All counter output patching is eliminated. Then, simply pre-select all of the numbered pushbuttons (64, 32, 16, etc.) so that N = 127. When the logic RUN mode is selected, and an event occurs, the timer will start counting. When the problem run is completed, simply add up the lighted number display on the pushbuttons. This total is the number of times that the event has occurred.

NOTE

If the logic CLR mode is entered, the toggle register resets and the display is extinguished. Do not depress CLR until the display is readout.

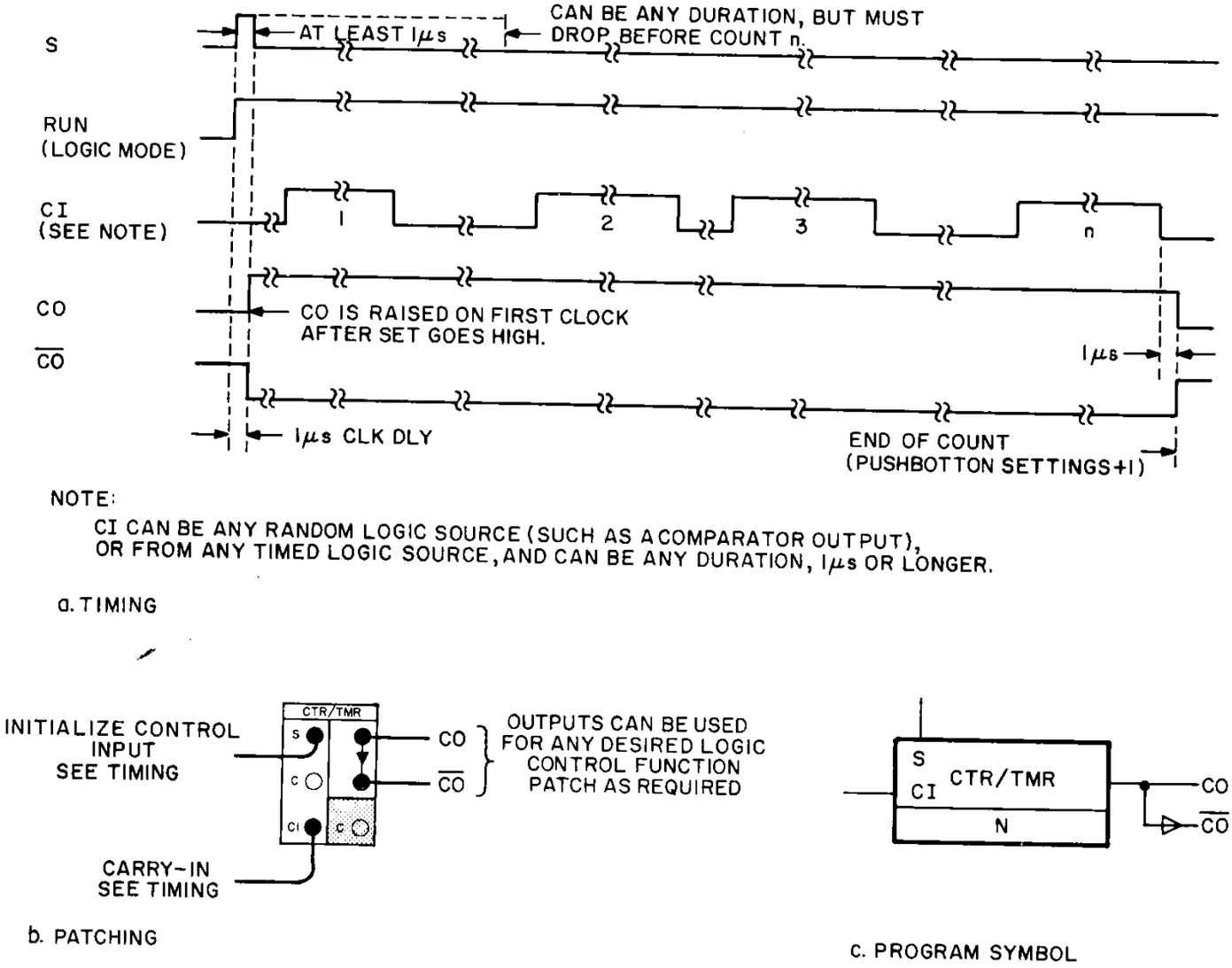
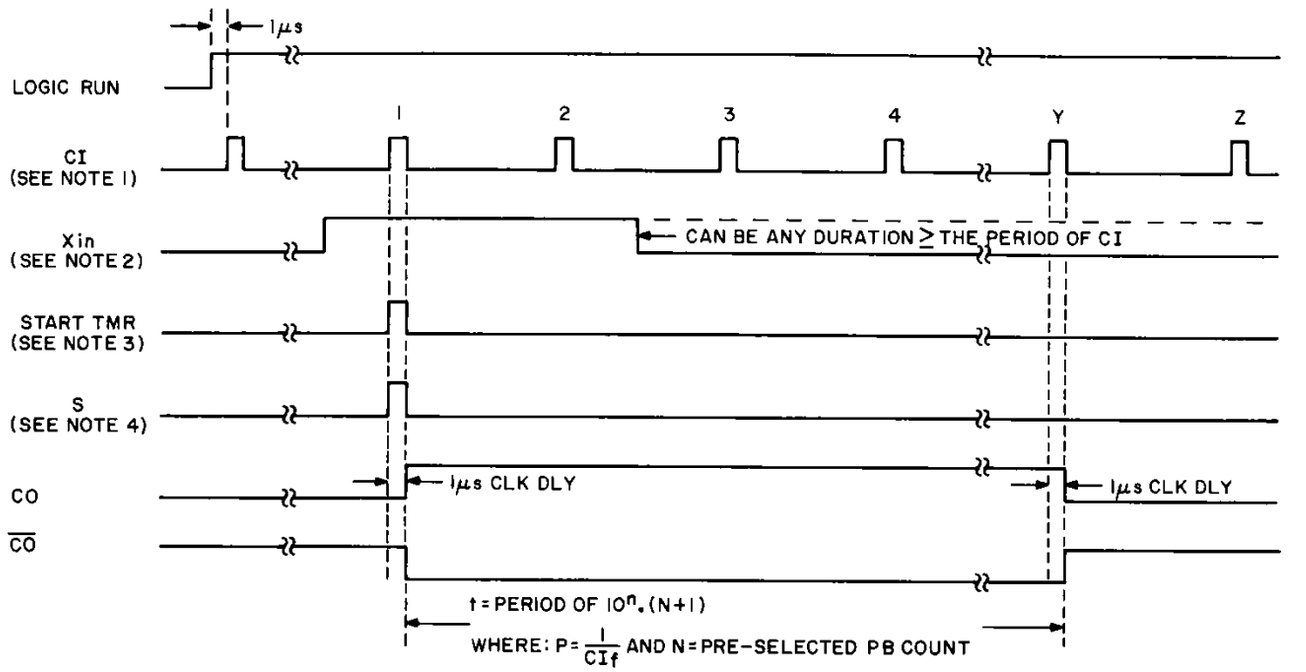


