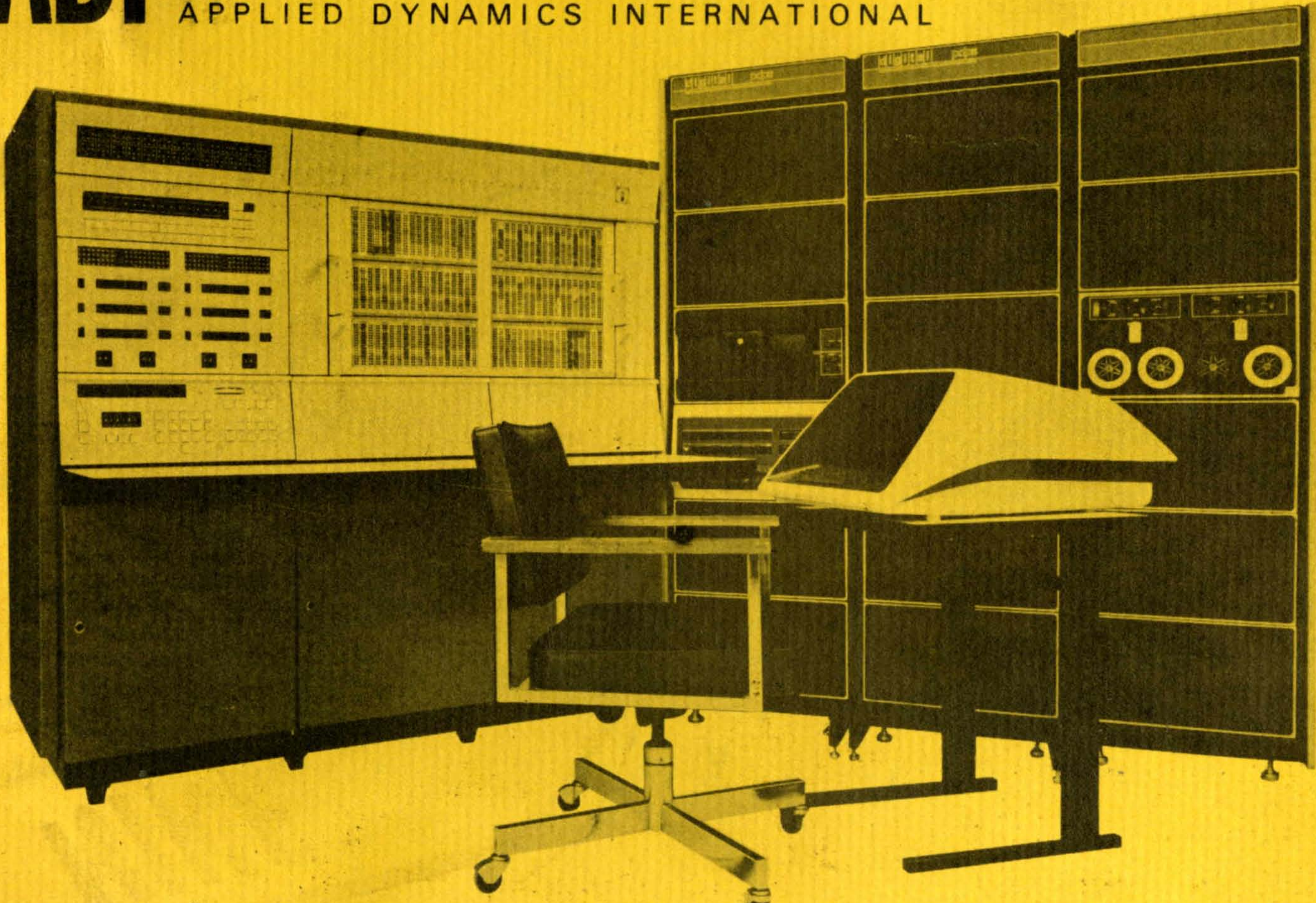
AD/FIVE REFERENCE MANUAL

VOLUME 1
ANALOG

ADI APPLIED DYNAMICS INTERNATIONAL



3800 Stone School Road • Ann Arbor, Michigan 48104 • Phone 313-973-1300

CHAPTER 1

INTRODUCTION

TABLE OF CONTENTS

ILLUSTRATIONS

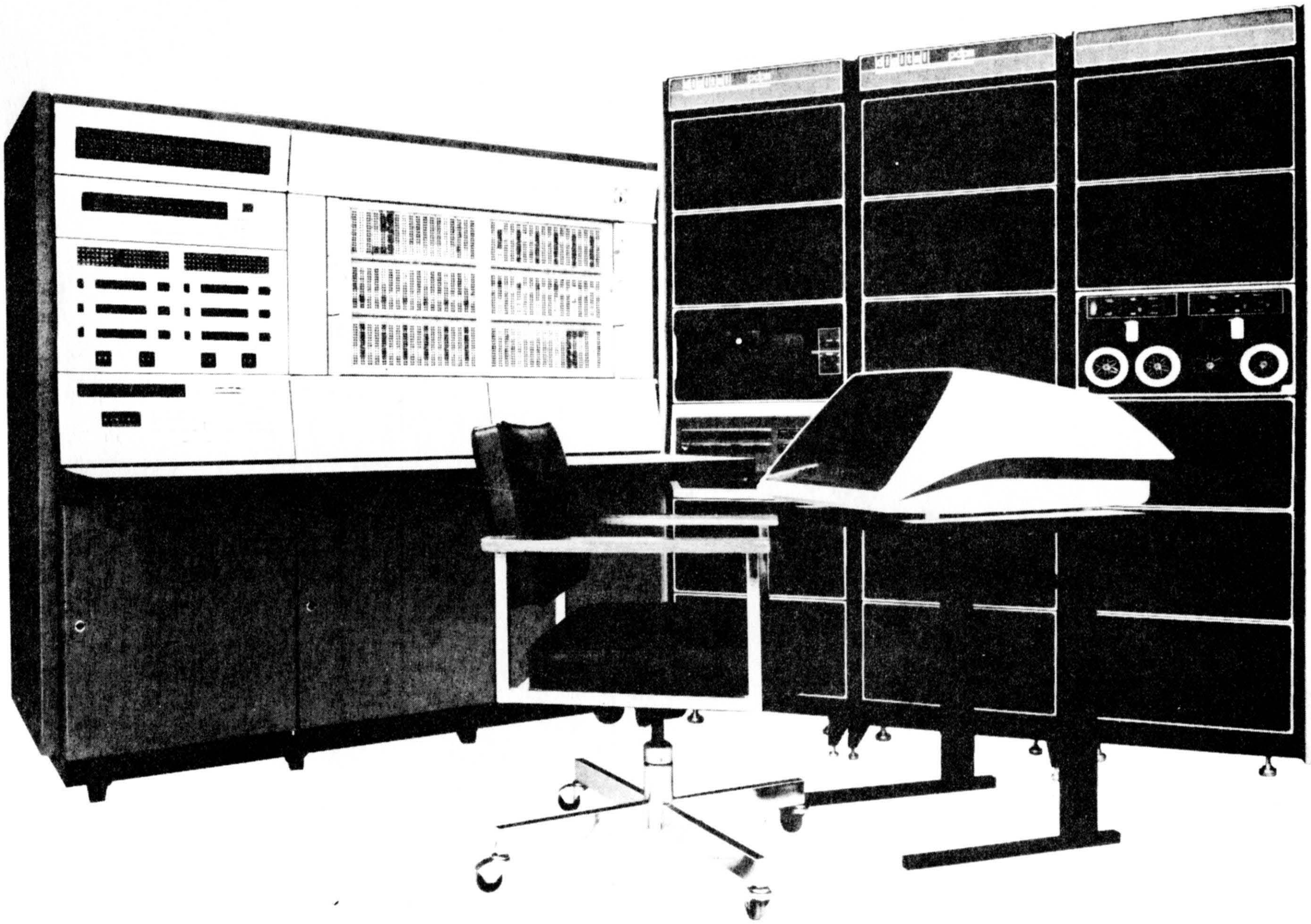


Figure 1-1 The AD/FIVE Analog-Hybrid SYSTEM 511

1.0 INTRODUCTION

The AD/FIVE Reference Manual was written as a comprehensive reference source for the Applied Dynamics AD/FIVE Analog Computer. Chapter 1 introduces the primary features of the AD/FIVE and gives a brief description of the various computing components of the computer. Chapter 2 gives a detailed view of the organization of the AD/FIVE. Chapter 3 discusses the Control System of the AD/FIVE. Chapter 4 gives all directions necessary to achieve proper patching of all patchboard elements of the AD/FIVE. Chapter 5 introduces some essential analog and logic circuits to help the user obtain full benefit of the computing capability of the AD/FIVE computer. Finally, the Arbitrary Function Generator is discussed fully in an appendix to the manual. The primary subject matter is thus logically developed to furnish a working knowledge of the AD/FIVE, and to enable a user unfamiliar with the computer to be able to implement a program and control it as he desires.

The AD/FIVE carries on several new concepts introduced by Applied Dynamics in earlier computers. For those unfamiliar with the design of AD computers, the following discussion should furnish insight into the guiding philosophies of the AD/FIVE Analog Computer.

No significant advance in major computer design would be expected without first making a detailed analysis of the shortcomings and strenghts of present generation equipment and, simultaneously, relating this knowledge to a qualified understanding of the required computational tasks. Applied Dynamics, through its problem-oriented staff, has done this. By building upon a foundation of achievement in scientific computation, AD designers have produced a new state-of-the-art computer system. Advanced concepts in basic configuration, control, component design, patchboard termination, slaving, man-machine communication, and hybrid interfacing all interact to widen the scope of scientific and engineering problems that can be solved.

The predominant philosophy in the development of the AD/FIVE was that it be inherently hybrid in design; this is reflected throughout the control structure, in the patchable logic system, and in an efficient integral hybrid interface. Because hybrid and high speed iterative analog computation places particular demands upon both the dynamic and static performance of the analog computer, emphasis has been placed on overall system performance through component and system design. The result is an optimum balance of static and dynamic accuracy.

In addition to the desirability of having the best computational accuracy obtainable, computer laboratories today place premiums on total system "up-time". This direct measure of return on the invested computer-dollar demands that the manufacturer account for all reliability parameters for each step of design and production. The Applied Dynamics continuous philosophy of conservative circuit design has been incorporated throughout the development of the AD/FIVE to insure compliance with the maximum uptime criterion. A decade of proven product integrity stands behind this assurance of reliability.

The AD/FIVE was designed to provide both simplicity and flexibility of operation. These apparently contradictory goals were met by adopting a philosophy of control by exception. For a small problem, very few patching connections are required to configure the amplifiers, and even on a large problem the number of patching connections is reduced. The control system incorporates this principle to an even greater degree. Designed to be used by both the inexperienced and most sophisticated programmer, the control by exception technique is found throughout the control system. All integrator mode switches and time scale selection switches are internally connected to pushbuttons so that the computer can be operated as a standard analog computer with no logic patchboard patching. However, groups of integrators or individual integrators can be separately controlled by patching. The AD/FIVE control system is as flexible as the operator desires for any specific problem.

The basic design of the AD/FIVE interface was conceived years after the design and application of the initial hybrid computing systems. Consequently, Applied Dynamics was able to incorporate into the AD/FIVE interface its own past knowledge and those innovations resulting from the advance of technology.

The most outstanding feature of the interface is its degree of integration into the AD/FIVE computing system. This was accomplished by assimilation of some of the special purpose devices previously required for hybridization in a basic hybrid system, and also by elimination of the need for other hybrid devices through the design of the control structure. The underlying design assumption was that the AD/FIVE would augment a digital computer.

The AD/FIVE control and monitoring structure has been specifically designed for straightforward multiple-console computation in both stand-alone and hybrid system operation. Two of the more obvious advantages inherent in a two-console system over a single computer of equivalent total equipment complement are flexibility and capacity for economical expansion. There is also the corollary benefit of improved overall computing laboratory efficiency from a problem loading standpoint, i.e., simultaneous running of two independent smaller scale problems. An overall reliability increase is also implied because of the system redundance and module-level interchangeability. Full two-console (or more) operation is accomplished through cross-trunking of analog and synchronized logic signals plus analog mode, logic mode, and time-scale control signals. The control functions are easily accomplished by using eight of the interconsole logic trunks. Two-phase clocking of the double-ranked logic elements, and external access to both clocking signal busses at local and console levels insures true synchronous operation between slaved machines. External access to the overriding Master Hold inputs on a console, as well as the ability to inhibit these imputs from the Master Overload bus on the master console, further augments the operational independence of multiple consoles in a system.

The unique modular capability of the AD/FIVE as a system, contrasted to previous computers which have been modularized for components only, introduces a growth capability not previously obtainable. Since the user need not originally invest in complete system wiring at the patchbay, and power and signal wiring to unexpanded areas, he is able to make an original modest investment, and then later he can expand the system as the computing components are expanded. Advanced wiring plane/cabling techniques make this feature possible.

1.1 COMPUTING COMPONENTS

1.1.1 Analog Computing Components

The heart of any analog system is the operational amplifier. The AD/FIVE offers an amplifier of exceptional performance and stability. The solid state amplifier offers the following typical performance features: a voltage range of ±13 volts; a gain bandwidth (10K/10K) of 1.5 MHz; noise peak-to-peak (0-800 KHz) of 1.5 Mv; T.I.D.E. at 1.0 KHz of 0.04%; and overload recovery from either voltage or current overload condition of 0.5 Msec. The amplifier is further short circuit proof to any voltage in the range ±15 volts.

Integrators can be time scaled at X1, X10, X100, and X1000 by individual patched control or by pushbutton operation. Six inputs are available: 3 gains of 1 and 3 gains of 10. Electronic switching of mode control allows operation in IC (Initial Condition), OP (Operate), and H (Hold), whether controlled by pushbuttons or individual patching; switching time is typically 900 nanoseconds between any two modes. Drift in Hold is typically 50 microvolts/sec. at X1 or X10 time scale.

Summer/High Gain amplifiers are available with 3 gain one inputs, 3 gain 10 inputs, 3 outputs, and 2 output diodes. Switch/Summer amplifiers have two electronically switched inputs and one unity gain input. Control of the 900 nanosecond switches is available on the logic patchboard.

Potentiometers are available as servo-set or hand set modules, with a resistance of 5 Kohms, typical resolution of 0.02%, and typical phase shift range at 1.0 KHz of $\pm 0.1^{o}$ for setting greater than 0.1.

Non-linear computing components available with the AD/FIVE include Dual Square Function Generators, Sin/Cos Function Generators, Log Function Generators, Arbitrary Function Generators, and Multipliers. Precision Multipliers in the AD/FIVE offer typical performance such as: static error ($|x| + |y| < 20$ volts) of ±5 Mv, with zero error ($x=y=0$) of ±0.5 Mv; bandwidth ($x = 10$ volts, $y = 20$ volts peak-to-peak) of 1.0 MHz; and phase shift of 0.03^{o} at 1.0 KHz. T.I.D.E. at 1.0 KHz is typically 0.06%.

Track Store Networks are available for use with Summer/High Gain amplifiers, Summer/Integrator amplifiers, or Inverter amplifiers. Network switching time is less than 900 nanoseconds.

Limiters are available on the AD/FIVE to limit the range of an operational amplifier between -10 volts to +10 volts on both +L and -L, with slope after limit typically 1.0 Mv/volt.

Free Impedance Network circuit cards are also available on the AD/FIVE to allow maximum flexibility to meet all user needs.

1.1.2 Logic Computing Components

A precision 1 MHz oscillator is the master timing source for the AD/FIVE logic. The full complement logic system includes flip-flops, flip-flop differentiators, two-decade BCD counters, gates with two, four, and six inputs, DPDT relay switches, function and logic switches, trunks, and hybrid lines. Indicators are provided for flip-flops, gates, comparators, and counters. Preset controls are a useful adjunct to BCD counters. In addition, there are trunks and hybrid lines for implementation of full logic control in hybrid usage.

An interval timer with three pre-settable periods with BCD circuitry can also be used as a logic computing component, with a carry-out available with every one of the three intervals.

1.2 CONTROL AND ADDRESSING SYSTEM

The Control Panel of the AD/FIVE offers numerous features to implement ease and flexibility of control. Pushbutton control is available for analog modes of integrators IC, OP, and H, and for logic modes LOAD, RUN, and STOP. Time scales may be controlled by pushbuttons labelled X10 and X100. Rep-op operation is pushbutton controllable, allowing analog modes to be controlled by the three pre-settable periods of the Interval Timer, which has a counter input derived from the 1.0 MHz logic clock.

Other pushbutton control features include COEF X IN for reading true potentiometer output values; TEST for ascertaining initial values of inputs to integrators; PROB VER for checking IC values of integrators; AUTO HOLD to put all integrators into Hold in case of the occurrence of any component overload; LOGIC STEP to allow logic to run for a programmable number of clock periods; UP and DOWN controls to allow manual slewing of servo-set pots; PB to allow patched inputs to be read via the Digital Ratiometer of the Addressing System; control of the pulses counted by the Interval Timer in decade steps from V1SEC to V10MS; PATCHBOARD to control the patchboard motors; and controls to enable slaving and hybrid operation.

In addition there are a number of useful controls associated with the Control Panel Analog Voltmeter.

The Addressing System is capable of addressing up to 721 addressable elements in a fully expanded AD/FIVE, with circuitry and switches all solid state. The multiplex rate for hybrid operation is 10,000 points/sec. when used with an A-D Converter.

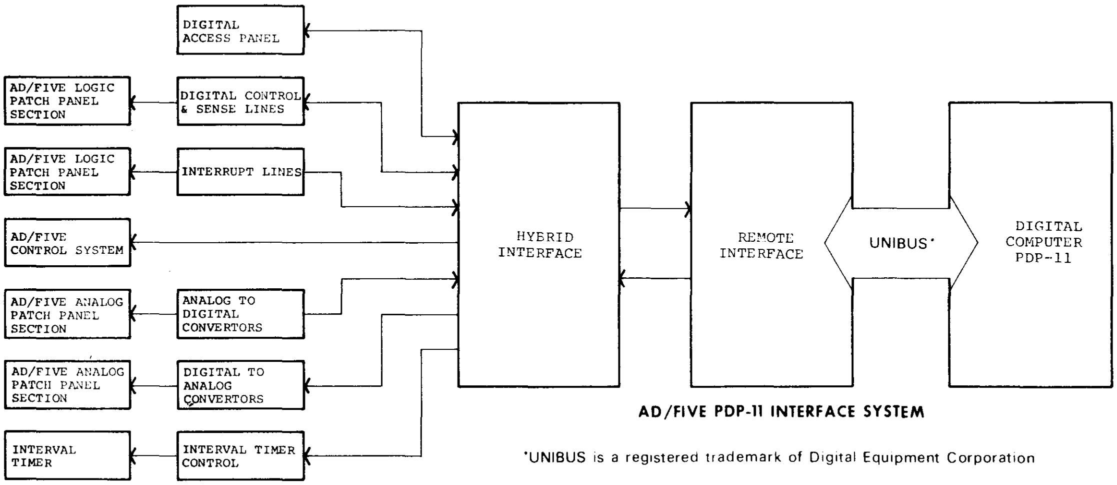
1.3 HYBRID FEATURES

Hybrid usage of the AD/FIVE is discussed in a separate manual. Only the main points of the hybrid system will be mentioned here. The AD/FIVE is joined by interface hardware to the digital computer to allow digital control of the analog and logic modes, repetitive operation, address and data registers, time scale, autohold, problem verify, setting and reading coefficient devices, interval timer, digital ratiometer, analog-to-digital conversion, digital-to-analog conversion, sense lines, control lines, and interrupt lines. The analog-to-digital converter system in the interface is a 100KHz, 14-bit plus sign, 10 volt unit. Sixteen multiplexer channels are supplied with the basic converter, and an additional sixteen channels may be added as desired. The A-to-D converter features random access or sequential modes of operation under computer control.

The Digital-to-Analog Converter (DAC) system includes double-bufferred, 14-bit plus sign multiplying D-to-A converters. Digitally Set Coefficient Units (DCU's) are also available, and are single-bufferred, 14-bit magnitude devices in two versions: two quadrant and four quadrant.

Sixteen control lines and sixteen sense lines, as well as eight interrupt lines, are terminated in the logic section of the AD/FIVE patchboard. The control lines are set and the sense lines read by the digital computer through the interface system, or by manual control from the Digital Access Panel option (DAP).

Figure 2 shows a diagram of the organization of the AD/FIVE hybrid system.


DIGITAL ACCESS PANEL
AD/FIVE LOGIC PATCH PANEL SECTION
DIGITAL CONTROL & SENSE LINES
AD/FIVE LOGIC PATCH PANEL SECTION
INTERRUPT LINES
AD/FIVE CONTROL SYSTEM
HYBRID INTERFACE
REMOTE INTERFACE
UNIBUS*
DIGITAL COMPUTER PDP-11
AD/FIVE ANALOG PATCH PANEL SECTION
ANALOG TO DIGITAL CONVERTORS
AD/FIVE ANALOG PATCH PANEL SECTION
DIGITAL TO ANALOG CONVERTORS
INTERVAL TIMER
INTERVAL TIMER CONTROL

AD/FIVE PDP-11 INTERFACE SYSTEM

*UNIBUS is a registered trademark of Digital Equipment Corporation

Figure 1-2 The AD/FIVE-PDP 11 SYSTEM 511

1.4 MAN-MACHINE COMMUNICATIONS

In the design of the AD/FIVE a great deal of emphasis was placed on optimizing interrelationships between the operator and the computer. Care was taken to arrange for maximum operator convenience in the placement of controls and arrangement of the patchboard. Ease of maintenance was also a major factor. The features of AD/FIVE man-machine communication are discussed below.

1.4.1 Operator Convenience Features

1. The Control System of the AD/FIVE is all housed in one bay, so that all control functions are available to the operator in one area.

2. Hand-set potentiometers are easily accessible behind a hinged door panel below the patchboard, so that the operator can conveniently set pots.

3. Diode Arbitrary Function Generators and Calibration Units are conveniently placed behind a hinged door panel above the patchboard, so that the operator can make all necessary calibrations and settings with ease and comfort.

4. The entire control area is well lighted for visual comfort and ease of reading indicator lights. The overload indicator panel, Digital Access Panel, and logic state indicator lights are all easily readable. Lighted pushbuttons give their information on the state of the computer to the operator quickly and accurately. The address and data register display windows are well lighted and easily readable.

5. The organization of the patchboard is designed to give maximum efficiency to the operator in patching. The patchboard is divided into two fields horizontally, with each field having an upper logic section, and upper and lower analog sections. Each field has six areas which extend horizontally for six patchboard holes per area, and vertically through all three sections of the logic and upper and lower analog. Each area has components clearly labelled and numbered, so that inputs and outputs are easily visible for convenient patching. Every component has an individual number associated with it, so that a component can be given a unique address through the use of its field, area, and individual numbers. A consistent layout of components area by area throughout the patchboard maximizes simplicity and ease of correct patching.

A harmonious, well-balanced, and subdued color scheme which is consistent throughout the patchboard is not only restful to the eye, but greatly facilitates location of components for patching. In the analog sections, integrators and the various summers and inverters are colored green, pots are colored yellow, trunks and hybrid elements in soft blue, non-linear components, free impedance networks, diodes, and limiters in brown, system lines and analog reference in grey. Analog/Logic devices such as comparators and track-store networks are in white on both the analog and logic sections of the patchboard. In the logic section, integrator and amplifier controls are green to match the analog colors, flip-flops and counters are in green, gates and relays in yellow, function lines and logic switches in brown, system controls in brown, peripheral device and hybrid lines and trunks in soft blue, and logic reference in grey.

Figure 1-3 shows the AD/FIVE patchboard and a typical patchboard module with the components terminated in it.