Figure 29.6. Event Counter: Patching and Timing

29.3.7 USED AS A ONE-SHOT

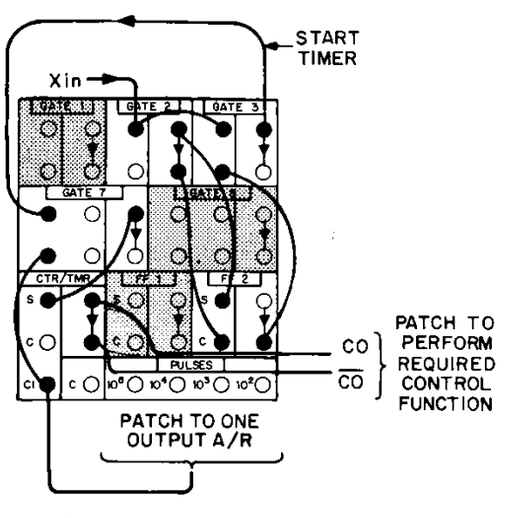
Some problems may require an input or control pulse of precise duration after some given event occurs. This can be by using the counter/timer as a one shot device. As described in Chapter 15, the system clock is downcounted to provide four clock frequencies available at the patch panel PULSES terminations. Any one of these (10^5, 10^4, 10^3, or 10^2,) can be used as a carry-in (CI) signal to obtain an output pulse of fixed duration as shown in Figure 29.7. For example, to generate a one-second output pulse, one hundred 10^2 PULSES must be counted. Note that a 10^2 pulse has a period (P = 1/F) of 0.01 second. Therefore, 100 x 0.01 is equal to 1.0 second. To obtain an output pulse of fixed duration, the event T that initiates the one-shot time (Xin) must be differentiated and then gated with the CI input. This is necessary to synchronize the leading edge of CO with the 10^2 clock to obtain an accurately timed output. For this example, the CTR/TMR pushbuttons must be set to a count of 99. The first CI input (after Xin) coincides with the START TIMER signal and sets the counter (CO is raised). The counter steps one count on each succeeding CI input for a total of 100 counts (99 + 1), or one second. After the 100th count, CO drops and cannot go high until Xin is dropped and again raised.



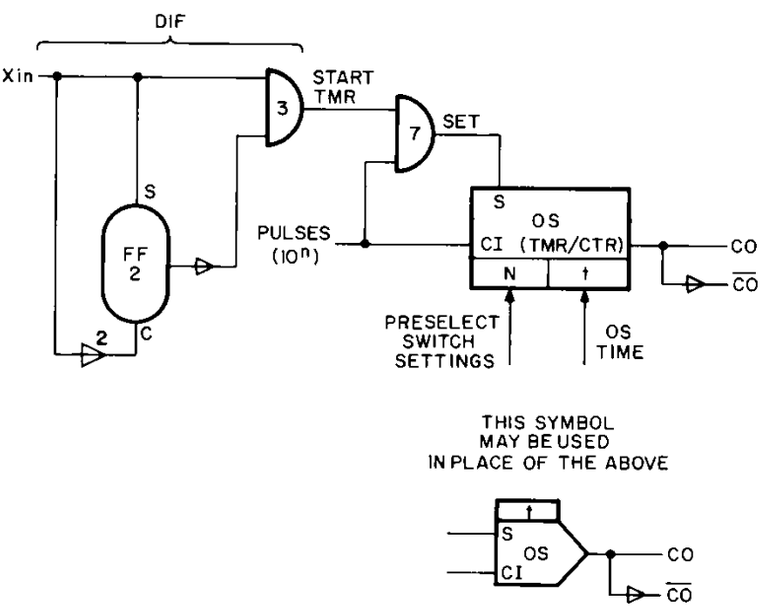
NOTES:

1. CI = 10^n PULSES (10^5, 10^4, 10^3, OR 10^2)
2. X = ANY SYNCHRONOUS LOGIC EVENT
3. START TIMER = 1ms PULSE (X DIFFERENTIATED)
4. S=SET TIMER MUST BE GATED WITH THE 10^n PULSES PATCHED TO CI.

a. TIMING



b. PATCHING



c. PROGRAM SYMBOL

Figure 29.7. One-Shot: Patching and Timing

Numerous one-shot times can be generated by using various combinations of pushbutton count selections in conjunction with the PULSES frequency selected as the CI input. The following list gives some basic combinations and the corresponding pulse duration for each.

CI	N	ONE SHOT TIME
10^2 (P-0.01S)	99	1.0S
	9	0.1S
10^3 (P=1 millisecond)	99	0.1S
	9	0.01S or 10 milliseconds
10^4 (P=0.1 millisecond or 100 microseconds)	99	10 milliseconds
	9	1 millisecond
10^5 (P=10 microseconds)	99	1 millisecond
	9	10 microseconds