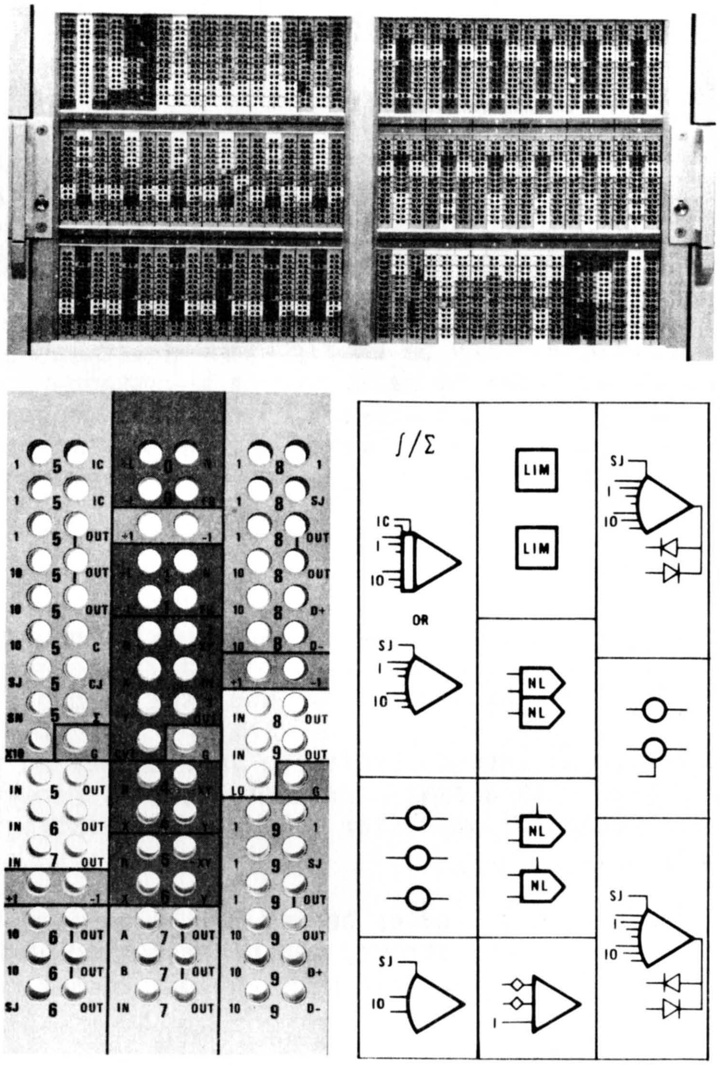


Figure 1-3 Patchboard Layout and Component Terminations

6. A workshelf across the entire width of the computer is a great convenience to the operator, giving ample room for comfortable writing and programming, and for supplies of patchcords and tools when needed. In addition, there is ample comfortable leg room for prolonged operator use of the computer.

7. It is virtually impossible to damage any part of the computer or components through improper patching of any terminations at the patchboard.

8. A cost saving convenience on an AD/FIVE up to one field full is the reversible patchboard, which can be rotated 180°, allowing two separate problems to be patched on one board, so that the user needs to pay for only half as many patchboards as if this unique feature were not available.

1.4.2 Ease of Maintenance

1. The AD/FIVE has operational temperature and humidity ranges such that it does not require a controlled atmosphere; thus it may be moved freely about any laboratory or building without affecting its computational ability, provided the broad environmental specifications are complied with.

2. The computer offers unmatched mobility because of its compact size. The front workshelf may be easily removed without the need for any tools, allowing the computer to pass through any standard thirty-inch doorway. The six casters are of a special composition chosen to support the computer without flat spotting while being pliable enough to pass over linoleum without scoring the surface.

3. All trim panels are removable for maintenance purposes without the need for any tools.

4. Non-linear components are located behind doors at the front of the AD/FIVE, so that most normal calibration can be accomplished conveniently from the front of the machine, without the need of removing any panels.

5. Amplifier balance controls are located behind the pot panel, and can be easily used with a small screwdriver, thus removing the danger of accidental alteration of amplifier balance and maintaining ease of necessary servicing.

6. Access to circuit cards behind the patchboard is easily gained. Only two bolts fasten each patchbay pin block behind each patch panel module, and removing the patchbay pin block allows access to three component cards at a time.

7. The various sections of the control bay are easily accessible on the AD/FIVE; only four screws need to be removed per section to lift out the overload indicator panel and the logic control panel. The DAP conveniently slides out on a rack for servicing. The control panel is mounted on a hinged door for ease of maintenance.

8. A number of useful maintenance kits are available with the AD/FIVE, and a most useful feature is the Programmable extender, which allows component cards to be physically removed from their computer locations, and yet still be functional as if they were in the computer, allowing diagnostic check-out and calibration to be easily and reliably performed.

1.5 MODULARITY AND EXPANSION CAPACITY

The AD/FIVE offers exceptional capacity for simple and efficient expansion. The modularity of the AD/FIVE computing components allows numerous variations of the patchboard component layout to suit varying user needs. The computer can be expanded from one to two fields by the addition of a wiring plane and power supply. Patchboard components can be expanded merely by plugging circuit cards to appropriate slots behind the patchboard. Non-linear components can be added beneath and above the patchboard in non-linear slots; non-linear access cards need to be present behind the patchboard, and, as the access feature is sometimes part of another computing card, it is sometimes easily possible to add a non-linear computing element without even removing the patchboard, depending on individual computer configurations.

All expansions of the AD/FIVE can be achieved without the need for soldering or extensive rewiring. Even expansion from a stand-alone analog computer to a hybrid system is easy and straightforward. The modularity of AD/FIVE components and the ease of analog and hybrid expansion, coupled with the precision and ease of use, as well as great flexibility, of the AD/FIVE make it the ideal analog computer for the most modern laboratory and professional or industrial applications.

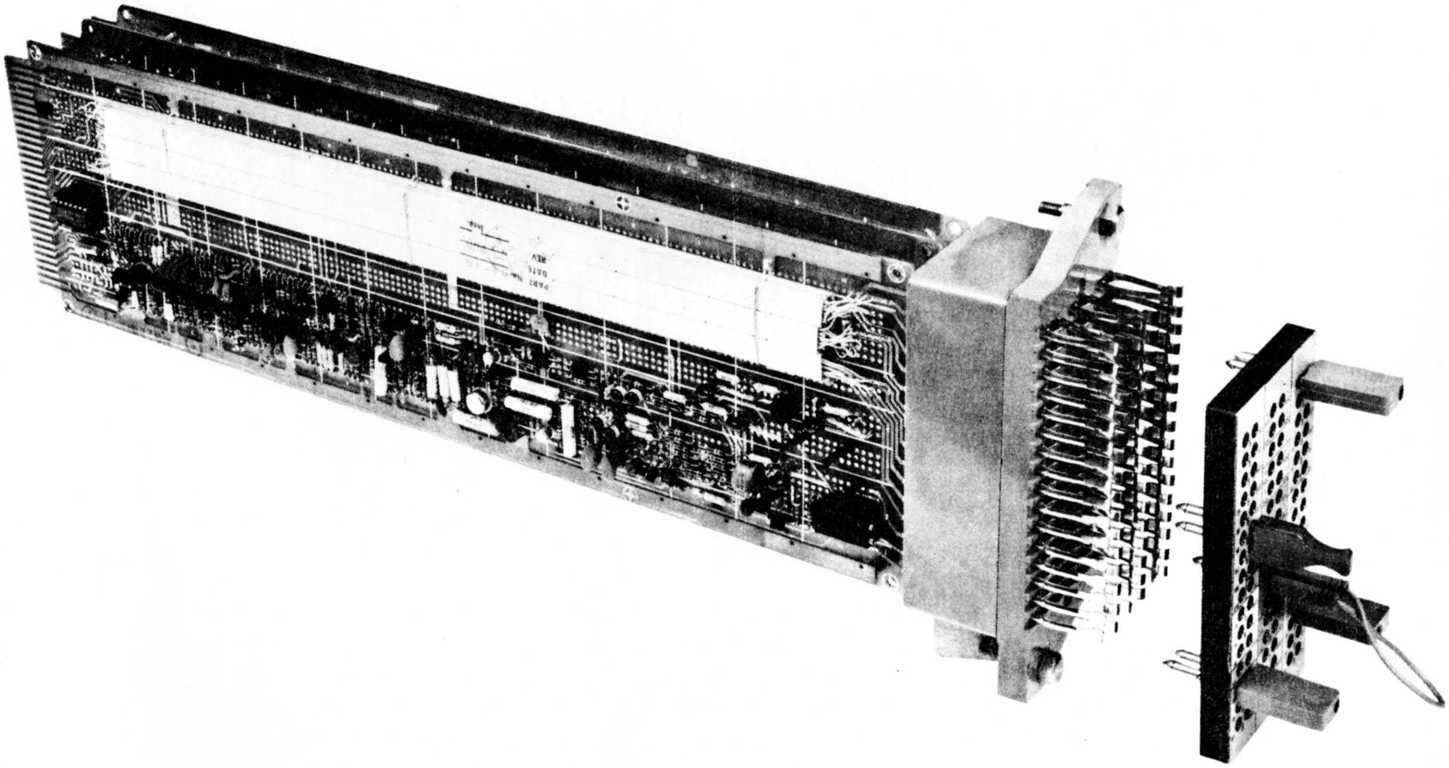


Figure 1-4 Patchboard Module, Patchbay Pin Block, and Circuit Cards

CHAPTER 2

SYSTEM ORGANIZATION

TABLE OF CONTENTS

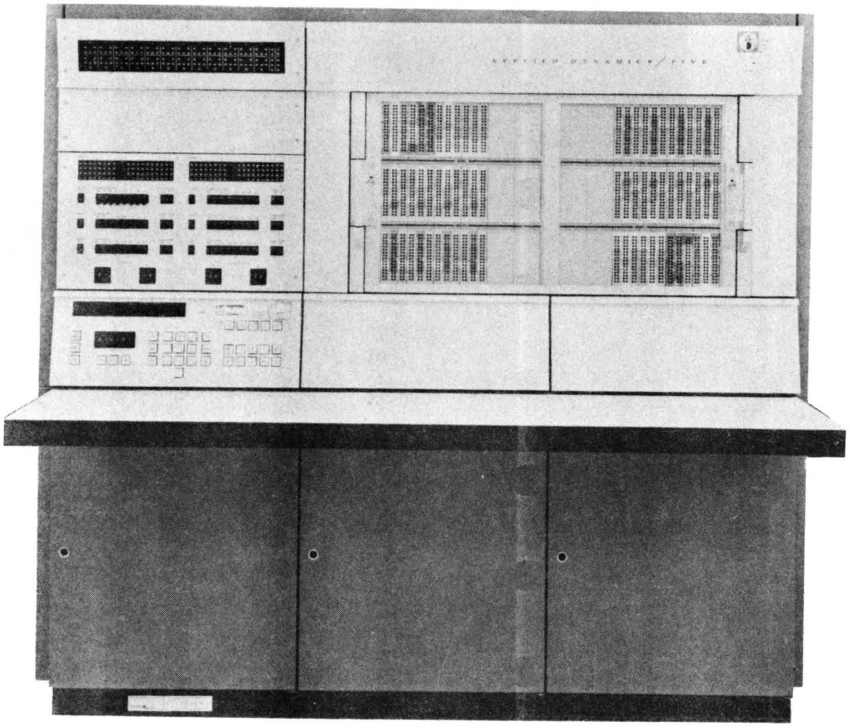
ILLUSTRATIONS

2.0 SYSTEM ORGANIZATION

The AD/FIVE is a sophisticated computer designed to be used either as a self-contained general purpose analog computer or as an integral part of a hybrid computer system. A versatile logic system for control of the computer and potential for considerable expansion of the analog and logic computing components are characteristics of the AD/FIVE. Chapter 2 describes the organization of the AD/Five as a basic analog computer. Hybrid operation of the AD/FIVE in an expanded system is dealt with in Volume 2 of this manual.

2.1 PHYSICAL ORGANIZATION

2.1.1 General Description

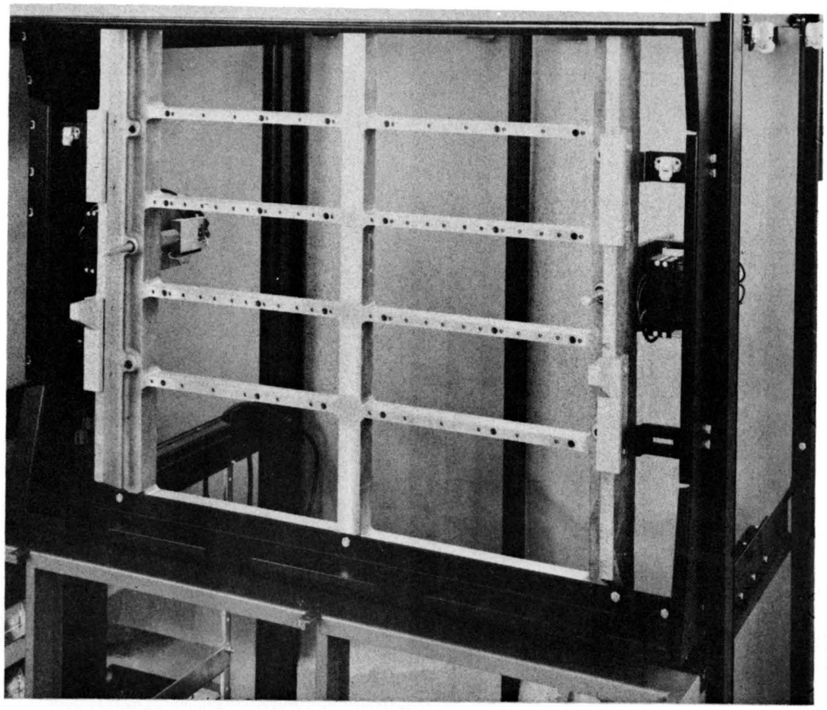


AD/FIVE Console

Figure 2-1

The AD/FIVE Console is shown in Figure 2-1.

The cabinetry of the AD/FIVE is carefully designed for strength and efficiency, with rigid steel frame construction and mounted on large, heavy-duty castors which allow ease of movement. A rugged cast-aluminum patchbay frame provides excellent structural integrity and insures proper alignment of parts in the patchbay area. The front of the cabinet showing the patchbay frame is illustrated in Figure 2-2.



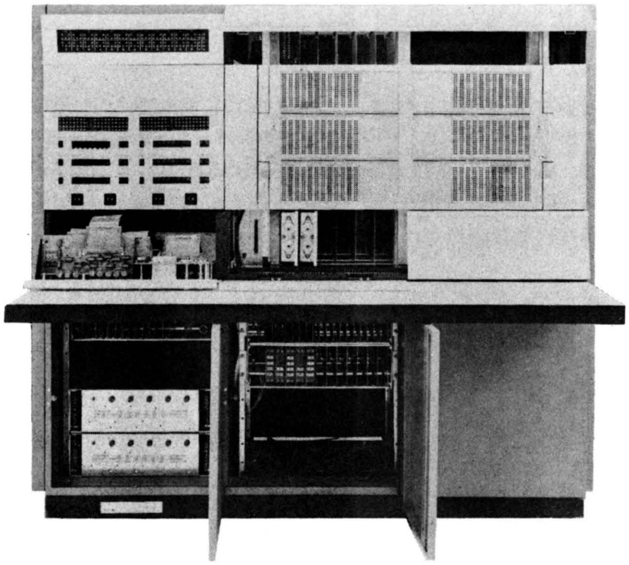
AD/FIVE Frame and Patchbay Casting

Figure 2-2

The patchboard connections are vital to the proper operation of an analog computer. The AD/FIVE is equipped with patchboard motor drive mechanisms to insure uniform movement of the patchboard. Regardless of the number of patchbay blocks in the patchbay frame, perfect connections are consistently assured through the action of the drive mechanisms.

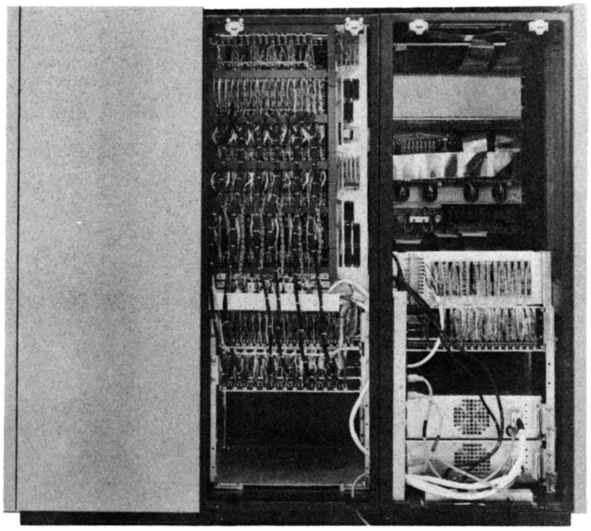
A work shelf for the convenience of the operator is mounted across the front of the AD/FIVE. This work shelf can be easily removed to allow the computer to pass through a thirty-inch doorway.

Ease of maintenance is a prime feature in the design of the AD/FIVE. Access to all parts of the computer is easily gained through the side and rear panels which are held in place by snap-in latches. No tools are required for the removal of these panels. The area below the work shelf in the front of the computer is provided with hinged doors for convenient entry. The sloping control panel and the panels above and below the patchbay are all hinged to allow access to the controls and to the components mounted above and below the patchbay.



AD/FIVE Front Access

Figure 2-3



AD/FIVE Rear Access

Figure 2-4

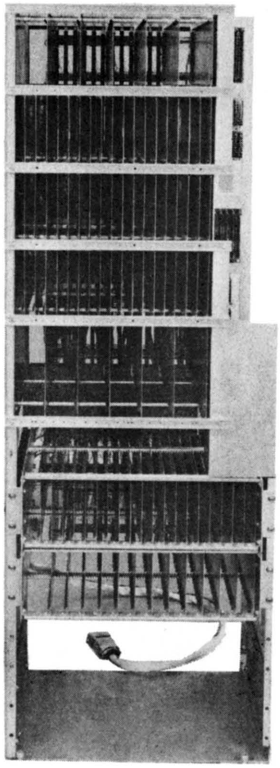
2.1.2 Modular Construction

It is a feature of analog computation that each class of problems often requires a particular combination of computing elements which may not be well suited to the solution of other classes of problems.

Aerospace applications, for example, require a significant complement of multipliers, sin/cos generators, and variable diode function generators.

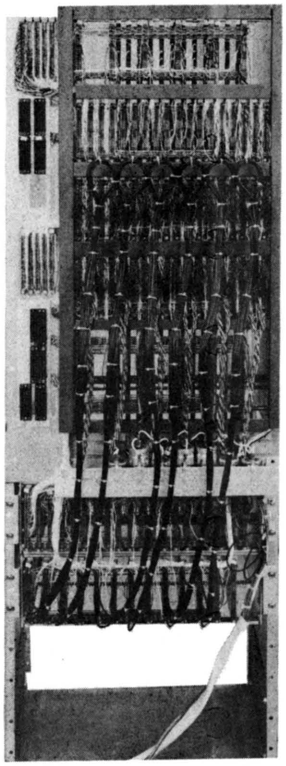
Chemical kinetics, however, require an abundance of integrators and log generators, A successful design of an analog computer must take this feature into account in order to provide useful service in a wide range of applications. Flexibility of configuration is therefore of prime importance in the design of the AD/FIVE.

An analog computer must also be able to grow along with the growing needs of its user. In order for a computer to be of maximum service to the user, expansion of its capabilities must be easily and efficiently achieved, without the need for extensive rewiring or soldering.



AD/FIVE Wiring Plane Assembly

Front View



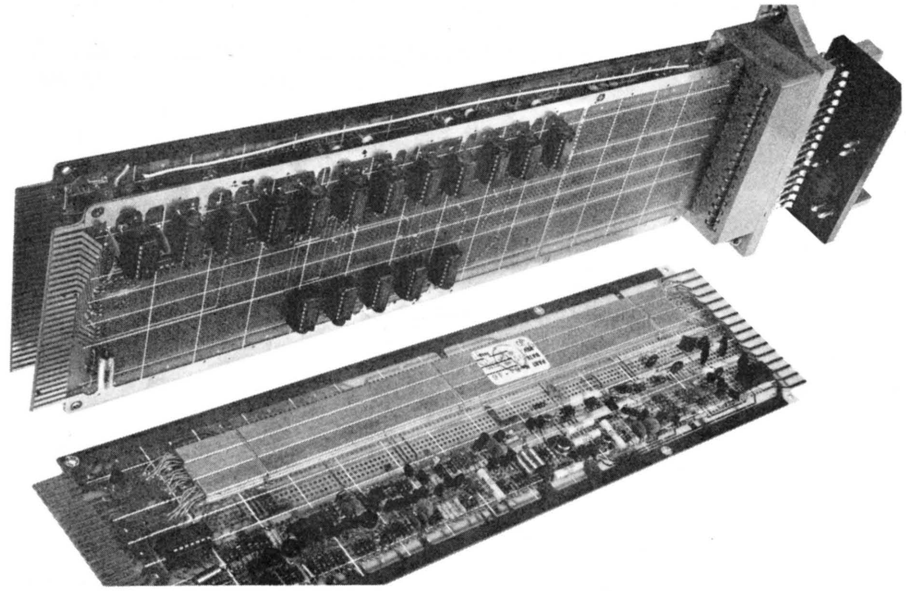
AD/FIVE Wiring Plane Assembly

Rear View

Figure 2-5

These particular needs of analog computation have resulted in the carefully designed modular construction which is a feature of the AD/FIVE. The computing components are all on circuit boards which need merely to be plugged into the wiring plane assembly in order to plan the configuration of computing components, or in order to expand an already existing system. Each AD/FIVE can therefore easily be tailored to meet the specific needs of any customer. Likewise, each AD/FIVE can be easily modified and expanded with a minimum of time and expense to keep up with the changing needs of its user. The basic AD/FIVE contains the cabinetry, control structure, and reference power supply capacity required for a fully expanded system. It also contains the power supplies and wiring plane assembly for expansion of one-half of the computer. Expansion into the second half requires the addition of a power supply unit, a wiring plane assembly, and some miscellaneous hardware and cables. No soldering, cable lacing, or sheet metal modification is required for any expansion.

The addition of computing components in areas other than directly behind the patchbay simply involves plugging the appropriate circuit card into the proper area of the wiring plane assembly. The circuit boards for computing components which are located in the patchbay behind the patchboard are plugged into the appropriate place in the wiring plane assembly and terminate directly behind the patchboard in connector blocks which contain the patchbay springs, as illustrated in Figure 2-6.

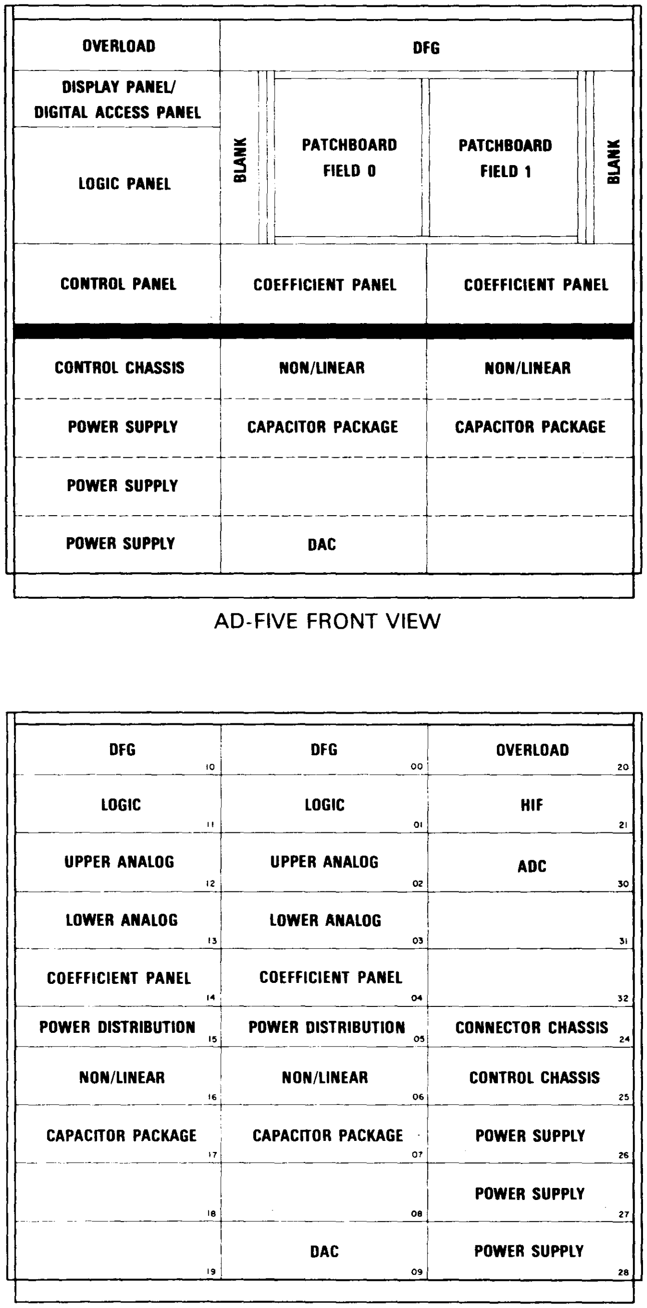


AD/FIVE Patchboard Expansion Method

Figure 2-6

2.1.3 Component Locations

Location assignments for all components are illustrated in Figure 2-7, showing both front and rear views of the AD/FIVE. For the convenience of the operator, as well as ease of maintenance, the controls, indicators, interface hardware (when present), and power supplies are all located in one bay. The other two bays are used exclusively for patchable control (such as interval timer circuitry) and computing elements.



OVERLOAD	DFG	
DISPLAY PANEL/ DIGITAL ACCESS PANEL LOGIC PANEL	BLANK PATCHBOARD FIELD 0	PATCHBOARD FIELD 1 BLANK
CONTROL PANEL	COEFFICIENT PANEL	COEFFICIENT PANEL
CONTROL CHASSIS	NON/LINEAR	NON/LINEAR
POWER SUPPLY	CAPACITOR PACKAGE	CAPACITOR PACKAGE
POWER SUPPLY		
POWER SUPPLY	DAC	

AD-FIVE FRONT VIEW

DFG 10	DFG 00	OVERLOAD 20
LOGIC 11	LOGIC 01	HIF 21
UPPER ANALOG 12	UPPER ANALOG 02	ADC 30
LOWER ANALOG 13	LOWER ANALOG 03	31
COEFFICIENT PANEL 14	COEFFICIENT PANEL 04	32
POWER DISTRIBUTION 15	POWER DISTRIBUTION 05	CONNECTOR CHASSIS 24
NON/LINEAR 16	NON/LINEAR 06	CONTROL CHASSIS 25
CAPACITOR PACKAGE 17	CAPACITOR PACKAGE 07	POWER SUPPLY 26
18	08	POWER SUPPLY 27
19	DAC 09	POWER SUPPLY 28

AD-FIVE REAR VIEW

AD/FIVE Component Locations

Figure 2-7

Each of the blocks in the AD/FIVE rear-view diagram has a number in the lower right-hand corner. This number is the chassis/location address. These addresses are used in the maintenance manuals, wiring lists, etc.

2.1.4 Computing Components

Patchable computing components are available in the following types for the AD/FIVE:

Summer/Integrator/High Gain Amplifiers
Summer/High Gain Amplifiers
Inverter/High Gain Amplifiers
Switch Amplifiers
Limiter Networks
Track-Store Networks
Comparators
Multipliers
Sin/Cos Function Generators
Log Function Generators
Potentiometers
Diode Function Generators (DFG's)
Flip-flops
Flip-flop Differentiators
Gates
Counters
Function Switches
Logic Switches

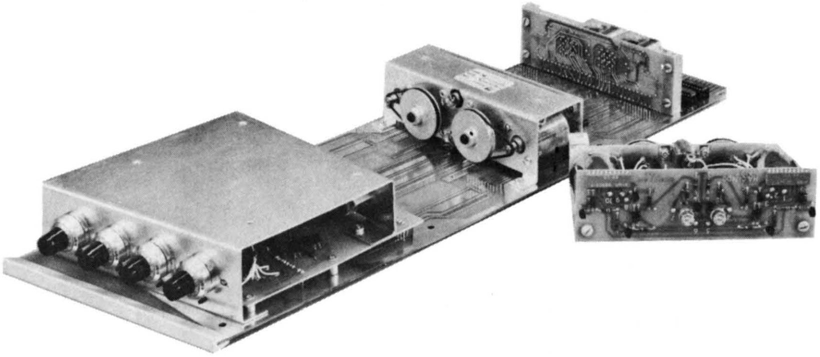
Patching configurations and operation of these elements are described in detail in Chapter 4 of this volume. Patchable controls such as the Interval Timer are described in Chapter 3.

LOGIC computing components and the circuitry for the PATCHABLE CONTROLS are located in the area behind the upper third of the patchboard.

ANALOG and ANALOG/LOGIC computing components are located in the remaining area of the patchbay. (Analog/Logic computing components are those elements which have both analog and logic signal terminations on the patchboard, such as comparators.)

VARIABLE DIODE FUNCTION GENERATORS are located in the area above the patchboard. This area can be used for a variety of other computing components as desired, such as multipliers or other fixed function generators.

COEFFICIENT POTENTIOMETERS and AMPLIFIER BALANCE POTENTIOMETERS are located in the area immediately below the patchboard behind the hinged door. Both hand-set and servo-set potentiometers are available. Each servo-set potentiometer has its own individual drive motor. Figure 2-8 illustrates the coefficient potentiometer assembly.



AD/FIVE Potentiometer Module

Figure 2-8

The hand-set potentiometer module contains four potentiometers; the servo-set potentiometer module contains two potentiometers. The arm of each coefficient potentiometer is protected by an incandescent lamp which acts as a nonlinear resistance to prevent damage to the potentiometer in the case of a patching error.

TRUNKS, DISPLAY LINES, and other lines used for communication between the patchbay and external devices are terminated in connectors located on the rear of the wiring plane assembly. (See Figure 2-5.)

The CONTROL BAY provides the major communication link between the operator and the AD/FIVE.

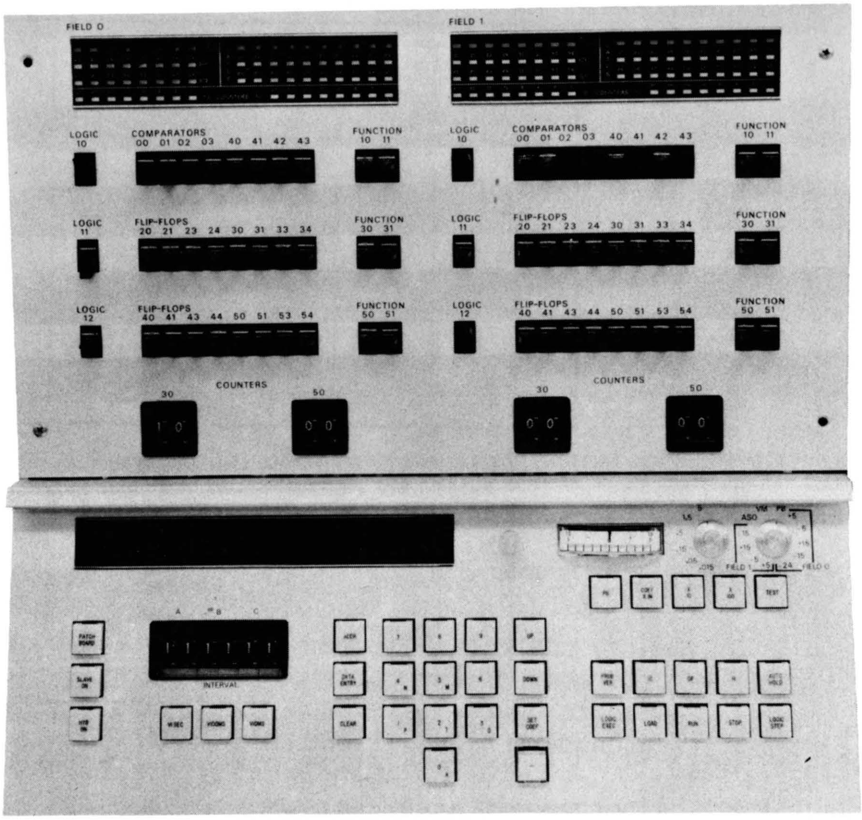
The OVERLOAD PANEL at the top of the control bay contains the overload indicators for amplifiers and function generators.

The DISPLAY PANEL is immediately below the overload panel (See Figure 2-7). This area, 5 1/4" high, can be used for any of a variety of functions. In a hybrid system this area is often used for mounting the DIGITAL ACCESS PANEL, which provides for manual monitoring and control of the hybrid interface.

The LOGIC PANEL contains the logic state indicators for the patchable gates, flip-flops, comparators, and counters; the logic and function switches; the control switches for comparators and flip-flops; and the precount thumbwheel switches for the counters.

The CONTROL PANEL contains the pushbutton controls and readout indicators for the ADDRESS SELECTOR SYSTEM, the DATA REGISTER, and the DIGITAL RATIOMETER (DRM); the INTERVAL TIMER controls; the ANALOG VOLTMETER; and the various pushbutton controls for analog and logic modes, time scales, etc.

The Logic Panel and Control Panel are shown in Figure 2-9.


FIELD 0
FIELD 1
LOGIC 10
COMPARATORS 00 01 02 03 40 41 42 43
FUNCTION 10 11
LOGIC 11
FLIP-FLOPS 20 21 23 24 30 31 33 34
FUNCTION 30 31
LOGIC 12
FLIP-FLOPS 40 41 43 44 50 51 53 54
FUNCTION 50 51
COUNTERS
30
50
A
B
C
INTERVAL

AD/FIVE Logic Panel and Control Panel

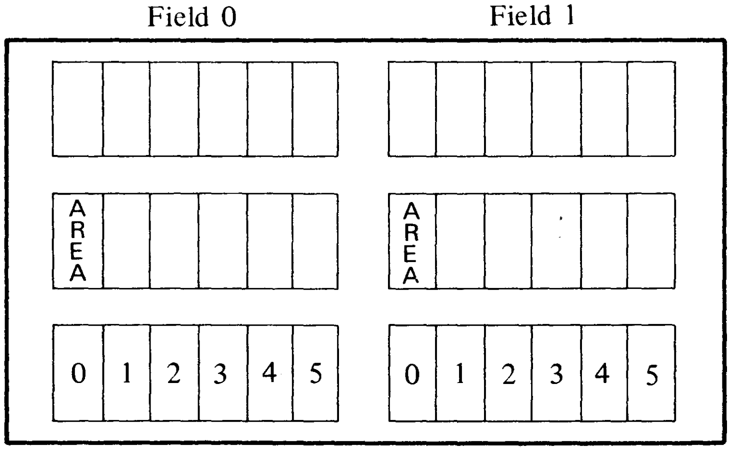
Figure 2-9

2.1.5 Fields and Areas

The AD/FIVE is divided into three bays. One bay houses the control components and power supplies. The other two bays house the patchable computing components, with a wiring plane assembly and power supply required to incorporate the computing elements into each bay.

Each of these two computing bays is called a "FIELD" as shown in Figure 2-7, and is numbered 0 or 1 from the operator's left to right. The Logic Panel (Figure 2-9) and the Overload Indicator Panel (Figure 2-13) are also divided into two fields to reflect the division of the computing components in the computing bays.

The patchboard is clearly divided into two fields, one in each bay. Each patchboard field is further divided into six "AREAS" numbered 0 through 5 as illustrated in Figure 2-10. Each area extends from the upper third of the patchboard, which contains the logic computing components, through the lower two thirds of the patchboard



AD/FIVE Patchboard Fields and Areas

Figure 2-10

which contain the upper and lower analog computing components. Each area of the patchboard contains three double vertical rows of patchboard holes from the top to the bottom of the board. Each patchboard block is one area wide.

A patchboard area can contain a maximum of ten components of a given type (i.e., ten addressable amplifiers, ten coefficient devices, etc.). The components are numbered from 0 through 9 for each type of component. The labelling on each patchboard block includes the component number for each computing component which can terminate in that block. Chapter 4 of this manual gives detailed descriptions of each of the various available patchboard blocks for the AD/FIVE.

A component location is specifically defined by a four digit address consisting of its component type (represented by a letter of the alphabet), and of its field, area, and individual component numbers. This is the method of location address used in the operation of the AD/FIVE. A detailed explanation is given in a following section of this chapter (The Address System Controls and Display, 2.2.7.1). By means of this class-field-area-component number designation the operator may rapidly and accurately locate any of the computing components.

2.1.6 Power Switch and Patchboard Drive Control

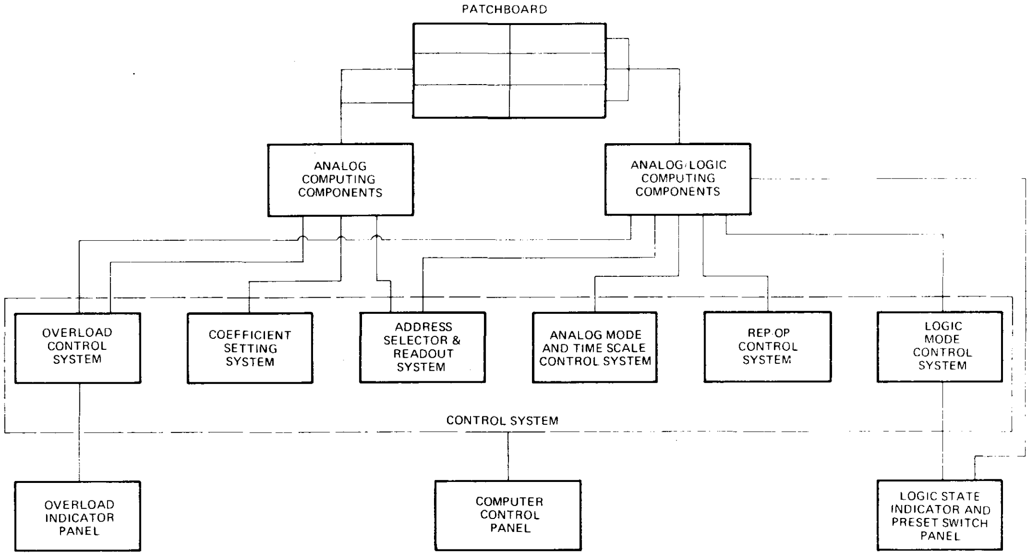
The power on/off switch and the patchboard motor drive control are two important controls for the operator. The power on/off switch is located behind the hinged door of the pot panel in field 0 (See Figure 2-7), at the operator's far left. When power is applied to the AD/FIVE various control panel indicators are illuminated. The power supply voltages can be monitored via the analog voltmeter on the control panel.

The patchboard motor drive control is a pushbutton labelled PATCHBOARD, and is located on the control panel at the operator's far left. This is a momentary contact control: it is only necessary to depress the pushbutton until the patchboard begins to move. The patchboard motor drive will then complete the cycle initiated by the depressing of the pushbutton, i.e., if the patchboard is disengaged, depressing the pushbutton will cause the motors to engage the patchboard, and if the patchboard is engaged, the pushbutton will cause the motors to disengage the patchboard. The patchboard motor drive mechanism contains limit switches to stop the motors when the patchboard is properly engaged or disengaged.

2.2 FUNCTIONAL ORGANIZATION

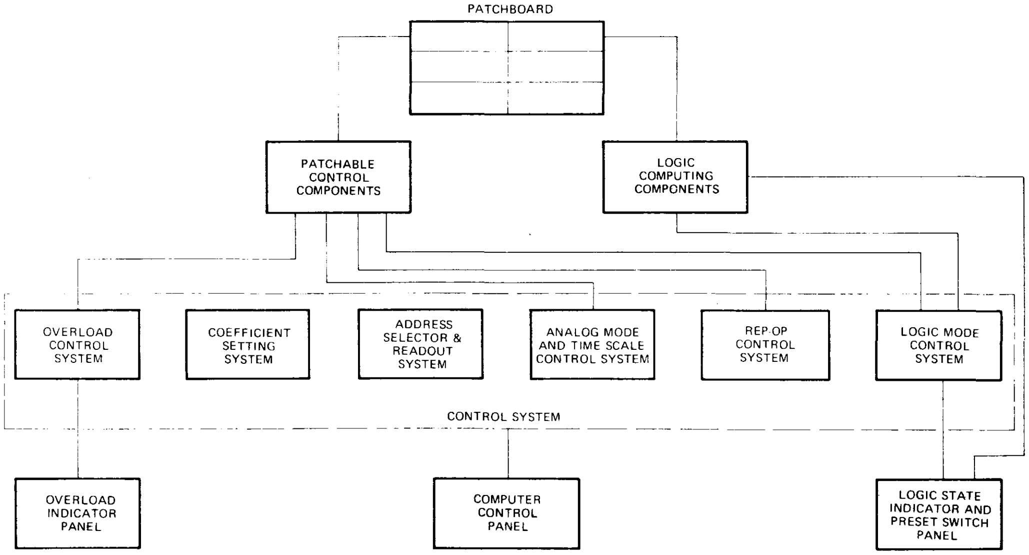
2.2.1 General Description

The functional organization of the AD/FIVE is shown in Figure 2-11 and Figure 2-12.


PATCHBOARD
ANALOG COMPUTING COMPONENTS
ANALOG/LOGIC COMPUTING COMPONENTS
OVERLOAD CONTROL SYSTEM
COEFFICIENT SETTING SYSTEM
ADDRESS SELECTOR & READOUT SYSTEM
ANALOG MODE AND TIME SCALE CONTROL SYSTEM
REP OP CONTROL SYSTEM
LOGIC MODE CONTROL SYSTEM
CONTROL SYSTEM
OVERLOAD INDICATOR PANEL
COMPUTER CONTROL PANEL
LOGIC STATE INDICATOR AND PRESET SWITCH PANEL

Functional Diagram for Analog and Analog/Logic Components

Figure 2-11


PATCHBOARD
PATCHABLE CONTROL COMPONENTS
LOGIC COMPUTING COMPONENTS
OVERLOAD CONTROL SYSTEM
COEFFICIENT SETTING SYSTEM
ADDRESS SELECTOR & READOUT SYSTEM
ANALOG MODE AND TIME SCALE CONTROL SYSTEM
REP OP CONTROL SYSTEM
LOGIC MODE CONTROL SYSTEM
CONTROL SYSTEM
OVERLOAD INDICATOR PANEL
COMPUTER CONTROL PANEL
LOGIC STATE INDICATOR AND PRESET SWITCH PANEL

Functional Diagram for Logic and Patchable Control Components

Figure 2-12

2.2.2 Computing Components

The computing components in the AD/FIVE are of three types as indicated in the functional organization diagrams. The ANALOG COMPUTING COMPONENTS have only analog signals as inputs and outputs and have no terminations in the upper one-third of the patchboard. The analog computing components are:

Summers
Inverters
Coefficient devices
Limiters
Diode Function Generators
Multipliers
Sin/Cos Generators
Log Generators

The LOGIC COMPUTING COMPONENTS have only logic signals as inputs and outputs and are terminated exclusively in the upper one-third of the patchboard. The logic computing components are:

Gates
Flip-flops
Flip-flop Differentiators
Variable Carry-Out BCD Counters
Multiple Carry-Out BCD Counters

The ANALOG/LOGIC COMPUTING COMPONENTS have both analog and logic signal terminations on the patchboard. The analog/logic computing components are:

Integrators
Comparators
Switch Amplifiers
Function Switches
Track-Store Networks

The various computing components and their patching configurations are discussed in Chapter 4 of this manual.

2.2.3 Patchable Control Components

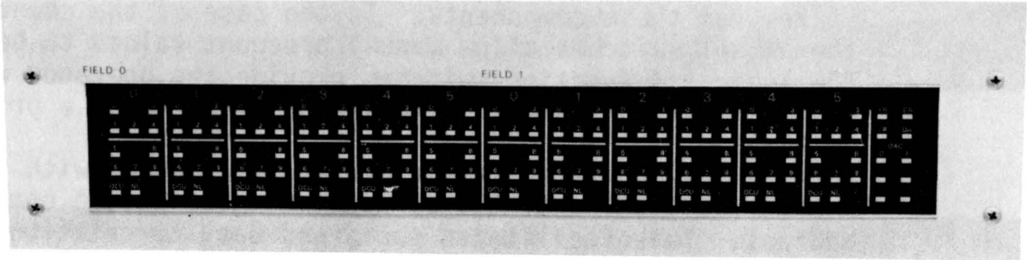
In addition to the computing components listed above, the AD/FIVE contains patchable control components which are terminated in the upper one-third of the patchboard. These are:

Logic Switches
Interval Timer, Three Period
Reference Timer, Six Decade
Console Mode Control
Console Time Scale Control
Clock Step Control

Time Scale Controlled Clock Outputs

These patchable control components are discussed in detail in Chapter 3.

2.2.4 Overload System



Overload Indicator Panel

Figure 2-13

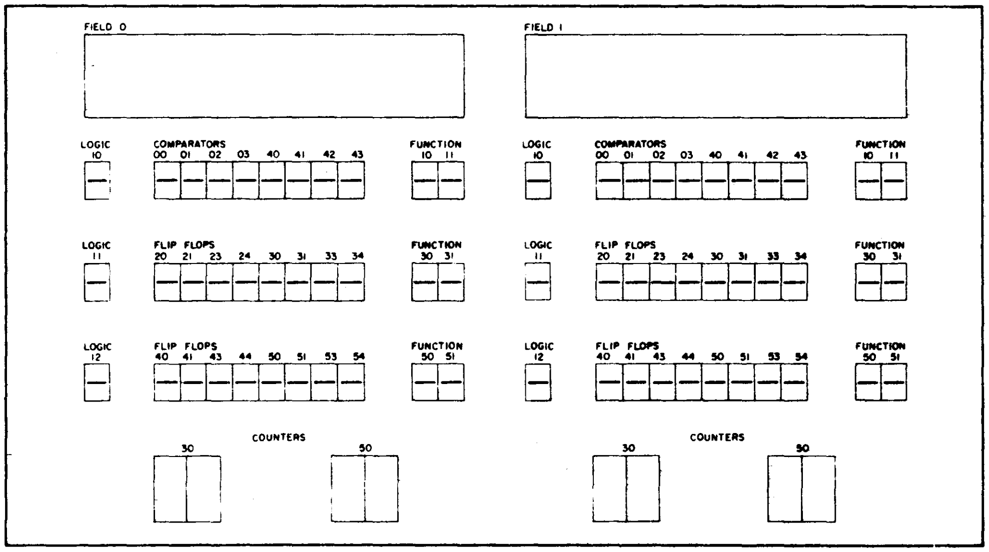
The AD/FIVE contains a high-speed overload detection system which presents both visual and logic signal indication of an overload. The OVERLOAD INDICATOR PANEL, shown in Figure 2-13, is completely wired for a fully expanded system. The indicators in the panel correspond on a one-to-one basis with the amplifiers in the lower two-thirds of the patchboard. An illumined indicator reveals an overload condition in the component to which it is assigned. Individual indicators are provided for the maximum of ten amplifiers (i.e., summer-integrators, summers, inverters, and switch amplifiers) which terminate in each area of each field. Each indicator is labelled with the number of the component to which it is assigned. One indicator, labelled NL, is provided for each area of each field to serve as a common overload indicator for the various non-linear computing devices (i.e., DFG's, multipliers, etc.) which terminate in that area. In addition, there are indicators on the right-hand side of the panel for the plus and minus reference power supplies (+R and -R), the Servo Amplifier (SA), and the Unloading Amplifier (UA).

The overload indicator drive signals are used to generate a logic signal indicating an overload condition in the computer. This signal is available to the programmer on the upper third of the patchboard in Field 0, Area 1 (See Chapter 4, Section 4.2.3). This logic signal is also used by the Auto Hold circuit for Master Hold (See Chapter 3, Sections 3.1.1.2 and 3.1.1.3).