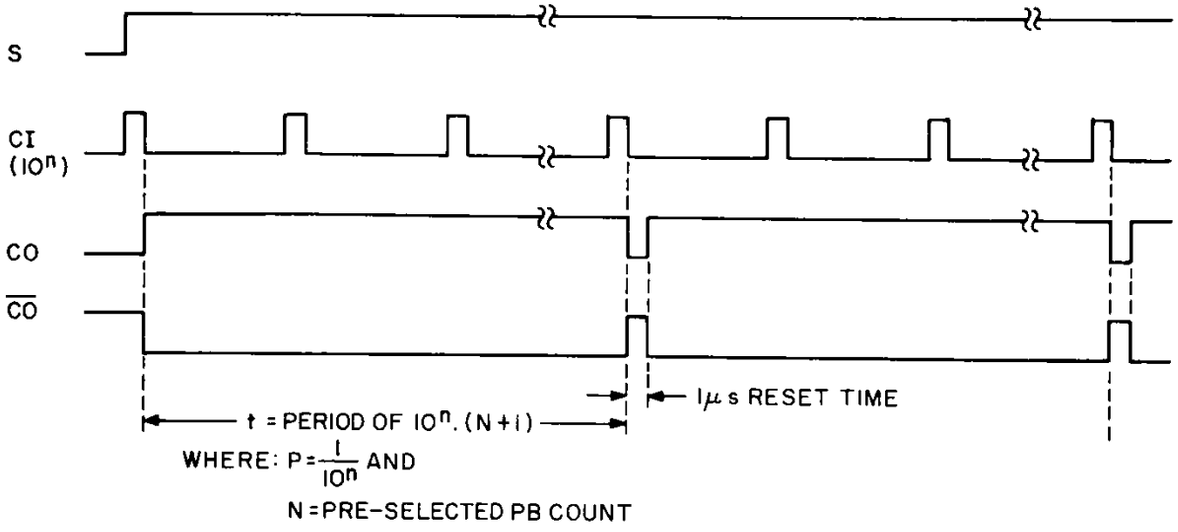
29.3.8 USED AS A LOGIC TIMER

The counter/timer may be used to time logic events so that they re-occur at predetermined intervals. When used as a timer, the device re-cycles continuously and produces a pulse train output. The timer is forced to re-cycle through the pre-selected count (plus one) by patching the S input high (Figure 29.8) and applying a down counted clock (PULSES) to the CI input. The counter will cycle continuously until S drops, or, the CLR or STOP logic mode is selected. If S is high, the true output (CO) goes immediately after the first carry-in. CO remains high until the preselected number (pushbutton setting) plus one is counted. This drops for one microsecond and the cycle is repeated. As shown in the timing diagram in Figure 29.8, the CO output is a positive pulse train whose duration is determined by the pre-selected count and the frequency of the PULSES applied to CI. The CO output is used to provide a fixed time for control of a logic function. Conversely, the $\overline{CO}$ output is a series of timed positive one microsecond pulses. The $\overline{CO}$ output can be used for timing the occurrance of a logic event.

As in the case of the one shot (Paragraph 29.3.7) numerous timed intervals can be obtained by using various combinations of pre-selected counts in combination with the frequency (10^5, 10^4, 10^3, and 10^2,) of the PULSES applied to the CI termination.

29.4 THE MONOSTABLE MULTIVIBRATOR

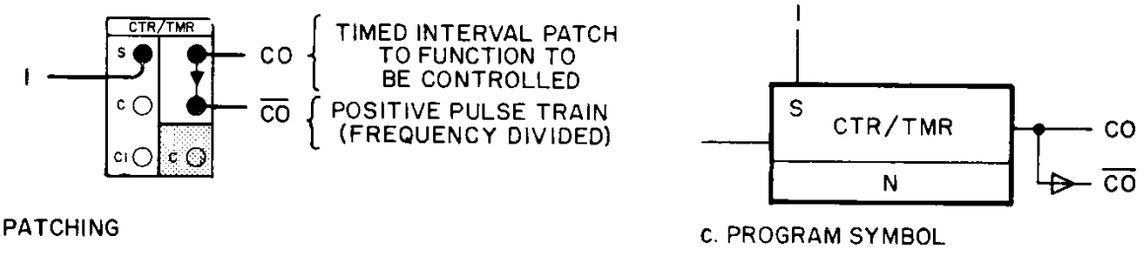
The monostable multivibrator (MONO) is a logic device whose output goes high for approximately 0.5 to 1.0 milliseconds whenever its input goes high. This device can be used to control any logic function as required by the overall program, and is commonly used to provide the analog Hold (H) mode, during three-mode iterative operations as described in Chapter 14. Figure 29.9 illustrates the basic patching requirements and program symbol for the MONO.



NOTES:

1. CI = 10^n PULSES (10^5, 10^4, 10^3, OR 10^2)
2. TIMER WILL CONTINUE TO CYCLE UNTIL S INPUT IS DROPPED, OR THE LOGIC STOP OR CLR MODES ARE SELECTED.

a. BASIC TIMING



b. PATCHING

c. PROGRAM SYMBOL

Figure 29.8. Frequency Divider: Patching and Timing

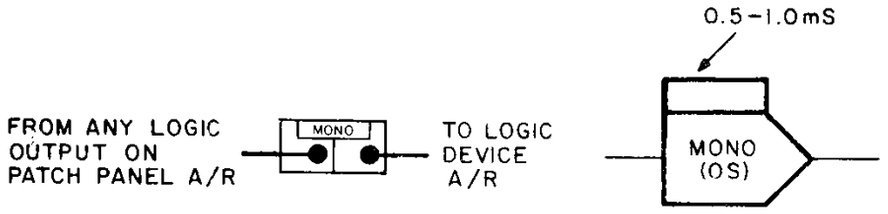


Figure 29.9. The MONO