2.2.5 The Logic Panel

State indicators and switches are provided for the logic computing components and for certain of the analog/logic computing components. These indicators and switches are located on the Logic Panel. (See Figures 2-9 and 2-14.) The indicators are illuminated when the true output of the logic computing components and the comparators is logic 1. These status indicators are quite useful to the operator, particularly during problem checkout. The toggle switches for the flip-flops and comparators allow the operator to set initial logic states for these components. In the case of the counters, the thumbwheel switches allow manual precount values to be entered. The logic and function switches provide the operator with a convenient means for controlling various aspects of a problem.

Each of the indicators and switches is labelled with the area and number designators of the corresponding computing component's address. This facilitates rapid and easy correlation of a particular indicator or switch with the patchboard location of the component with which it is associated.


FIELD 0
FIELD 1
LOGIC 10
COMPARATORS 00 01 02 03 40 41 42 43
FUNCTION 10 11
LOGIC 10
COMPARATORS 00 01 02 03 40 41 42 43
FUNCTION 10 11
LOGIC 11
FLIP FLOPS 20 21 23 24 30 31 33 34
FUNCTION 30 31
LOGIC 11
FLIP FLOPS 20 21 23 24 30 31 33 34
FUNCTION 30 31
LOGIC 12
FLIP FLOPS 40 41 43 44 50 51 53 54
FUNCTION 50 51
LOGIC 12
FLIP FLOPS 40 41 43 44 50 51 53 54
FUNCTION 50 51
COUNTERS
30
50
COUNTERS
30
50

State Indicators and Switches

Figure 2-14

2.2.5.1 Gate State Indicators

Individual indicators are provided for the thirty gates which can terminate in each field. The indicator is illuminated when the gate output is a logic 1.

2.2.5.2 Flip-flop Status Indicators and Switches

The complement of indicators and switches for each field includes sixteen indicators and sixteen single-pole triple throw spring return toggle switches for the eight flip-flop pairs which can be terminated in that field. The individual indicator lamp is illuminated when the corresponding flip-flop output is a logic 1. The toggle switch for each flip-flop serves two purposes. First, it is used to establish the initial state of the flip-flop in the LOAD mode; second, it can also be used when in the logic STOP mode to complement the output of the flip-flop. (See Chapter 4, Section 4.2.2.2 for complete details of flip-flop operation.)

2.2.5.3 BCD Counter Indicators and Switches

Indicators and thumbwheel switches are provided for the two two-decade Variable Carry-Out BCD Counters which can be terminated in each field. Each counter has eight indicator lamps to indicate the current count of the counter in BCD format. The thumbwheel switches associated with each counter permit the operator to establish a desired count for which the counter will issue a carry-out pulse. This two-decade BCD Counter, which is discussed in detail in Chapter 4, Section 4.2.2.4, counts up two decades (100 counts), produces a carry-out pulse, resets itself, and repeats the two-decade up count. The precount feature, determined by the thumbwheel switches, produces a carry-out pulse on a patchboard terminal labelled CO VAR when the desired precount is reached. This precount does not interfere with the normal two-decade counting cycle, but allows an intermediate carry-out pulse to be obtained.

2.2.5.4 Comparator Indicators and Switches

Each of the eight comparators normally contained in a fully expanded field has an indicator and a single-pole triple throw toggle switch associated with it. The indicator lamp is illuminated when the comparator output is a logic 1. The three-position toggle switch allows the operator to preset the comparator output at the beginning of a problem, thus insuring that the comparators start in specified states. See Chapter 4, Section 4.2.1.13 for comparator programming details.

2.2.5.5 Logic Switches

The Logic Panel contains three single-pole triple throw toggle switches labelled LOGIC for each field. These switches terminate in the upper third of the patchboard in area 1 of each field as the computer is expanded. See Chapter 4, Section 4.2.3.3 for details. Each switch terminates in two patchboard holes, labelled UP or DN. When the switch is in the up position, a logic 1 is available at the UP terminal; when the switch is in the down position, a logic 1 is available at the DN terminal. When the switch is in the center position, logic 0 is present at both the UP and DN terminals.

2.2.5.6 Function Switches

The Logic Panel contains six single-pole double throw toggle switches in each field labelled FUNCTION. Patchboard terminations for these switches are provided as the computer is expanded. (See Chapter 4, Section 4.2.3.3 for details on the operation of the Function switches.) Each of these switches serves a dual purpose: each is a logic switch and each provides manual control for a double-pole, double-throw (DPDT) relay. Two terminations are associated with these switches in logic area 1 of each field of the patchboard. The termination labelled FSW is an output controlled by the switch, and provides a logic 1 when the switch is in the up position. The termination labelled RLY is a logic signal override control input for control of the DPDT relay. Any logic source patched to RLY overrides the toggle switch control of the DPDT relay, and results in direct logic control of the relay. The DPDT relay terminates in areas 1, 3, and 5 of the upper analog section of the patchboard. Details of control of the DPDT relay are to be found in Chapter 4, Section 4.2.3.3.

2.2.6 The Control System

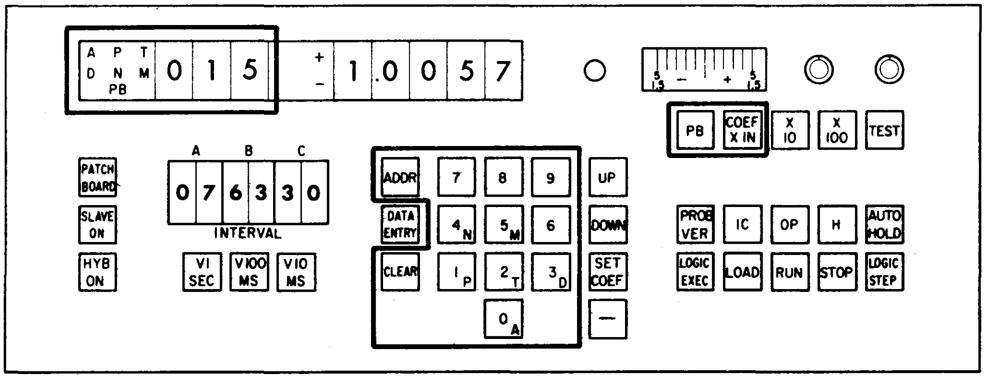
That part of the AD/FIVE denoted as the Control System in Figures 2-11 and 2-12 includes all of the computer internal control circuitry such as the logic clock, the interval and reference timers, the coefficient setting system, etc. Chapter 3 presents a complete discussion of the Control System.

2.2.7 The Control Panel

The Control Panel is the central operator interface with the AD/FIVE. This panel contains: the in-line readout display for the ADDRESS SELECTOR SYSTEM, the DIGITAL RATIOMETER (DRM), and the data register for the COEFFICIENT SETTING SYSTEM; the INTERVAL TIMER controls; the pushbutton controls for the ADDRESS SELECTOR SYSTEM, the COEFFICIENT SETTING SYSTEM, the LOGIC MODES, the ANALOG MODES and TIME SCALES; and the ANALOG VOLTMETER. A detailed description of the functions of these pushbuttons, thumbwheel switches and displays is included in Chapter 3. Only the general functions of these controls are presented in this chapter.

2.2.7.1 The Address System Controls and Display

The display indicators and pushbutton controls outlined in Figure 2-15 are associated with the Address Selector System. The purpose of the Address Selector System is to select the specific component whose output is to be monitored by the computer's Readout System. The following classes of components are addressable: Amplifiers (A), Potentiometers (P), Trunks (T), Digital-to-Analog Converters (D), Non-Linear (N) (i.e., DFG's, Multipliers, etc.), and Miscellaneous (M) (a class currently held in reserve for possible future use). An address consists of the following four parts: component class, field, area, and individual component number (i.e., P018 is the potentiometer in field 0, area 1, with the component number 8). When the pushbutton labelled ADDR is illuminated, the numerical entry group of pushbuttons is used to enter an address in the address register. The current contents of the address register are normally displayed in the left-hand half of the in-line display.



Address System Controls and Display

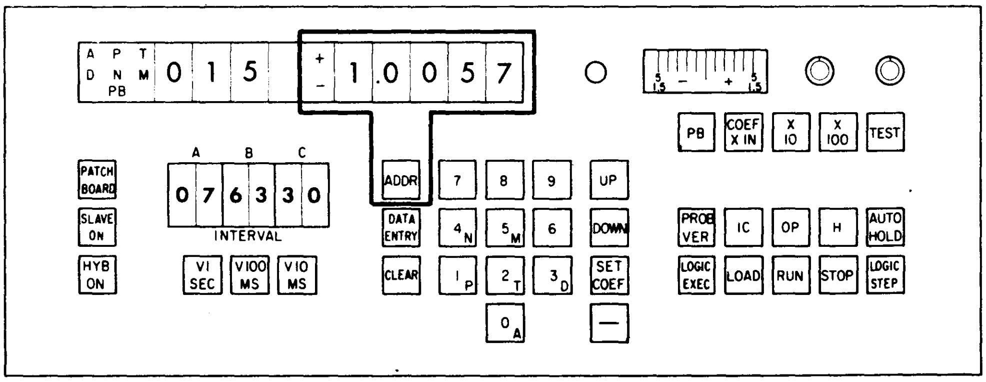
Figure 2-15

A different condition exists, however, if the pushbutton labelled PB is depressed and illuminated. In this case, the address portion of the in-line display is not illuminated except for the letters PB, and the input to the Readout System comes from a patchboard input termination labelled PB.

The CLEAR pushbutton, if depressed while the ADDR pushbutton is illuminated, will clear the address register to the address A000, i.e., Amplifier 000. It will also return the hardware pointer for the sequential entry address register to the left-most (class) entry designator, so that the address register is set to accept a new address from the numerical entry pushbuttons. See Section 2.2.7.3 of this chapter for the function of the CLEAR pushbutton when the DATA ENTRY pushbutton is depressed, instead of the ADDR pushbutton.

If a coefficient device is addressed, the input to that device is removed from its patchboard input termination and is connected to the computer reference voltage so that the coefficient setting is monitored. However, if the COEF XIN pushbutton is depressed, then the input to the coefficient device remains connected to its patchboard input termination and the value of the problem variable at the output of the coefficient device is monitored.

2.2.7.2 Readout System



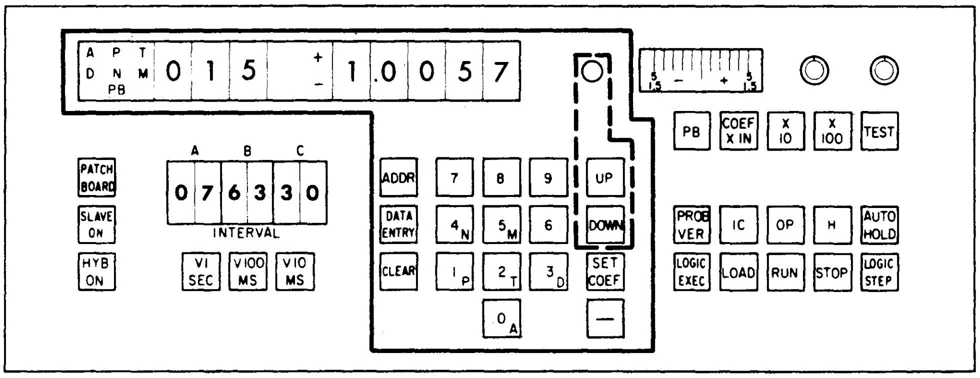
Readout System Displays and Controls

Fiqure 2-16

The AD/FIVE Readout System consists of an Unloading Amplifier (UA), a Digital Ratiometer (DRM), and in-line display indicators. The Unloading Amplifier is a very high input impedance device which is used to eliminate errors caused by loading effects that might otherwise occur if a component such as a potentiometer is monitored directly by the DRM. The DRM provides as its output the ratio of the voltage being monitored to the computer reference voltage. This ratio is displayed by the right-hand section of the in-line display whenever the ADDR pushbutton is illuminated.

2.2.7.3 Coefficient Setting System

Figure 2-17 illustrates the display indicators and controls which are used in setting or changing a coefficient.



Coefficient Setting System Displays and Controls

Fiqure 2-17

The procedure for setting a coefficient is as follows: 1. The coefficient device to be set is addressed through the Address Selector System. 2. The DATA ENTRY pushbutton is depressed and illuminated. (Depressing the DATA ENTRY pushbutton accomplishes two things: the right hand group of display indicators is transferred from the DRM output to the data register output so that the operator can monitor the contents of the data register, and the numerical entry group of pushbuttons is transferred from the address register function to the data register function so that the operator can enter the value of the desired coefficient.) 3. The numerical entry pushbuttons are depressed sequentially to enter the value of the desired coefficient from left to right in the data register display windows. If a wrong pushbutton is depressed during data entry, pushing the CLEAR button will clear the data register to 0.0000, and the desired coefficient may be correctly reentered. 4. The SET COEF pushbutton is depressed, and remains illuminated for the time during which the coefficient is being set (approximately 1 second); pushing the SET COEF pushbutton automatically sets the coefficient device to the value entered in the data register. The desired coefficient is now set.

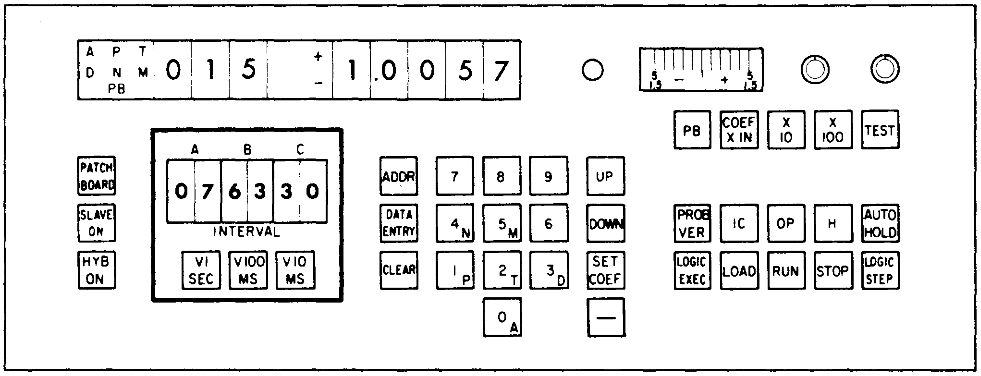
The minus (-) entry pushbutton located beneath the SET COEF pushbutton is used in the setting of DAC's and four-quadrant DCU's. See the hybrid reference manual for details of DAC and DCU setting.

At the end of the setting operation the DATA ENTRY pushbutton light is extinguished and the ADDR button becomes illuminated; the right-hand section of the indicator display is automatically returned to the DRM output so that the operator can verify the setting of the coefficient device, and the numerical entry pushbuttons are automatically returned to the address selector function, so that a new address may be selected as desired. Note that the output from a coefficient device must be patched to an appropriate linear load for correct setting of the desired coefficient in the problem; a coefficient device should not be set before patching the output to the desired problem input of a linear computing component.

The two pushbuttons labelled UP and DOWN contained in the section outlined by a dashed line in Figure 2-17 allow the operator to vary the coefficient setting of a servo-set potentiometer in an open loop manner. If a servo-set pot is addressed, depressing the UP pushbutton will increase the value of the coefficient in a continuous manner, and depressing the DOWN pushbutton will decrease the value. The slew rate can be controlled by the potentiometer located above the UP and DOWN pushbuttons on the control panel.

If a potentiometer is addressed, the input to the pot is switched from the patchboard input termination to computer reference, and the indicator display shows the setting of the potentiometer with respect to the computer reference voltage. Depressing the COEF XIN pushbutton, however, causes the input to the coefficient device to remain connected to the patchboard input termination, and the indicator display then reads the value of the coefficient times the input; i.e., the value of the problem variable at the output of the coefficient device is monitored. Using the UP and DOWN pushbuttons with COEF XIN depressed allows the operator to vary the setting of the coefficient device in a continuous manner without removing it from the computational configuration. It should be noted here that, even though COEF XIN is depressed, using the DATA ENTRY and SET COEF pushbuttons will cause the coefficient device to be removed from its patched input for the time during which the coefficient is being set and the SET COEF pushbutton is illuminated, thus in effect removing the device from the computational configuration. This does not occur when the UP and DOWN buttons are used with COEF XIN depressed. This permits the operator to conveniently vary a parameter or coefficient during a problem solution.

2.2.7.4 The Interval Timer Controls

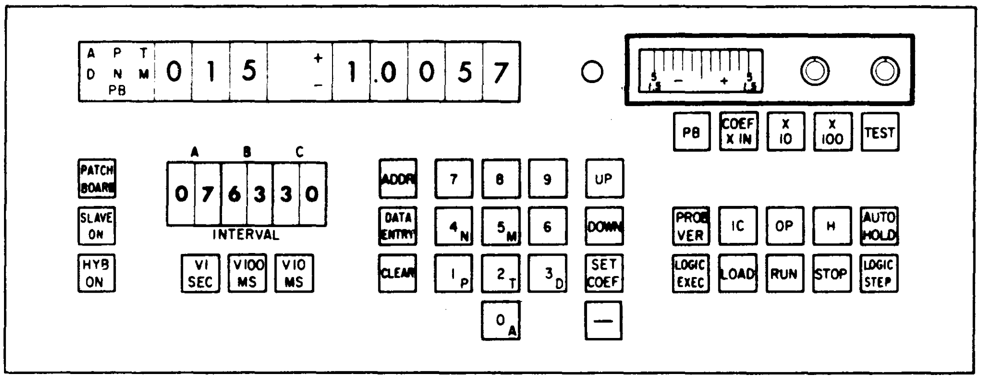


Interval Timer Controls

Figure 2-18

The pushbuttons and thumbwheel switches which are used to control the Interval Timer are illustrated in Figure 2-18. The Interval Timer provides three sequential timing intervals for cyclic control of the computer analog modes when the LOGIC EXEC pushbutton is depressed. Interval A corresponds to initial condition, B to operate, and C to hold. Each interval duration is set with a two-decade thumbwheel switch. The three pushbuttons below the Interval Timer set the decimal point for the three periods by multiplying the setting on the thumbwheel switches by 1.0, 0.1, or 0.01 seconds. The Interval Timer outputs are terminated on the patchboard in field 0, logic area 1, and are accompanied by control terminations. The outputs of the Interval Timer are also internally connected to the computer analog mode control buses, thus allowing repetitive operation if desired. The Interval Timer is also very useful as a ring counter with a carry-out pulse generated every time the timer changes state. Details of the Interval Timer and its operation are presented in Chapter 3, Section 3.1.4, and Chapter 4, Section 4.2.2.6.

2.2.7.5 The Analog Voltmeter

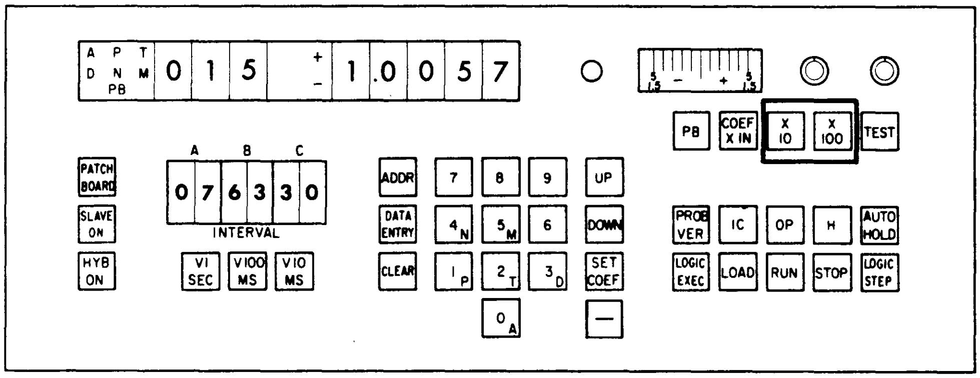


The Analog Voltmeter

Figure 2-19

The Analog Voltmeter and its controls are shown in Figure 2-19. One rotary switch allows a variety of scale selections for the voltmeter. The other rotary switch selects the input to the voltmeter. The choice of inputs includes the Address Selector Output (ASO), two patchboard terminations labelled PB and VM, and the outputs of the various power supplies. When ASO is selected, the voltmeter is connected to the output of the Unloading Amplifier and reads the value at the point addressed.

2.2.7.6 Time Scale Controls



Time Scale Controls

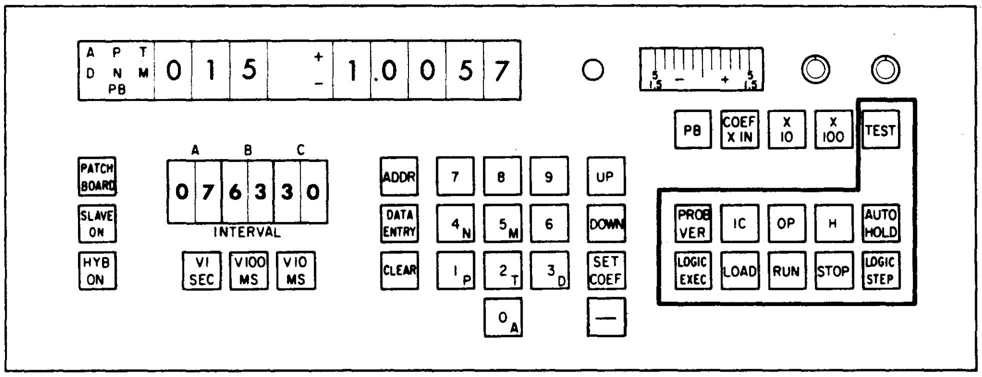
Figure 2-20

The two pushbuttons indicated in Figure 2-20 determine the computer time scale. Four time scales are available with the use of these pushbuttons. These are:

X1 If neither pushbutton is depressed and illuminated;
X10 If the X10 pushbutton is depressed and illuminated;
X100 If the X100 pushbutton is depressed and illuminated;
X1000 If both the X10 and X100 pushbutton are depressed and illuminated.

These pushbuttons control the integrator time constants and allow the operator to speed up or slow down the problem solution. The pushbuttons also control the selection of clock subfrequencies so that the Interval Timer and the logic computing components may also be appropriately time scaled.

2.2.7.7 Mode Control Pushbuttons



Mode Control Pushbuttons

Figure 2-21

The group of pushbuttons outlined in Figure 2-21 are associated with the computer mode control. The three pushbuttons labelled IC, OP, and H activate the three normal analog operational modes of Initial Condition, Operate, and Hold. These controls do not affect the logic system. The three pushbuttons just below the analog mode controls labelled LOAD, RUN, and STOP are the logic mode control pushbuttons, and control the clocking signals and the logic. In the LOAD mode, all logic components are set to their initial states or cleared when it is appropriate. In the RUN mode, clocking signals are applied to all logic components as patched. The STOP mode interrupts the clocking signals and freezes the logic in its current state.

The LOGIC EXEC pushbutton completely disables the analog mode pushbuttons and transfers the console analog mode control buses to Interval Timer control. Thus the time in IC, OP, and H is determined by the thumbwheel switches and V pushbuttons of the Interval Timer. The logic mode control pushbuttons can provide complete control of repetitive operation without any special patching while LOGIC EXEC is depressed and illuminated.

The LOGIC STEP pushbutton is operative when the logic mode is STOP, and allows advance of the logic program by a predetermined number of clock pulses. See Chapter 3, Sections 3.1.2.4 and 3.1.2.5 for details on the use of the LOGIC STEP pushbutton.

AUTO HOLD activates automatic hold of all integrators if any analog component overloads. This high-speed feature is most helpful in problem checkout. See Chapter 3, Sections 3.1.1.2 and 3.1.1.3.

The PROB VER pushbutton controls relays that disable certain patchboard reference terminals and activate others, allowing test initial conditions to be applied to integrators for problem verification. See Chapter 3, Section 3.1.1.4.

If the TEST pushbutton is depressed when the analog mode is IC, an addressed integrator is automatically converted to a summer whose output is proportional to the integrator's initial derivative input current. This feature is useful in the static check-out of a problem. See Chapter 3, Section 3.1.6.5 for details.

2.2.7.8 Miscellaneous Controls

Three pushbuttons on the left-hand side of the control panel remain. The PATCHBOARD pushbutton activates the patchboard motor drive mechanism. The SLAVE ON pushbutton provides an enable signal to computer control circuits to allow some of the AD/FIVE controls to be operated remotely from another computer. The HYB ON pushbutton provides an enable signal to the hybrid interface to allow control of the AD/FIVE from the digital computer in a hybrid computer system.

CHAPTER 3

THE CONTROL SYSTEM

CHAPTER THREE

THE CONTROL SYSTEM

Table of Contents

CHAPTER THREE

THE CONTROL SYSTEM

List of Illustrations

3.0 THE CONTROL SYSTEM

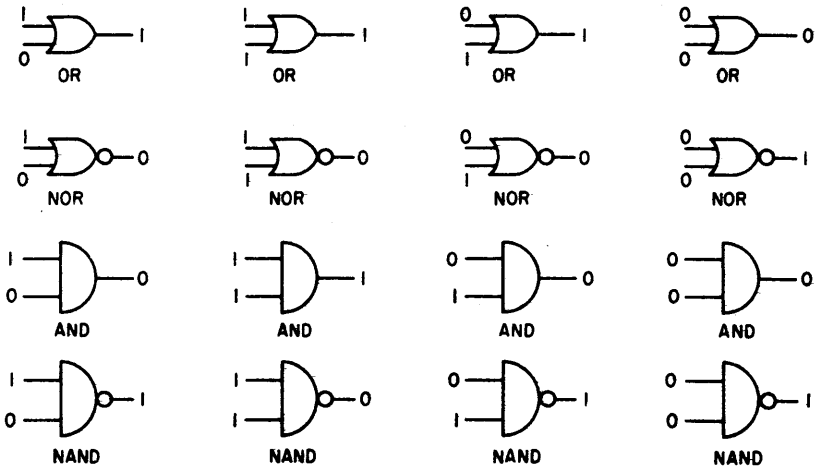
The AD/FIVE control system was designed to realize three primary objectives:

1. Simplicity and convenience of operation;
2. Flexibility in the use of components;
3. Ease of hybrid integration.

Simplicity and flexibility may appear to be contradictory goals, since an increase in flexibility makes more choices available to the operator, which requires the operator to become familiar with a wider range of possibilities and to be able to make decisions based on this wider range. The Applied Dynamics philosophy of CONTROL BY EXCEPTION resolves this apparent contradiction. Complete control of the AD/FIVE may be exercised from the control panel pushbuttons, without the need for logic patching or a digital computer. The operator thus does not need to consider complex control features until he becomes familiar with them and desires to use them. Sophisticated control features are available and may then be implemented one by one as the problems become more complex and as the operator's familiarity with the potential of the versatile control system increases.

Ease of hybrid integration is achieved through the hybrid interface, which gives the digital computer access to all operator controls, without, however, excluding the operator from the system. The operator thus retains the ability to directly interact with a hybrid problem without having to go through the digital computer. The hybrid interface is described in detail in a separate section of the reference manual.

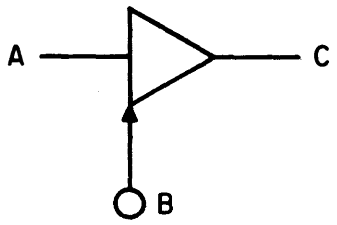
The symbols for OR/NOR and AND/NAND gates are used in this chapter. The reader should be familiar with signal steering through logic gates, and should be thoroughly familiar with the concepts of "logic one" and "logic zero". Figure 3-1 shows the relationships of logic gates and their logic inputs and outputs as used in this manual.



OR/NOR and AND/NAND Relationships

Figure 3-1

A symbol is also introduced here which is not commonly encountered except in the documentation of Applied Dynamics computers. This is the symbol for the electronic emitter-follower override, which is an improvement in cost and simplicity over the patchbay-pin-switch scheme used on some computers. The symbol is illustrated in Figure 3-2.



Emitter-Follower-Override

Figure 3-2

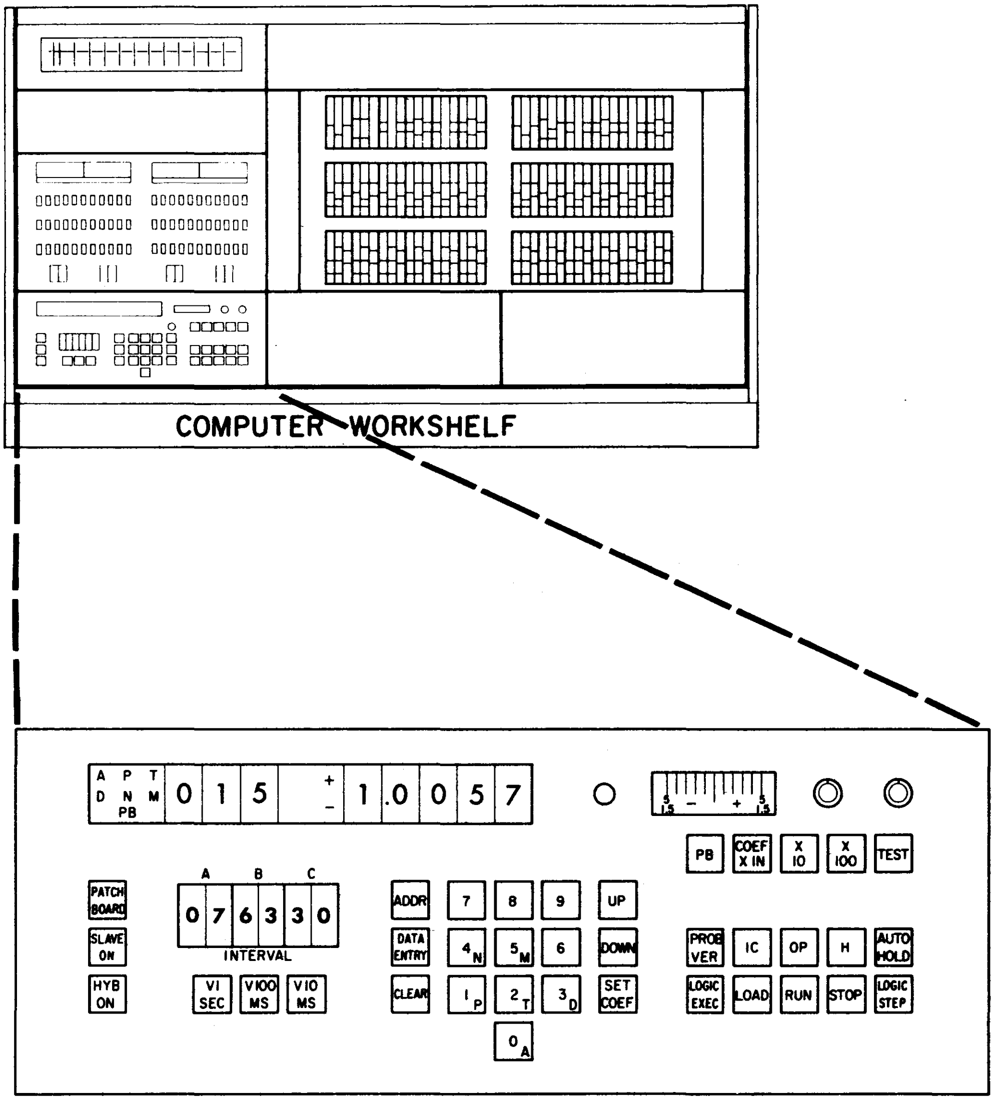
The emitter-follower-override symbol is used to represent the following control scheme; Normally a control signal from source A is delivered to destination C. If, however, a control signal is patched into the patchhole at B (from some logic output), then the signal patched at B will be delivered to destination C and signal A is ignored. The above definition is true regardless of the state (logic one or logic zero) of either A or B.

3.1 PUSHBUTTON CONTROL

The AD/FIVE operator can have complete pushbutton control of:

Analog mode
Logic mode
Time Scale
Repetitive Operation
Addressing
Coefficient Setting
Problem Checkout

No patchable logic controls are necessary to control the above functions. The control panel and pushbuttons are shown in Figure 3-3. This section describes only these control panel features. Control by exception from the logic patchboard is discussed in Section 3.2.



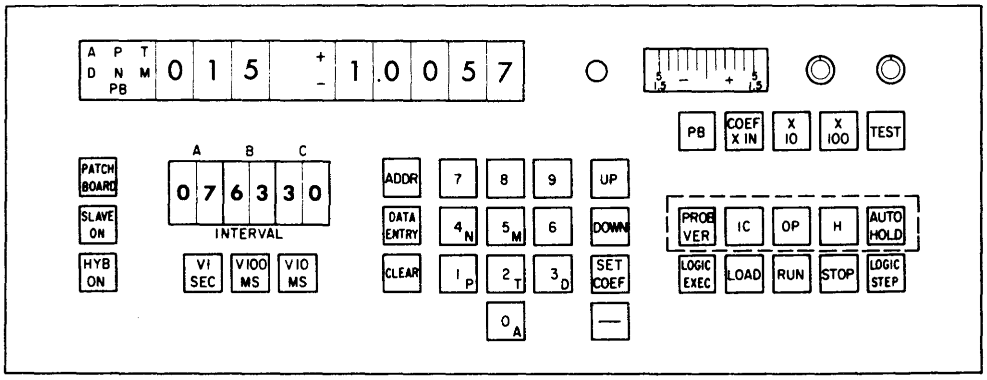
AD-FIVE CONTROL PANEL

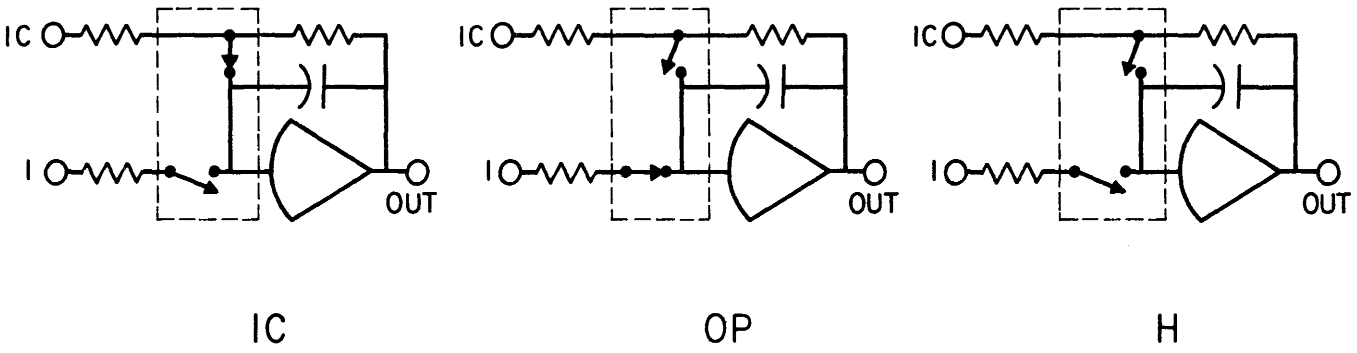
Figure 3-3

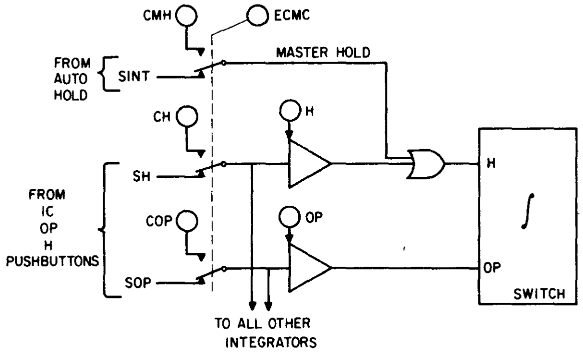
3.1.1 Analog Mode Control

3.1.1.1 Initial Condition, Operate, Hold

The AD/FIVE has three basic analog modes: Initial Condition (IC), Operate (OP), and Hold (H). These three modes correspond to the states of the integrator mode control switches and are defined schematically in Figure 3-4. Mode switching is actually done with a high speed electronic switch. All integrator mode switches are normally connected to the console mode control bus lines and are controlled by the control panel pushbuttons IC, OP, and H.







		ANALOG MODE		
		IC	OPERATE	HOLD
SW	HOLD	0	0	1
INPUTS	OP	0	1	1 or 0

The mode switches are designed to respond to the control inputs in accordance with this truth table.

Figure 3-4

3.1.1.2 Master Hold

Master Hold is an analog mode in which all integrator controls are put in hold, and cannot be overridden. Master Hold may be generated in a number of ways, and these are discussed in section 3.2 of this Chapter. Master Hold may be generated by the use of the AUTO HOLD pushbutton as described below.

3.1.1.3 Auto Hold

When the AUTO HOLD pushbutton is depressed, the analog and logic operations will run normally unless an overload occurs in any analog component. When an overload occurs, the Master Hold Bus is automatically activated, and forces all integrator mode switches to the hold position, regardless of any individually patched mode control. Use of the AUTO HOLD pushbutton will cause one of two conditions to occur in the case of an overload in an analog component:

1. If LOGIC EXEC is not depressed, then the analog operation only will freeze;
2. If LOGIC EXEC is depressed, then both the analog and the logic operations will freeze.

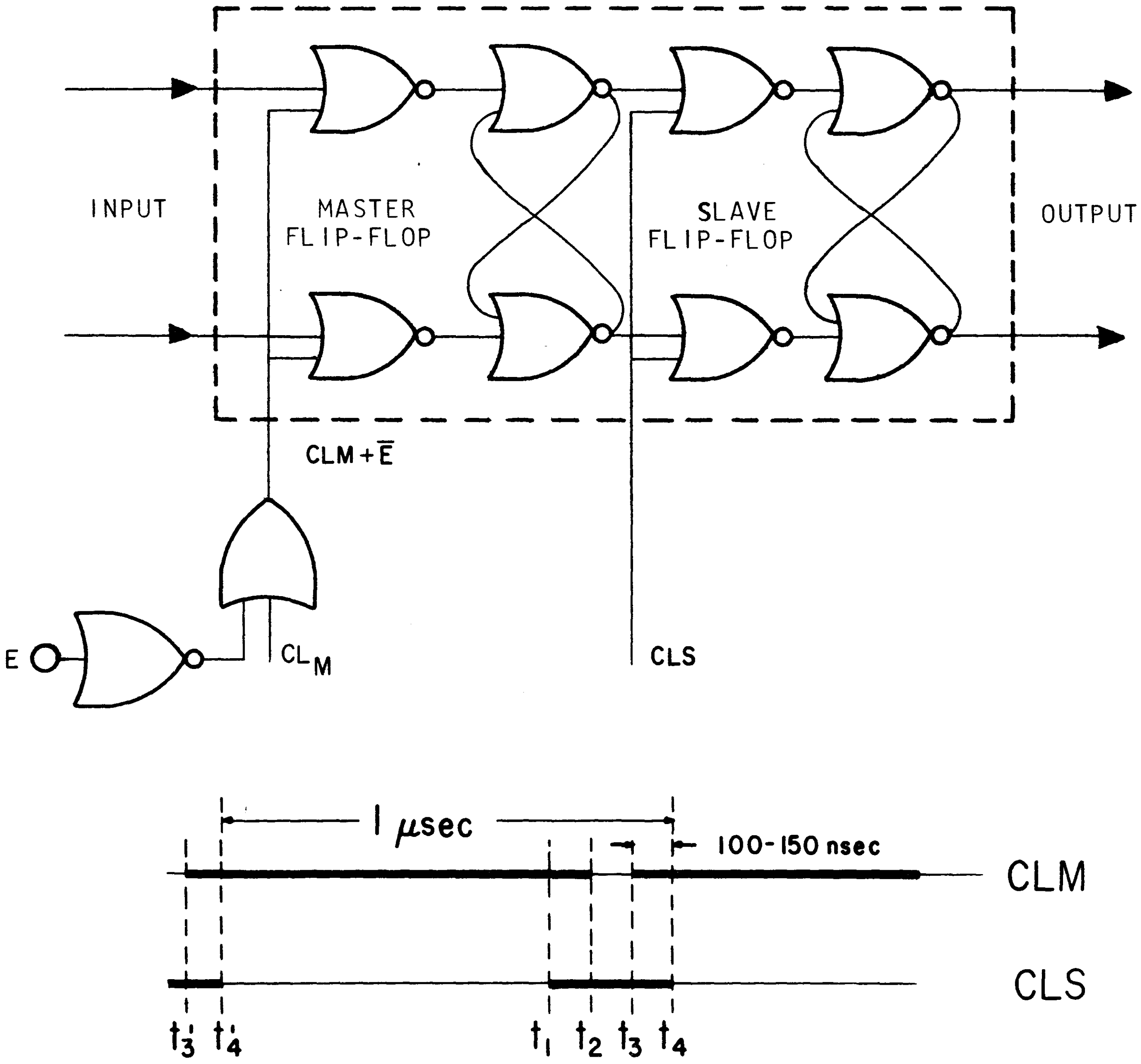
If the AUTO HOLD button is activated, only manual intervention by the operator will release it and deactivate the hold condition. When the AUTO HOLD button is desengaged by the operator, the integrator mode switches will return to the states commanded by the console mode buses before the auto hold feature was activated.

3.1.1.4 Problem Verify

The problem verify pushbutton provides a very useful relay switch controlling four paired output terminals in the analog patchboard. When the operator pushes the PROB VER pushbutton, the analog patchboard terminals labelled +TS and -TS are switched from ground to the + and - computer reference supplies and the terminals labelled +IC and -IC are connected to ground. When the operator unlatches the PROB VER pushbutton, the +IC and -IC terminals are connected to reference and the +TS and -TS terminals are grounded. These terminals are located in the upper analog portion of Field 0, Area 0.

The relay switching of these terminals is very useful in the static check of equipment in the IC mode. Since many initial conditions of integrators are zero, the operator can apply non-zero initial conditions to integrators through the use of the problem verify feature without having to patch the integrators to reference individually and then having to remove the patching after the static check is performed. The problem verify terminals can be chained out to as many inputs as necessary, and the integrator IC's can all be connected to ground or reference through the use of the PROB VER pushbutton.

Another use for the problem verify terminals can be found when it is desired to operate an integrator (or integrators) from one of two different initial conditions. Potentiometers may be patched between the problem verify terminals and the integrator IC terminals. The IC values can then be set as desired, and the use of the PROB VER pushbutton will conveniently switch from one IC to the second at will.



CL_M - Master Clock
CL_S - Slave Clock
t'_3-t_2 - Master Flip-Flop locked.
t_2-t_3 - Sampling Interval--Master Flip-Flop reads its inputs at t_3 and locks.
t_4 - Slave Flip-Flop reads Master Flip-Flop output.
t_1-t_4 - Slave Flip-Flop locked.

Simplified Flip-Flop

Figure 3-5

3.1.2 Logic Mode Control

3.1.2.1 General

Since synchronous logic is used in the AD/FIVE (i.e., all components, except the gates, require clocking pulses for operation), special clocking pulses are generated in the computer and applied automatically to the logic components by the logic mode control pushbuttons. The frequency of these pulses is 1 MHz.

3.1.2.2 Synchronous Logic

All AD/FIVE logic components, with the exception of gates, operate on two-phase clocking signals, generated from an internal square-wave oscillator. Figure 3-5 is a simplified typical flip-flop showing how the two clock signals, CLM and CLS, are used to synchronize master and slave operation. Every synchronous device has an enable (E) terminal which must be patched to an enabling signal (i.e., P signals, V signals, logic one, or the outputs of other logic components) in order to allow the device to respond to its inputs. See Figure 3-5, which clearly illustrates that when E is logic 0 the inputs to the master flip-flop are inhibited.

Each flip-flop consists of an input stage, called the Master, and an output stage, called the Slave, which are interconnected as shown in Figure 3-5. Separate locking signals are applied to each flip-flop stage: a master clock line drives the master flip-flop; a slave clock line drives the slave flip-flop. During most of each clock period, the master flip-flop is not allowed to change (CLM during T3 - T2). Only while all output (slave) stages are locked (CLS during T1 - T4) can the master flip-flop respond to input signals (CLM during T2 - T3). Note that there is a period overlap to insure that all output stages are locked before input stages are allowed to respond to input signals, and to insure that all input stages are again locked before output stages are allowed to respond to new commands (T1 - T2 and T3 - T4). Thus the logic operates as follows, starting just prior to T1 (when all input stages are locked):

T1 All output stages are locked, and since all patched inputs come from other logic components, no logic signals are allowed to change.

T2 All input stages are unlocked and respond to the patched inputs. Input flip-flops change state as commanded by the program, in the period T2 - T3, so that no timing problems occur.

T3 Input stages are locked.

T4 Output stages are unlocked and allowed to respond to the input stages. Output flip-flops are updated as determined by the program.

This process is repeated each enabled clock period. An enabled clock period is defined as that period during which a cycle of CLM and CLS (i.e., T4 - T4 in Figure 3-5) coincides with an enabling signal input to a synchronous logic component. The period of 100-150 nanoseconds allowed for this process completely avoids timing errors that can easily occur on leading or trailing edge switching systems, and also permits logic components in two or more AD/FIVE consoles to be interconnected.

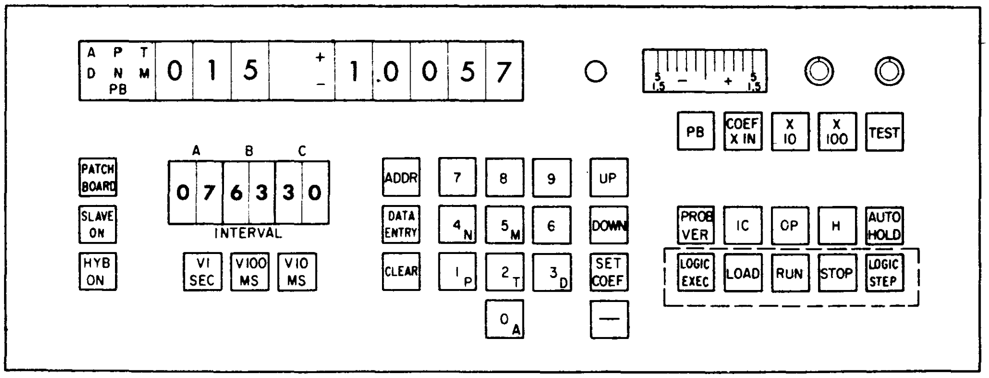
The clock signals CLM and CLS are controlled by logic mode control, and operate when the logic mode is RUN. These clock signals are distributed throughout the system to each flip-flop. The operator can control (turn on and off) clock signals to each component by patching P or V signals (see section 3.1.3.2 of this chapter for details), but he cannot route them, i.e., they do not appear on the patchboard. The operator patches the enable input to a logic component on the patchboard which, when provided with a logic one signal, allows the clock signal to reach that particular component. Thus, when the enable input to a flip-flop is logic zero, the flip-flop is latched.

3.1.2.3 Load, Run, Stop

The primary logic modes are LOAD, RUN, and STOP. These are selected by depressing one of three interlocked pushbuttons on the computer control panel (Figure 3-6).

In the RUN mode, clocking pulses (CLM and CLS) are automatically delivered to all logic elements and the logic program proceeds. In the STOP mode, CLM remains at logic one, locking all flip-flop stages. In the LOAD mode, CLM stays at logic one, locking all master flip-flops, and an internal LOAD bus takes all logic devices to their LOAD states. LOAD states are defined for individual devices in Chapter 4.

It is important to note that Logic Mode Control is normally completely independent of analog mode control. Logic and analog modes interact only through LOGIC EXEC as discussed in Section 3.1.4. It should be further noted that the logic modes of LOAD, RUN, and STOP are analogous to the analog modes of IC, OP, and H.



AD/FIVE Control Panel

Logic Modes

Figure 3-6

3.1.2.4 Logic Step

The LOGIC STEP pushbutton (Figure 3-6) allows normal operation of CLM and CLS for one clock period (i.e., one microsecond), and is operative only when the logic mode is STOP. Use of the LOGIC STEP pushbutton thus allows advance of the logic program for a desired number of steps. One push of the LOGIC STEP button advances the logic by one clock period. The use of this button in conjunction with the analog time scale control pushbuttons, the Interval Timer pushbuttons, or the P and V signals allows the operator to control the advance of the logic for a desired number of steps. If it is desired to step the logic by more than one clock pulse per push of the button, the Advance Step terminal may be used (See Section 3.1.2.5).

NOTE: THE NORMAL MEANS OF IDENTIFYING THE STATE OF THE INTERVAL TIMER (SECTION 3.1.4) IS TO OBSERVE THE LIGHTS UNDER THE IC, OP, AND H PUSHBUTTONS IN LOGIC EXECUTE, WHERE THE ANALOG MODES ARE CONTROLLED BY THE A, B, OR C STATE OF THE INTERVAL TIMER. HOWEVER, IF THE LOGIC MODE IS STOP, AS IS REQUIRED FOR LOGIC STEP, THEN MASTER HOLD IS ACTIVATED IN LOGIC EXECUTE TO PREVENT RUNAWAY ANALOG OPERATION. THEREFORE, IF THE OPERATOR IS USING LOGIC STEP TO DEBUG A LOGIC PROGRAM, HE SHOULD PATCH INDICATOR LAMPS IN THE FORM OF OR GATES TO THE INTERVAL TIMER OUTPUTS IF HE DESIRES TO MONITOR THE STATES OF THE INTERVAL TIMER.

3.1.2.5 Advance Step

The Advance Step feature enables the operator to step the logic by more than one clock pulse per push of the LOGIC STEP pushbutton. When a signal which is logic one is patched into the AST terminal on the patchboard and the logic mode is STOP, depressing the LOGIC STEP pushbutton will cause the clock to run the same as in RUN mode until the AST signal returns to logic zero. Thus a counter can be programmed to provide a burst of any desired number of clock pulses at the push of a button. The AST terminal is in field 0, logic area 1.

3.1.3 Time Scale Control

3.1.3.1 Analog

Two electronically interlocked, self-illuminating pushbuttons on the Control and Address Panel select the computer time scale (Figure 3-8). These are labelled X10 and X100.

When both pushbuttons are off, the time scale is X1. This corresponds to gains of 1 and 10 on all integrator inputs as labelled. Time scales of 1, 10, 100, and 1000 are available through the use of these two pushbuttons.

TIME SCALE	PUSHBUTTONS
X1	BOTH OFF
X10	X10 ONLY ON
X100	X10 OFF, X100 ONLY ON
X1000	BOTH X10 AND X100 ON

3.1.3.2 Logic

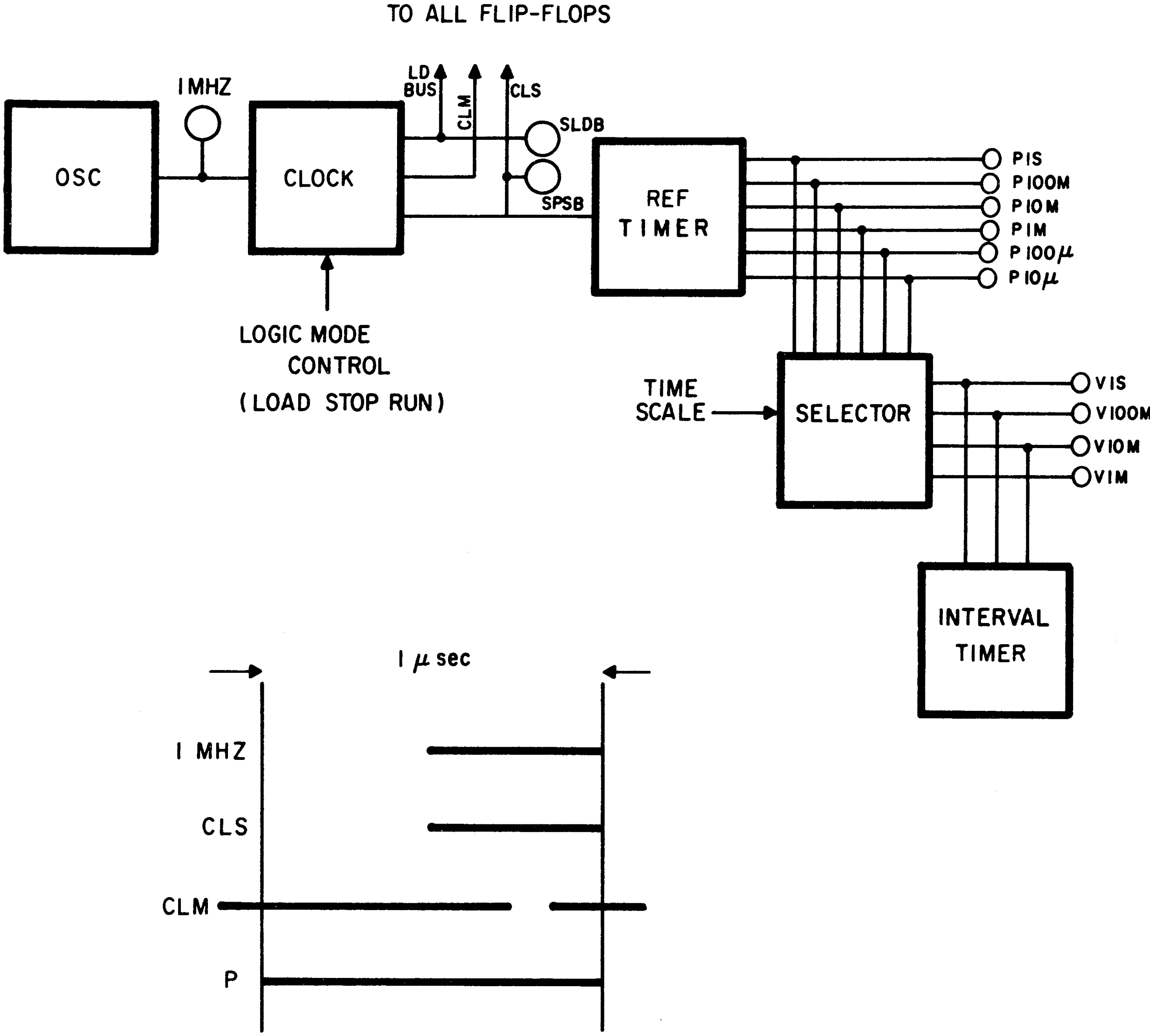
As discussed in Section 3.1.2.2 (Synchronous Logic), the AD/FIVE control system generates CLM and CLS signals that, in conjunction with a signal patched into an Enable (E) Input, control the rate of operation of all clocked logic components. A reference timer in the AD/FIVE receives the CLS signal and provides six outputs (P signals). These P signals go through a time scale selector where variable (V) signals are created. When the logic program is to be synchronized with the analog program in a problem using different computer time scales (1, 10, 100, 1000) the V signals are the signals that should be patched into enable inputs to control the logic program rate; this achieves the same time scaling for the logic program as for the analog program, since the V signals are affected by the analog time scale control buttons, but the P signals are not. Thus, when the AD/FIVE time scale is increased, the V signals are stepped up by the same factor and all logic components enabled with V signals operate at the higher rate consistent with the analog program.

Figure 3-7 is a simplified block diagram showing how the CLS, CLM, P, and V signals relate to the oscillator, each other, and the computer time scale.

Note that the P signal is logic one for the full clock period of one microsecond, and is assured to be present when needed during the clock pulse, regardless of timing variations in the clock pulse. P1S gives one microsecond pulse every second, which coincides with the enabled clock period, and thus it is logic one for one microsecond and then logic zero for 999,999 microseconds. The remaining P signals are faster in decade steps up to P10μ, which is a logic one for one microsecond, then logic zero for nine microseconds. There is one signal which is ten times faster yet. It is called logic one. If the computer time scale is X1, then the V signals are the same as the corresponding P signals. V-P relationships are tabulated below (M = millisecond):

		COMPUTER TIME SCALE			
		X1	X10	X100	X1000
V SIGNAL	V1S	P1S	P100M	P10M	P1M
	V100M	P100M	P10M	P1M	P100μ
	V10M	P10M	P1M	P100μ	P10μ
	V1M	P1M	P100μ	P10μ	Logic One

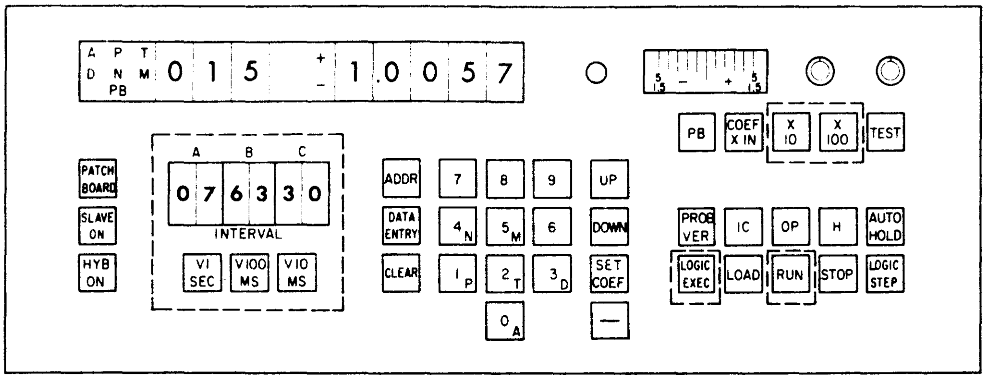
Note that X1 = real time, X10 = 10 times faster than real time, etc.



AD/FIVE Timing Signals

Figure 3-7

3.1.4 Repetitive Operation



AD-FIVE CONTROL PANEL

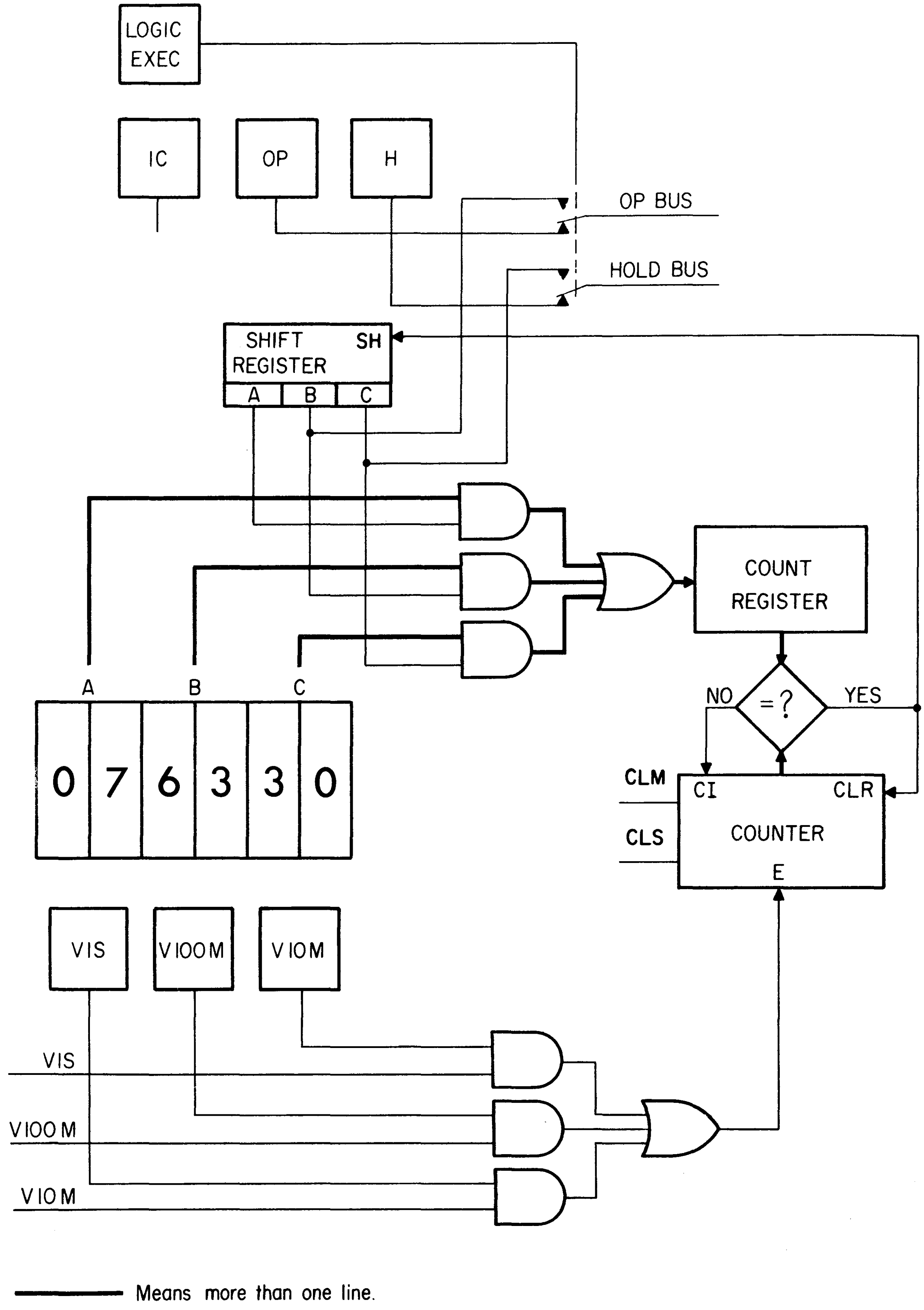
$$\left(\text{Time} = \frac{\text{(Thumbwheel Setting) (V Signal)}}{\text{(Time Scale)}}\right)$$

Interval Timer Operation

Figure 3-8

The Interval Timer and Logic Execute pushbuttons are used to control analog repetitive operation from the control panel. Whenever the logic mode is RUN, the interval timer counts V signals and cycles through its A, B, and C states, staying in each state for the number of V signals set by the interval thumbwheels. The interval timer pushbuttons are used to select a particular V signal for the timer to count. Thus, since the V signals are subject to time scale control, the interval timer states are also subject to time scale control. Figure 3-8 shows the relationship between the thumbwheels, pushbuttons, and time scale in determining the time in each interval timer state.

When the LOGIC EXEC pushbutton is depressed, the interval timer states are given control of the analog mode buses and the time in each analog mode is precisely determined by the interval timer. This means that regardless of the analog mode pushbuttons, the analog modes will follow the interval timer states A(IC), B(OP), and C(H). If the logic mode is STOP, Master Hold will be activated. This occurs during Logic Execute because it is possible to enter the STOP mode while the interval timer is in the B(OP) state, and Master Hold then stops the analog and logic programs simultaneously in order to prevent runaway operation of the analog components. Figure 3-9 is a simplified block diagram showing interval timer operation under pushbutton control. In Figure 3-8 and other Figures in this chapter, the thumbwheels are shown with example values of 07,63,30. This corresponds to seven time units of IC, sixty-three time units of Operate, and thirty time units of Hold.

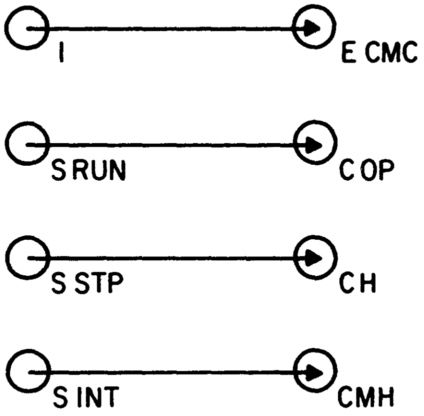


Interval Timer

Figure 3-9

3.1.5 Simultaneous Operation

Often one encounters the requirement to initiate both the logic and the analog mode controls simultaneously. When using Logic Execute to provide Repetitive Operation as described in the previous section, this simultaneous start-up is assured. One may, however, desire simultaneous start-up of the logic and analog programs without Rep-Op. It should be obvious that the RUN and OP pushbuttons could not be manually pushed with the degree of simultaneity required in high-speed computation. More than one method can be used to achieve simultaneous operation, but the method shown in Figure 3-10 is recommended. It is convenient to use when AUTO HOLD (Section 3.1.1.3) is additionally required.



This patching makes the analog modes track the logic modes. The relationship is

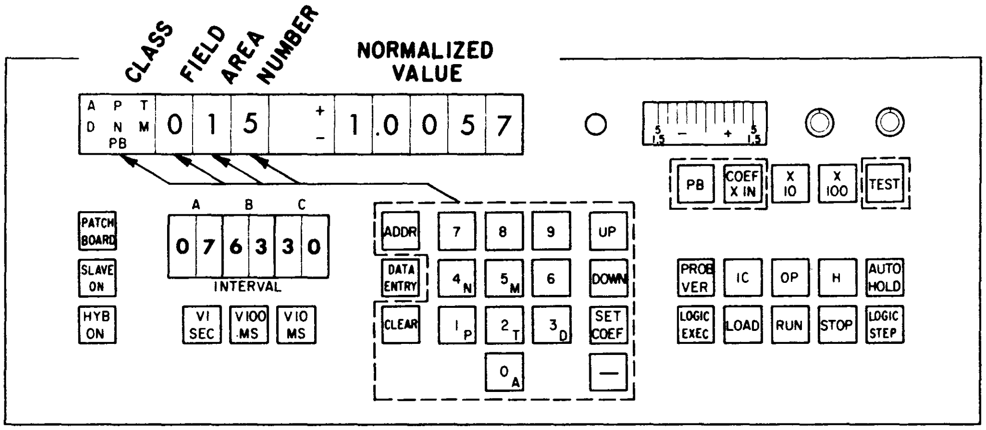
LOAD	–	IC
RUN	–	OPERATE
STOP	–	HOLD

To use auto-hold depress both LOGIC EXEC and AUTO-HOLD.

Simultaneous Logic and Analog Mode Control

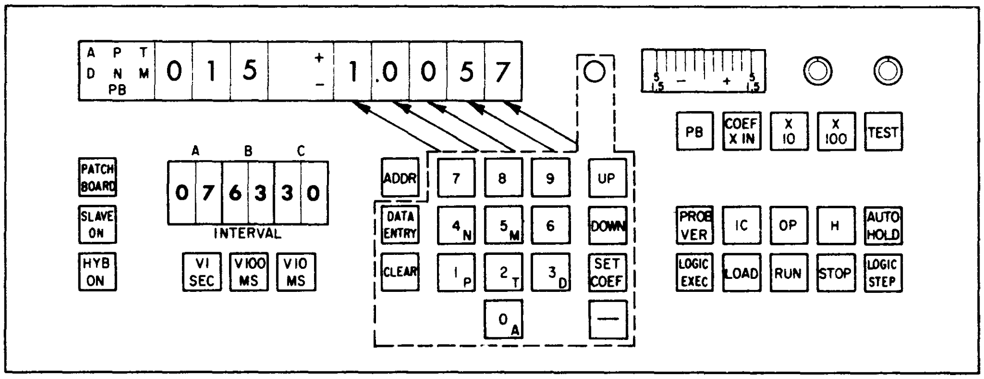
Figure 3-10

3.1.6 Addressing



AD-FIVE CONTROL PANEL

Addressing

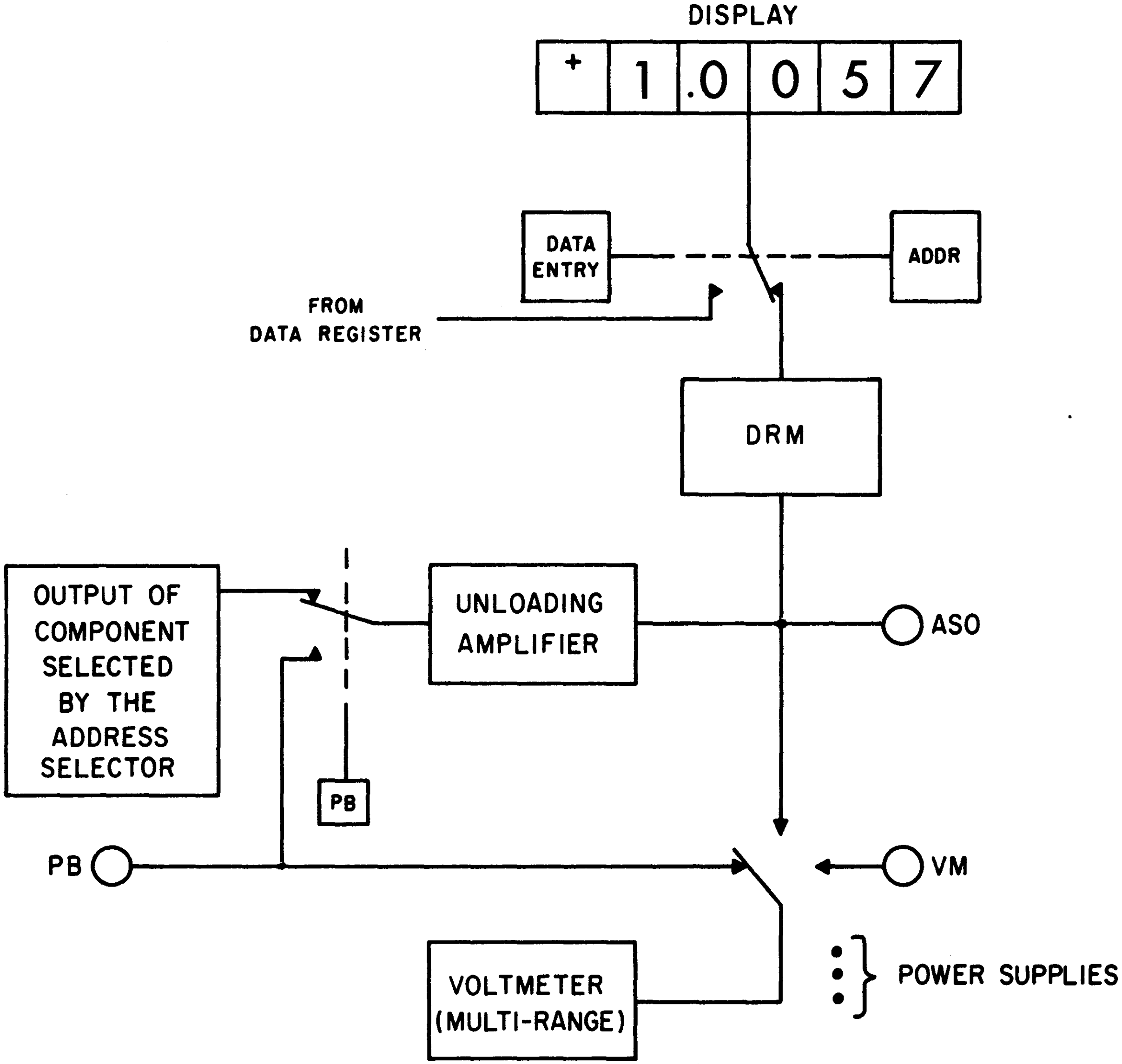


AD-FIVE CONTROL PANEL

Coefficient Setting

Figure 3-11

The Address Selector System allows the operator to monitor the output of the analog elements with a precision digital ratiometer (DRM). In addition to the Address Selector System, a panel voltmeter may be used to augment the DRM. The panel voltmeter is also used to monitor the power supplies. Figure 3-12 illustrates the Readout System.



Readout System

Figure 3-12

Address selection in the AD/FIVE is achieved by means of the numerical entry pushbutton group on the Control and Address Panel. Latching the ADDR pushbutton puts the panel in the address state, and the output of the numerical entry pushbuttons is then steered to the four display windows at the left top of the Control and Address Panel. As the numerical entry pushbuttons are pushed, the component address is displayed in the windows in the order in which the buttons are pushed, from left to right. Each component address consists of a letter for component class, and a number for field, area, and individual element. Pushbuttons 0 through 5 are marked with a single letter abbreviation for the addressable component classes and a miscellaneous (M) class reserved for future use, as shown in Figure 3-11. The numbers for field, area, and individual element refer to the unique location of each addressable component on the AD/FIVE patchboard.

The patchboard is divided horizontally into two fields, 0 and 1, and each field is divided vertically into six areas, 0 through 5 (See Figure 2-10). Within each area, similar components in each class are individually numbered to provide a completely unique address. The component number appears on the patchboard adjacent to the component patchboard holes.

Pushing the CLEAR pushbutton while the ADDR button is illuminated clears the address display windows to zero, i.e., A000 (Amplifier, field 0, area 0, individual element 0). The operator can then address any other component by pushing the appropriate numerical entry buttons in sequence to enter the class letter, and the field, area, and individual element numbers. (When the DATA ENTRY button is illuminated, however, the CLEAR pushbutton clear the data register to 0.0000.)

If the operator wishes to quickly scan the readouts for all the components of a given class in a single area without having to push all the numbers for each component idividually, he may do so by pressing down the ADDR pushbutton and holding it down when he gets to the digit whose value he wants to change individually. For example, if he desires to read the outputs of amplifiers 000, 001, 002, 003,.....009, he would push the buttons A, 0, 0, and then depress ADDR and hold it down. Then by pushing the numerical entry buttons in order 0,1,2,3,......9 he would be able to quickly and easily scan the gamut of amplifiers in area 0 of field 0, without the effort of pushing the buttons for class, field, and area for each component.

There are digits on the numerical entry pushbuttons which are meaningless for class, field, and area entry. Pushing a meaningless button for class and field will extinguish the display window for each of these. If any button from 6 to 9 is pushed for class, for example, the class window display will be extinguished. Likewise, pushing any button greater than 1 for field will extinguish the field display window. In the case of a meaningless number being selected for area, however, the number will be displayed in the area window.

In Figure 3-11 and in other Figures in this chapter the display is shown for a component with field 0, area 1, individual element number 5. The output value read by the digital ratiometer is 1.0057.

3.1.6.1 Amplifier, Trunk, and Nonlinear

When an amplifier, trunk, or nonlinear component class is addressed, the right hand portion of the display window contains the normalized value of the component output.

3.1.6.2 Miscellaneous

A miscellaneous component class has been reserved for use in further AD/FIVE development. If this address class is assigned in the future to any computer component, the subject will be covered in an appendix to this manual.

3.1.6.3 Potentiometers (COEF XIN)

Addressing potentiometers varies from other component addressing in that a setting and an output are both of interest to the operator. In normal usage the address system removes the patched potentiometer input and replaces it with computer reference, making the displayed output in the windows the potentiometer setting. CAUTION: IT MUST BE EMPHASIZED THAT THERE ARE TWO RELAYS FOR SWITCHING THE INPUTS OF POTENTIOMETERS PER AREA, I.E., WHEN A POT IS ADDRESSED, FIVE OF THE TEN POTS IN AN AREA ARE CONNECTED TO COMPUTER REFERENCE, NOT JUST THE ONE ADDRESSED. When the COEF XIN pushbutton is latched, however, the patched input is returned to the potentiometer, and the panel display is then the normalized product of the coefficient setting times the input, that is, the actual output of the potentiometer. IT SHOULD THUS BE REALIZED THAT WHEN A PROBLEM IS RUNNING ON THE COMPUTER, ADDRESSING ONE POT WILL CAUSE ALL FIVE POTS ON THE SAME RELAY TO BE REMOVED FROM THEIR INPUTS AND CONNECTED TO COMPUTER REFERENCE, UNLESS COEF XIN IS LATCHED. IT CAN READILY BE SEEN HOW THIS COULD AFFECT THE RUNNING OF A PROBLEM, AND CARE SHOULD BE TAKEN TO KEEP IN MIND THE FACT THAT THE POTS ARE SWITCHED IN GROUPS. The pots are grouped as follows: Pots 0, 2, 4, 5, 7 are on one relay; pots 1, 3, 6, 8, 9 are on the other relay.

3.1.6.4 PB

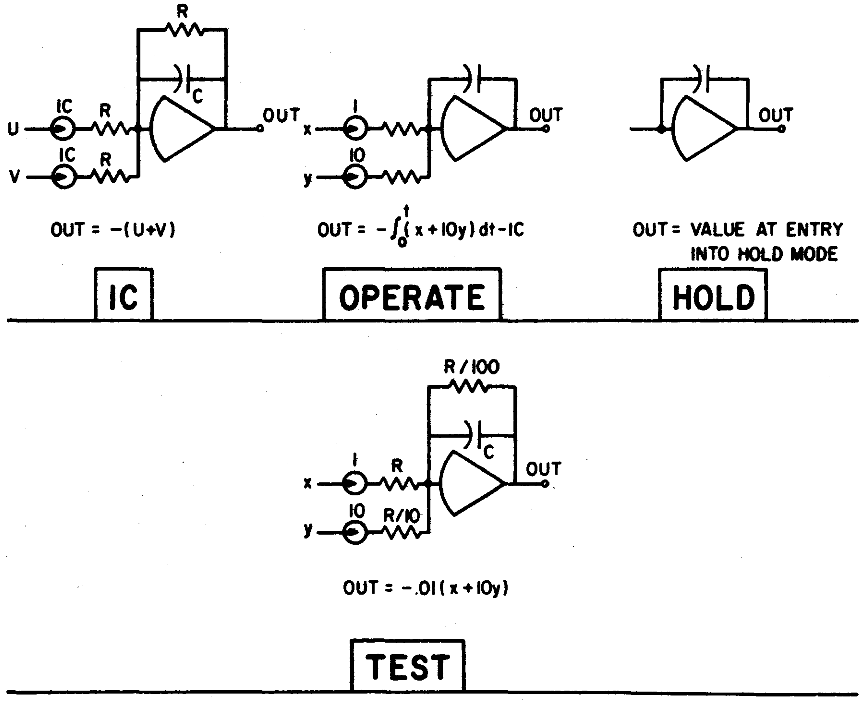
The PB (patchboard) pushbutton is not actually a part of the address system, but is an alternative to it. PB is a patchboard input terminal in field 0, area 0. When the PB pushbutton is latched, the value patched into PB is delivered to the DRM. When PB is unlatched, the DRM returns to the last component addressed. See Figure 3-12.

3.1.6.5 Test

Problem checkout often requires that the operator determine the value present at the input of an integrator in the IC mode. The AD/FIVE has been designed to allow the operator to determine this value by addressing the amplifier output and latching the TEST pushbutton. When TEST is latched in IC mode and an amplifier is addressed, a gain 100 (10K) resistor is switched into the amplifier feedback (in parallel with the capacitor), and the amplifier is put into Operate. The amplifier output will then go, exponentially, to a value proportional to the sum of its inputs (See Figure 3-13 for a detailed explanation). The overall gain and the rate of exponentiation will be a function of the individual integrator time scale:

TIME SCALE	TIME TO FULL SCALE
X1	~ .05 sec.
X10	~ .05 sec.
X100	~ .0005 sec.
X1000	~ .0005 sec.

The time to full scale must be considered when performing the static check under high-speed digital control. It is not a factor for manual pushbutton operation by the operator, but with a scope the operator could observe the exponential rise and perform a dynamic check. See Figure 3-13 for further information.



Test Feature

Figure 3-13

Computer mode must be IC to provide initial values at x and y from other integrators. Only the integrator addressed is configured for TEST.

Anticipating that the value of the patched inputs may exceed unity at t = 0, the scaling provided is .01 as shown. Should this scaling provide too small an output value (in the case of small inputs) the amplifier can be conveniently rescaled by pushing the X10 pushbutton or by patching the X10 terminal at the individual integrator. This will increase the gain by a factor of 10 and hence increase the output reading by a factor of ten. (Note that pushbutton or patched X100 has no effect in this configuration.)

3.1.6.6 Panel Voltmeter

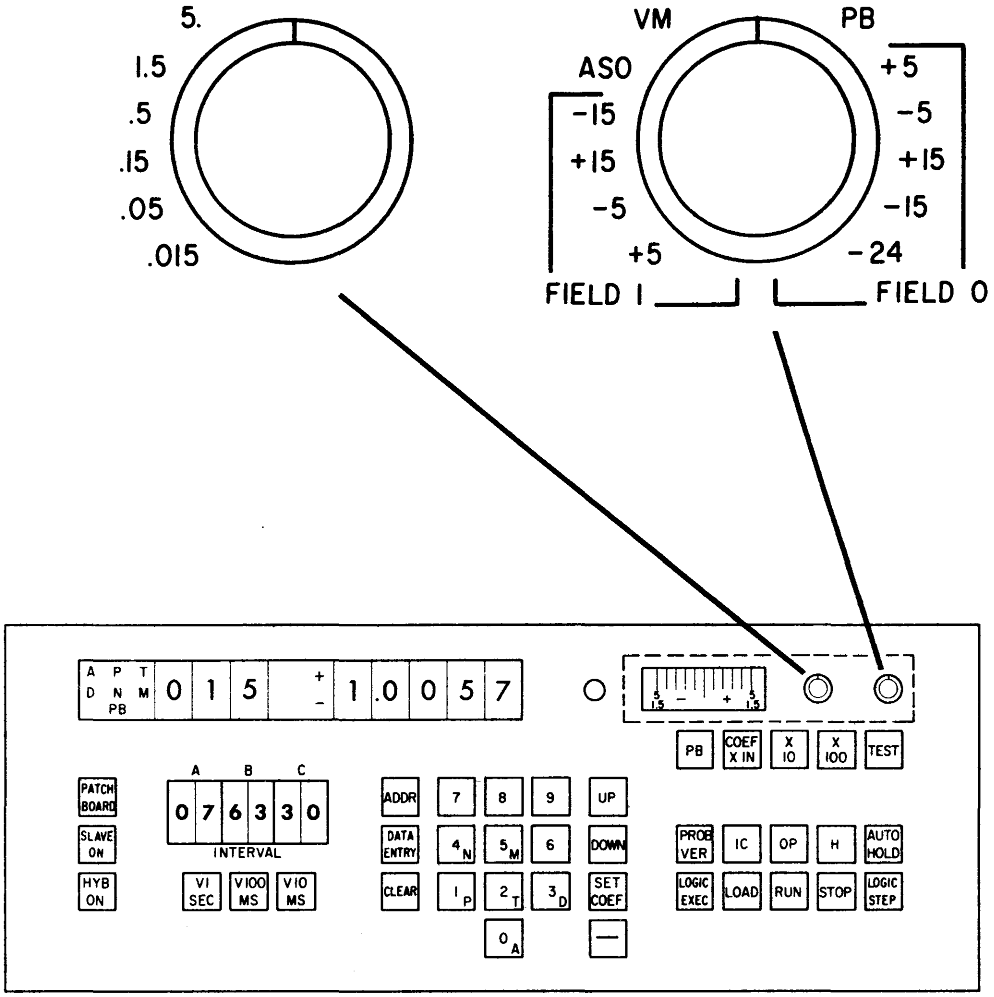
The controls for the voltmeter are illustrated in Figure 3-14. Note that the range of the voltmeter is calibrated in machine units for use by the programmer/operator. (One machine unit = 10 volts.) Thus the voltmeter is used in a way which is consistent with the DRM.

The purpose of the 5 unit range is to allow the monitoring of the 24 volt power supply. For all other used the 1.5 unit range is recommended. Good hardware performance (especially for non-linear hardware) depends upon good problem scaling practices, which in turn indicate the use of the 1.5 unit range. Sources for the voltmeter are selected by the selector switch. See Figure 3-12. The Panel Voltmeter is a low impedance input device. Do not leave its selector switch in the PB position if PB is a high impedance input. This could cause false pot loadings if pot outputs are read via the PB input.

3.1.7 Coefficient Setting

3.1.7.1 Hand Set Potentiometers

Handset potentiometers with calibrated dials may be located at potentiometer address numbers 0, 1, 4, and 9 in any area. The dials are behind a hinged door just below the patchboard.



AD-FIVE CONTROL PANEL

Panel Voltmeter Controls

Figure 3-14

3.1.7.2 Servo Set Potentiometers

Both the Address System and the Servo System receive inputs from the numerical entry pushbuttons on the control and address panel. The latching of the ADDR or the DATA ENTRY pushbutton determines whether the numerical information from the numerical entry buttons will be sent to the address system or the data register. In order to set a servo pot the operator first addresses the pot with the ADDR button latched. Then he depresses the DATA ENTRY button and latches it. The numerical entry information from the pushbuttons will now be steered to the data register and will be displayed in the right hand portion of the panel display windows.

Data is entered serially from left to right by depressing the numerical entry pushbuttons in the desired sequence. When the correct value appears in the data windows, the operator is ready to set the coefficient. If an error is made during data entry, the CLEAR button will clear the data register to zero, and the correct value may be reentered. To set the coefficient, the operator presses the SET COEF pushbutton, and the servo system then sets the potentiometer. The potentiometer input is connected to computer reference whenever the pot is addressed, unless COEF XIN is latched. However, even with COEF XIN latched, the pot is automatically disconnected from its patched inputs and connected to computer reference during the ~1 sec. during which the pot is setting. The only way to change the coefficient without removing the pot from its patched inputs is by the use of the UP and DOWN pushbuttons, as described in Chapter 2, Section 2.2.7.3. When the process of servo setting is completed, the address system automatically returns to the address state, and the ADDR button is illuminated. The windows then display the potentiometer setting, or, if COEF XIN was latched when the pot was set, the windows will then display the actual potentiometer output.

An alternate method of potentiometer control available to the operator is that described in Chapter 2, Section 2.2.7.3, which in effect lets him control the servo set pots in a manner similar to handset pots through the use of the UP and DOWN slew pushbuttons and the slew rate potentiometer on the control panel.

3.2 LOGIC PROGRAM CONTROL

The principle of control by exception has been utilized in the AD/FIVE to allow the operator to override pushbutton control from the logic patchboard. Each control override feature is individually enabled from the logic patchboard and is accomplished electronically without the use of mechanical switching. The AD/FIVE operator can have complete logic control of:

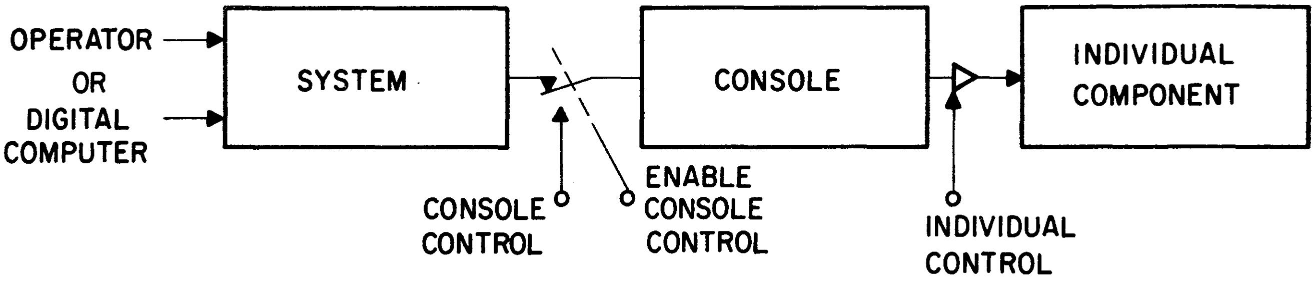
Analog Mode
Logic Mode
Time Scale
Clock Selection
Slaving

without using any pushbutton controls. The logic patchboard terminals used to exercise this control are all located in logic area 1 of field 0.

The AD/FIVE control structure is based on the concept of LEVELS OF CONTROL. Three control levels are provided: system, console, and individual. Individual control is the highest level, overriding both console and system control. The distinction between system and console levels is most important in a multi-console system. For instance, the system (consisting of several consoles) may be in analog mode Operate while one console within the system is in analog mode Hold.

This section of the Reference Manual discusses the system and console control as accomplished on the logic patchboard. Individual control of logic components is described in Chapter 4.

The normal control of the computer is as follows: either the operator pushes a control button or the digital computer activates the same circuit. The signal propagates to each console and then to each component in each console. The patched logic program can override this normal control at two levels.



Logic Program Control

Figure 3-15

In the first case an entire console can be controlled. Note that this is still important in a one-console hybrid because it means that, since the console control is downstream of the system control, the patched logic program can override the digital computer. Thus the operator can at his discretion override certain parts of a hybrid program. The lights under the mode and time scale pushbuttons will correctly indicate such control.

In the second case an individual element can be controlled separately from its neighbors.

NOTE: The AD/FIVE logic program priority levels are as follows:

A. Master Hold controls all integrators and cannot be overridden by individual integrator controls (however, see note * below).
Master Hold may be generated by:

1. Patching ECMC (Enable Console Mode Control) to a logic one, and then patching CMH (Console Master Hold) to a logic one;

*2. An overload in any analog component occuring while the AUTO HOLD pushbutton is depressed and illuminated;

*3. Depressing the logic STOP pushbutton while in LOGIC EXEC.

*Note that, though individual integrator controls will not override Master Hold when that condition is present, conditions 2 and 3 of Master Hold can be disabled if ECMC is a logic one, as that totally removes control of the computer from the control panel pushbuttons. (See Figure 3-4.)

B. Individual integrator patched controls will override all controls except Master Hold. However, it must be emphasized that unpatched integrator controls are not automatically logic zero. If the console mode is Hold, for example, then patching the integrator control OP (Operate) to a logic one will not cause the integrator to operate until the console hold is overridden with a logic zero patched to the integrator control H (Hold).

C. Console patch controls, when enabled with a logic one in ECMC, override the control panel pushbuttons and the interval timer.

D. The interval timer overrides the analog mode control pushbuttons when LOGIC EXEC is latched.

E. The analog mode control panel pushbuttons are the lowest level of control priority.

3.2.1 Analog Mode Control

In a previous section on pushbutton mode control it was pointed out that the operator could select Initial Condition, Operate, or Hold analog modes. The mode selected at the control panel or by the digital computer is represented at the patchboard as system outputs: System Hold and System Operate. The scheme for representing the three analog modes is:

ANALOG MODE	SYSTEM OUTPUTS	
	SH	S OP
IC	0	0
OP	0	1
H	1	0

The presence of analog Master Hold is signaled at the patchboard by a logic 1 on System Interrupt (S INT). Adjacent to these patchboard outputs (S INT, SH, and S OP) are three analogous patchboard inputs (C MH, CH, and C OP), and a fourth input used to enable Console Mode Control (E CMC). When a logic one is present at the input E CMC, the console analog mode will be determined by the logic signals present at the inputs C MH, CH, and C OP. Note that the AD/FIVE is designed to interpret an unpatched input to an enabled logic computing or control component as a logic zero. In components which are not enabled, such as individual integrator controls, an override signal may be necessary to chieve the desired operation of the component.

To control the console analog mode from the logic patchboard, the operator simply patches a logic 1 signal into E CMC and then controls the analog modes by using the following scheme of inputs:

ANALOG MODE	PATCHED INPUT	
	CH	C OP
IC	0	0
OP	0	1
H	1	0 or 1

A logic 1 in C MH causes the console mode to be Master Hold. If the operator wishes to slave the analog modes of two consoles, he patches the System Outputs of the master console to the Console Mode Control inputs on the slave console using trunk lines to carry the signals.

3.2.2 Logic Mode Control

The system Logic Mode, as determined by the mode control pushbuttons or the digital computer input, is represented by the following logic outputs:

LOGIC MODE	SYSTEM OUTPUTS	
	S STP	S RUN
LOAD	0	0
RUN	0	1
STOP	1	0

Console level logic mode control is accomplished by patching a logic 1 in Enable Console Logic Control (E CLC) and using the following patched logic inputs:

LOGIC MODE	PATCHED INPUTS	
	C STP	C RUN
LOAD	0	0
RUN	0	1
STOP	1	0 or 1

3.2.3 Time Scale Control

As discussed in Section 3.1.3.1, four time scales are controlled by two pushbuttons on the control panel, labelled X10 and X100. The pushbutton states are represented by the logic levels of system outputs SX10 and SX100, so that the system time scale may be determined at the logic patchboard as follows:

SYSTEM TIME SCALE	SYSTEM OUTPUTS	
	SX10	SX100
X1	0	0
X10	1	0
X100	0	1
X1000	1	1

The console time scale inputs override the system control signal whenever a logic 1 is patched into Enable Console Time Scale (E CTS).

CONSOLE TIME SCALE	PATCHED INPUTS CX10	 CX100
X1	0	0
X10	1	0
X100	0	1
X1000	1	1

If the operator wishes to slave the analog modes of two consoles, he patches the System Outputs of the master console to the Console Time Scale inputs on the slave console using trunk lines to carry the signals.

3.2.4 Miscellaneous

The following sections describe a group of miscellaneous control features available on the logic patching area of the patchboard.

3.2.4.1 Clock Selection

The normal AD/FIVE logic time base is terminated at the logic patchboard as a 1 MHz output. The AD/FIVE logic system may operate from an appropriate external timing source patched into the input terminal EXT. The input patched to EXT is substituted for the 1 MHz signal only when this feature is enabled by a logic 1 input in Enable External (E EXT).

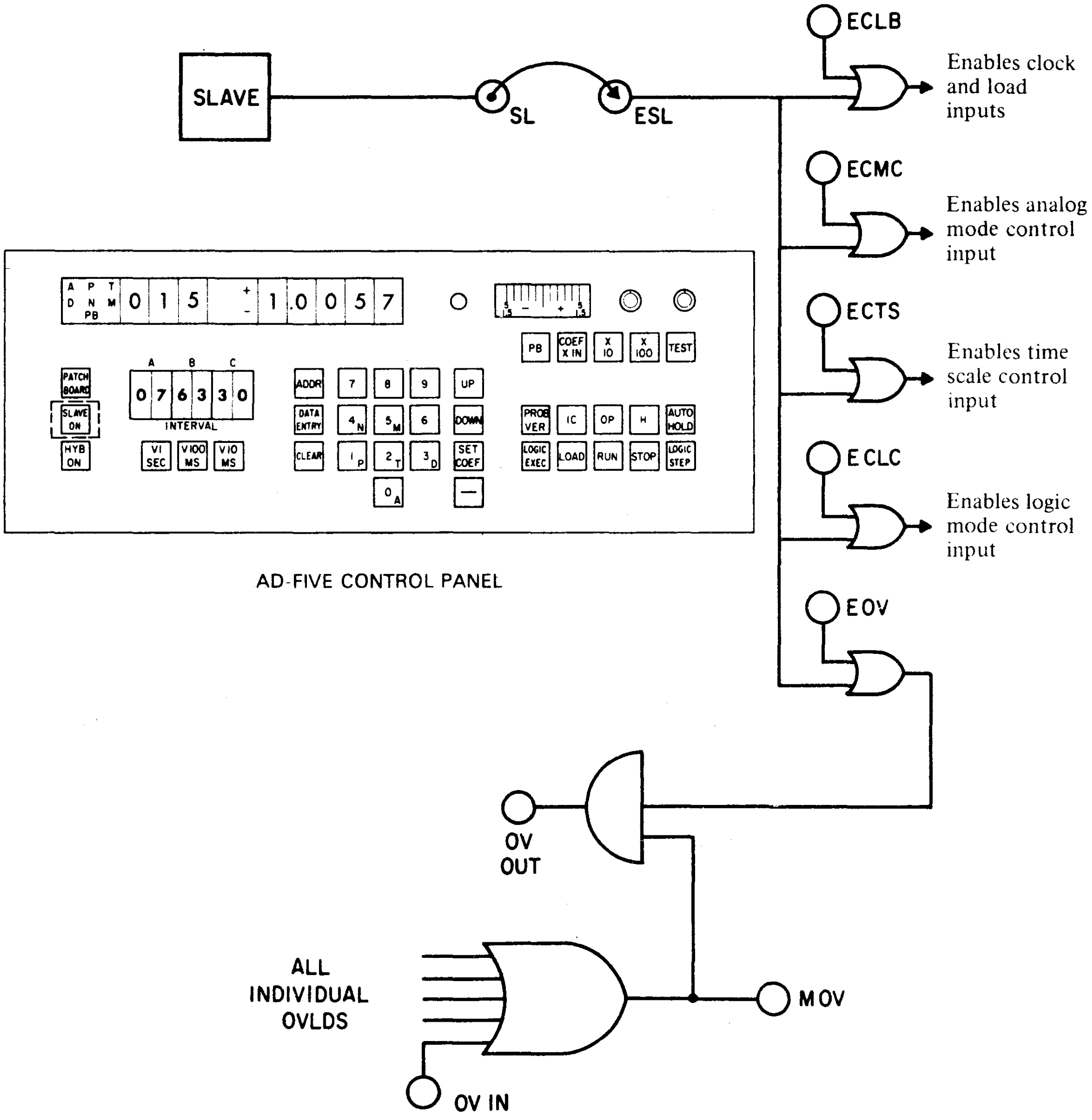
3.2.4.2 Slaving

The AD/FIVE may be operated slaved with either another AD/FIVE and AD/FOUR, or any other computer, as described below. It will be assumed that suitable trunking facilities have been installed to carry the signals between the consoles. Operation with more than two consoles is a simple extrapolation of the method described here for slaving two consoles.

However, before proceeding with the actual patching, the following concepts should be thoroughly understood. Refer to Figure 3-16. The output of the SLAVE pushbutton is terminated at the patchboard hole SL. If SL is patched to ESL, then the mode and time scale control is removed from the control panel pushbuttons and transferred to the logic patchboard. Also the OV OUT signal is gated on so that overload signals in the slave console can reach the master console. Note that any of the five enable signals could have been patched separately. The slave button (or any other patched input to ESL) just does this job with one action.

CAUTION: WHEN SLAVING TWO CONSOLES THE LOGIC MODE CONTROLS ARE NOT SLAVED. IF THEY WERE, THE TWO SEPARATE OSCILLATORS IN THE CONSOLES WOULD SURELY PRODUCE A DIFFERENT NUMBER OF CLOCK PULSES DURING ANY COMPUTER RUN. IN ORDER TO INSURE THAT THE LOGIC PROGRAM OF EACH CONSOLE ADVANCES THE SAME NUMBER OF STEPS THE CLOCK SIGNALS FROM THE MASTER CONSOLE ARE USED TO ACTIVATE THE LOGIC ELEMENTS IN THE SLAVE CONSOLE. THE SCHEMES PRESENTED HERE ALSO INCLUDE LOGIC MODE SLAVING, BUT THIS MERELY LIGHTS THE LOGIC MODE PUSHBUTTONS OF THE SLAVE CONSOLE, WHICH MAY BE DESIRABLE FOR VISUAL MONITORING.

Four possible slaving situations will now be considered.

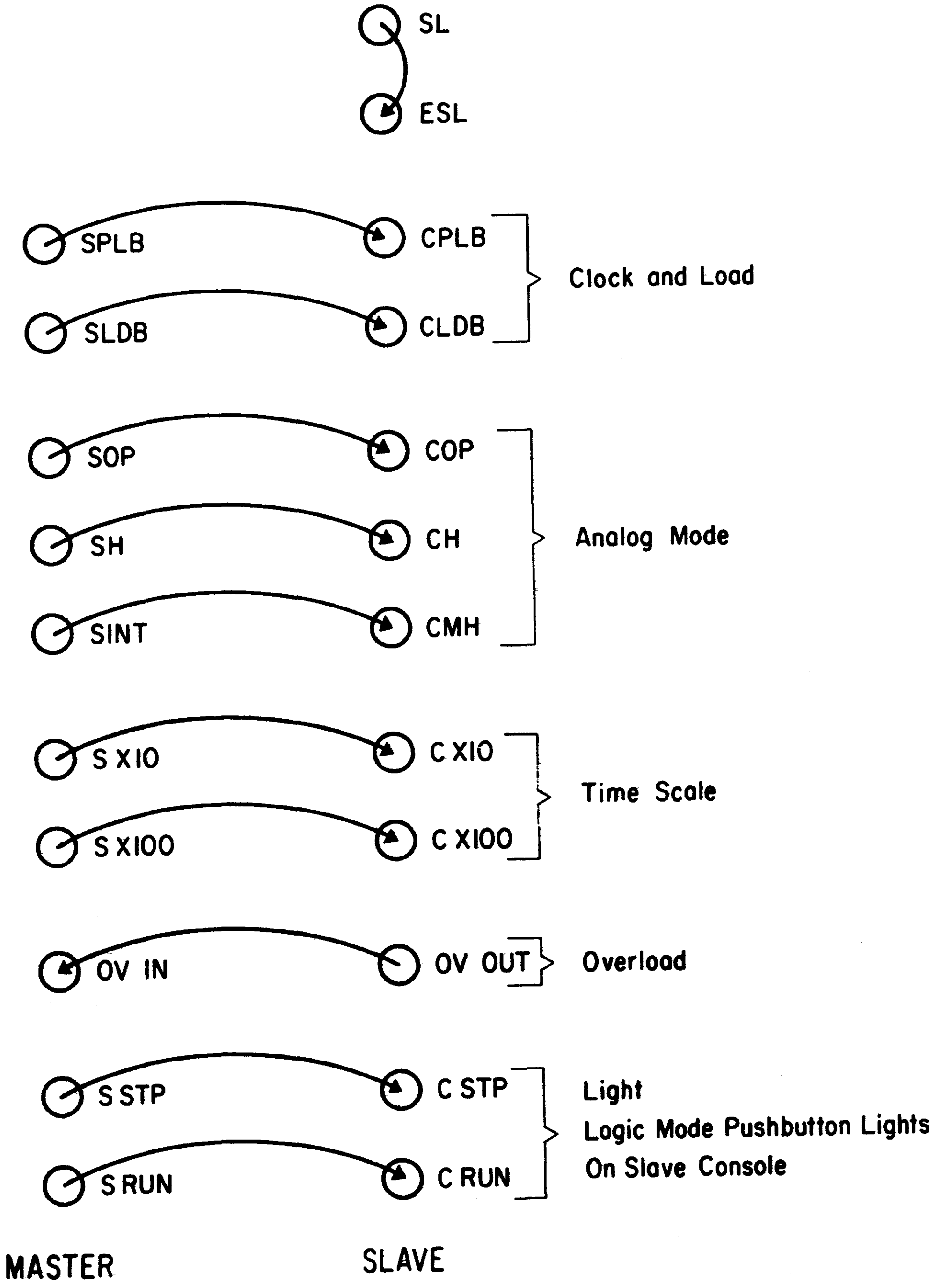


Slaving and Overload Signals

Figure 3-16

3.2.4.2.1 Slaving two AD/FIVE computers.

To slave two AD/FIVE computers, patch the control signals as shown in Figure 3-17.



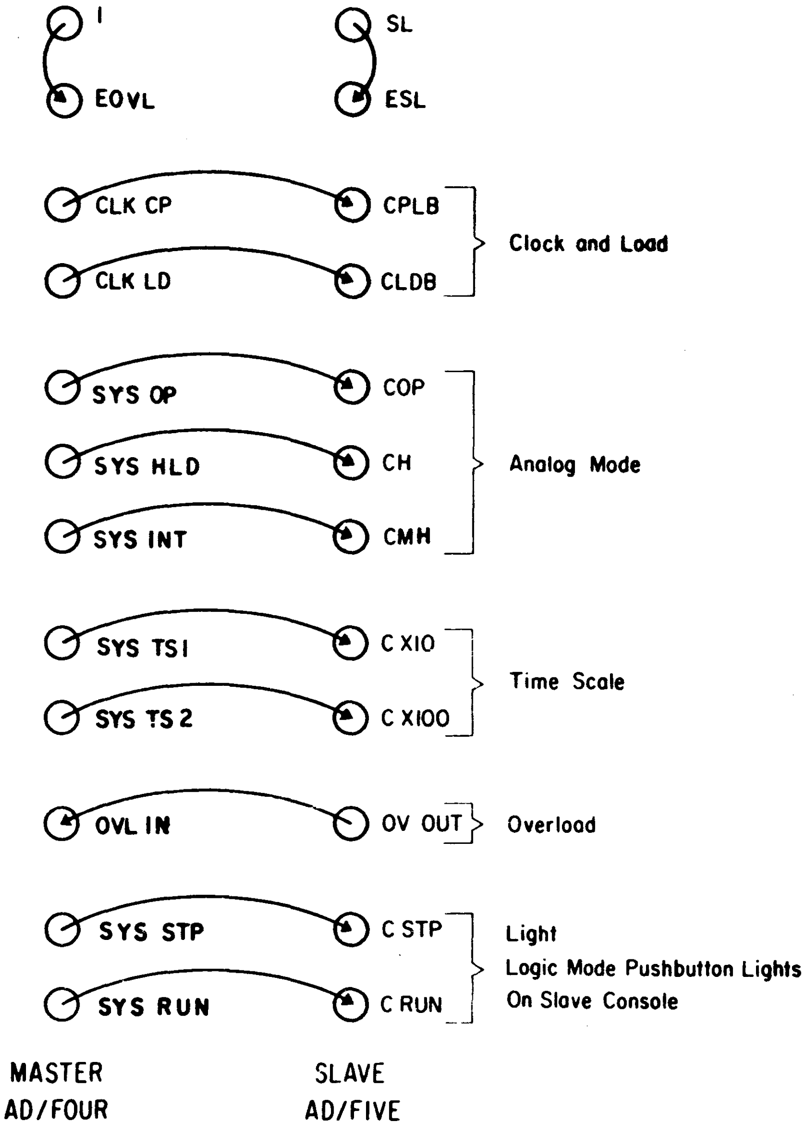
Slaving Two AD/FIVE's

Figure 3-17

3.2.4.2.2 Slaving an AD/FIVE to an AD/FOUR

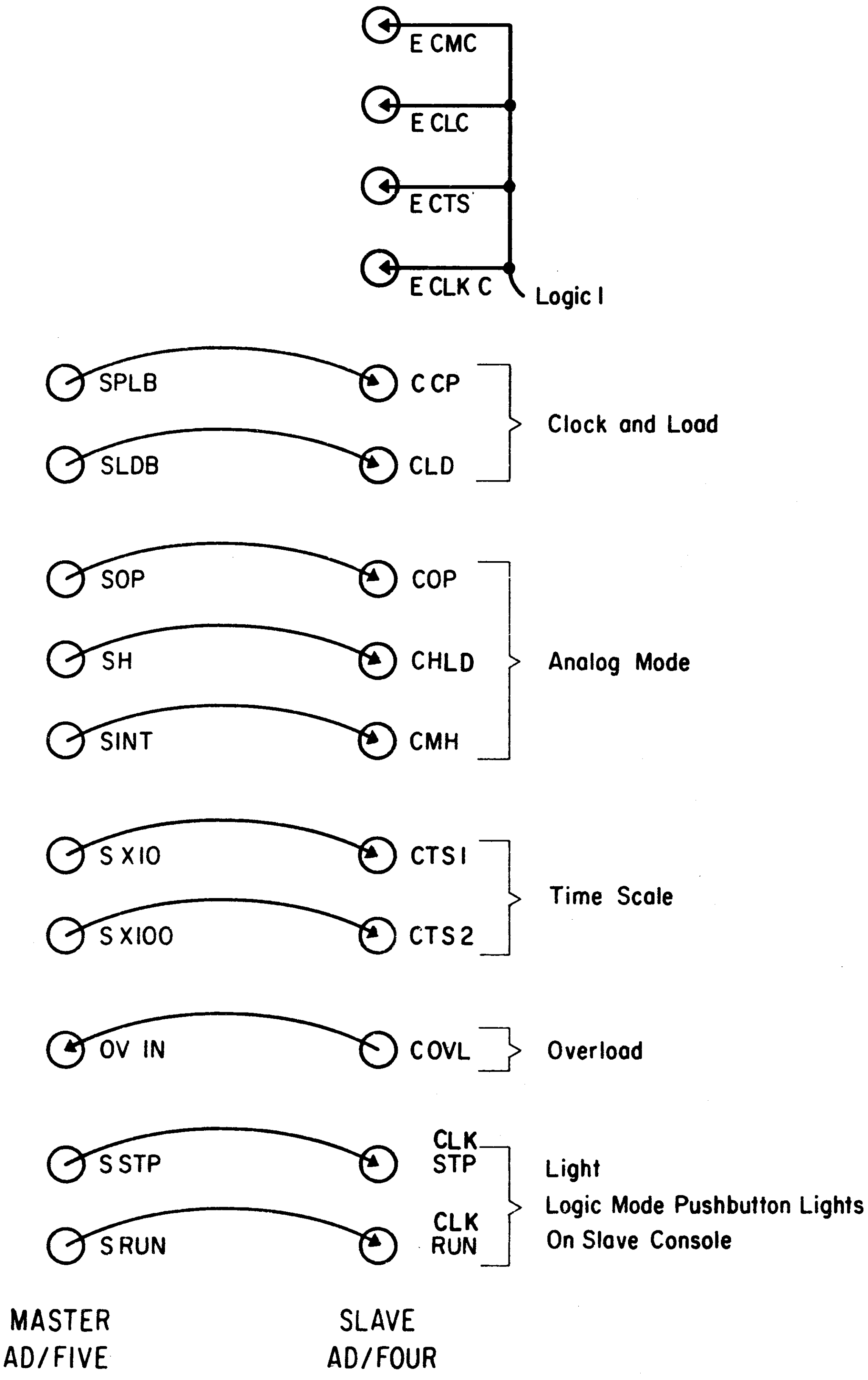
These two computers can be slaved with either one chosen as the master Patching varies according to that choice as shown in Figure 3-18 and Figure 3-19. Since the two computers use the same type of logic they are compatible.

CAUTION: IT IS ASSUMED THAT SENSIBLE PRECAUTIONS ARE TAKEN TO PREVENT DELIVERY OF 100 VOLT SIGNALS TO THE AD/FIVE. BY PROVIDING THIS CONTROL INFORMATION, APPLIED DYNAMICS DOES NOT ASSUME ANY RESPONSIBILITY FOR ANY DAMAGE CAUSED BY SUCH SIGNALS.



Slaving an AD/FIVE to an AD/FOUR

Figure 3-18



Slaving an AD/FOUR to an AD/FIVE

Figure 3-19

3.2.4.2.3 Slaving an AD/FIVE to Brand X

The previous examples illustrate the technique of slaving by defining the role of all necessary patching operations to use the AD/FIVE as either the master or the slave console. When slaving with computers not of our manufacture, the following comments may prove helpful:

a) AD/FIVE logic levels are: logic one: zero volts
logic zero: +5 volts

b) If slaving to a relay-mode-control computer, make the fast AD/FIVE be the slave.

The SLAVE pushbutton is latched on the Slave Console. This provides, automatically, the enabling of E CLB, E CMC, and E CTS on the slave console which then requires patchboard inputs to control the logic clock (not the logic mode), analog mode, and the time scale.

3.2.4.3 IO Devices

Terminations are provided for patching to various IO Devices. Since these terminations are labelled for a general use, individual installations may vary according to the owner's choice. The specific use of these holes in this installation is tabulated below.

RECORDER (REC)

R OP
RUN
RCA
RCB
MKA
MKB

PLOTTER (XY)

RUN
PEN

DISPLAY (DSP)

STS
DCA
DCB
DCC

NOTE: Each user is encouraged to record here, the use of these holes in his facility.

CHAPTER 4

PATCHING AND PATCHBOARD TERMINATIONS

TABLE OF CONTENTS

ILLUSTRATIONS

4.0 INTRODUCTION

This chapter describes the computing components of the AD/FIVE. Due to the high degree of modularity of the AD/FIVE the individual machine configurations can vary widely. All possible configurations are presented here in order to provide a reference for using any individual machine as well as to enable the reader to understand options available for expansion.

The chapter is organized as follow:

Section 4.1 is a brief description of the notation used. SCi notation is used except where specifically designated otherwise.

Section 4.2 is a description of computing elements.

Section 4.3 is a catalog of the printed circuit cards which mount directly behind the patchboard. In general these cards contain more than one of the computing elements. All of the legitimate locations for each type of card are shown.

Section 4.4 is a catalog of Patchboard Modules. Each of the Patchboard Modules provides patching to three of the printed circuit cards.

4.1 NOTATION

Applied Dynamics supports the contention that it is proper for technical societies to regulate professional standards such as uniform graphics. The appropriate society, in the case of an analog computer, is the Simulation Councils, Inc. Therefore, we have used their uniform standards wherever possible. There are additional symbols which are taken from other authorities, and the only symbols which we have invented are symbols to represent proprietary devices, e.g., the emitter-follower override defined in Chapter 3.

4.1.1 SCi Standards

Uniform graphics for *SIMULATION*

The symbols and the methods of laying out analog and hybrid computer diagrams presented here are advocated to alleviate the confusion caused by the uncoordinated invention of new symbols and diagramming practices. The increasing use of hybrid techniques and equipment has aggravated an already bad situation to the point that it is often no longer possible for one worker in our field to read another's diagram. Usually this is because symbols are devised and diagrams are drawn to include details peculiar to a particular kind of equipment. Such a wiring, or "patching," diagram is of course necessary for setting up and checking out an actual simulation, but hardware-peculiar details are only confusing to those with other kinds of equipment. With few exceptions, the use of a simplified signal-flow diagram to illustrate technical articles is much more effective.

With the foregoing in mind an SCi committee composed of

GEORGE BURGIN
JOE HUSSEY
HANS JORGENSEN
GRANINO KORN
JOHN McLEOD

selected the symbols and offers the following suggestions for their use. Primary considerations were current usage, clarity, and simplicity. We devised no new symbols and, unless there were overriding indications to the contrary, we adopted those already in widest use. Clarity and simplicity, we believe, will be enhanced by the choice of unique shapes to represent different components, and the elimination of all unnecessary details in diagrams.

There was no intent on the part of our committee to set up standards for the industry. However, all diagrams appearing in *SIMULATION* will be prepared according to the committee's recommendations (as they may be modified from time to time), and we hope that these recommendations will prove attractive to others. Suggestions for modifications and additions are solicited.

General rules

The following methods and symbols are recommended for the illustration of *technical articles* prepared for *publication*. Unless the purpose of the article is to describe the use of a particular kind of equipment, and the hardware details are pertinent to the subject, such illustrations should *not* be "hardware-peculiar." In other words, the objective should be to show signal flow, rather than "patching" details.

The primary, or overall, system diagram should show only the essential signal flow. Where it is necessary to show details, separate diagrams should be made and referenced to the primary diagram by enclosing the detailed area of the primary diagram in dotted lines with suitable notation.

The direction of signal flow should be indicated by arrowheads except where the shape of the symbols makes the direction of flow obvious. Primary signal flow (with the exception of feedback loops) should be from left to right, and, if practical, each "line" of symbols should be made to read like the mathematical relation it represents.

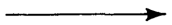
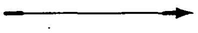
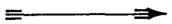
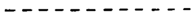
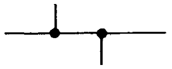
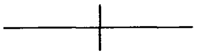
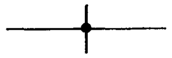
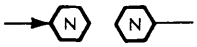
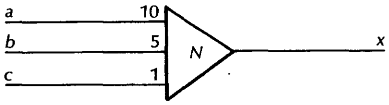
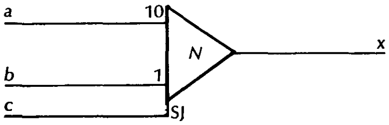
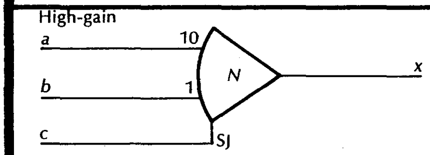
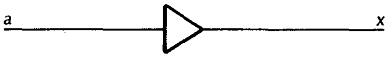
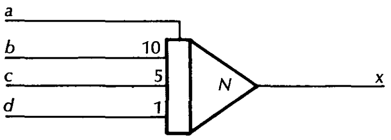
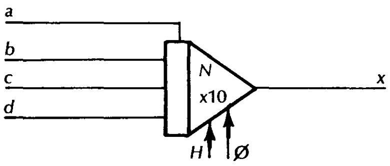
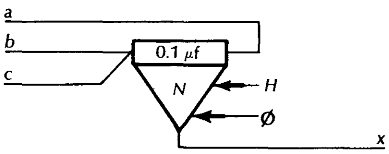
The choice of whether to end a line and label it (preferably with the symbol of the variable that the signal represents) when it reaches the right-hand side of the diagram, and then indicate its continuation with the same label as it enters again at the left-hand side, instead of drawing in the connection, should be made on the basis of clarity; if a line returning the signal from right to left will cross many other lines and be hard to follow, it should not be drawn in.

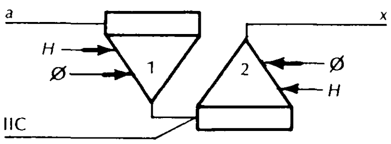
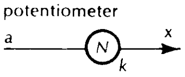
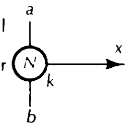
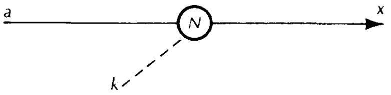
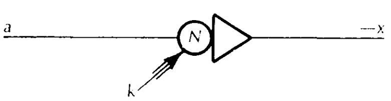
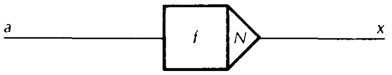
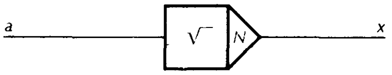
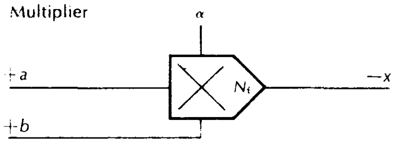
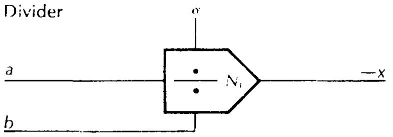
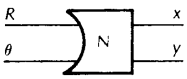
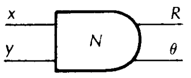
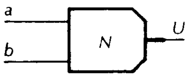
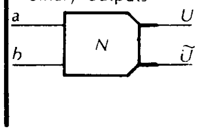
If a diagram involves a number of identical circuits, only one should be shown in detail, while the others should be indicated by boxes with appropriate notation.

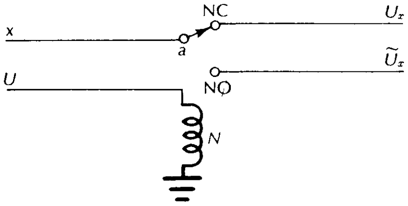
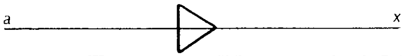
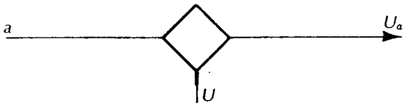
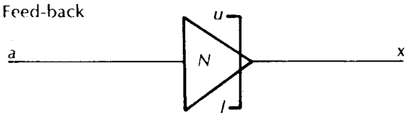
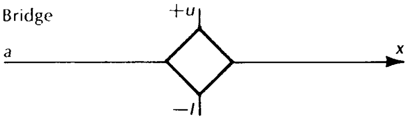
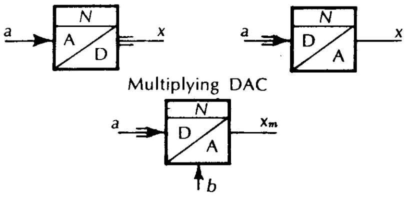
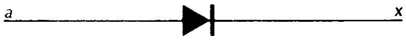
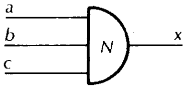
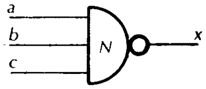
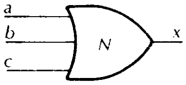
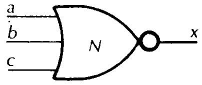
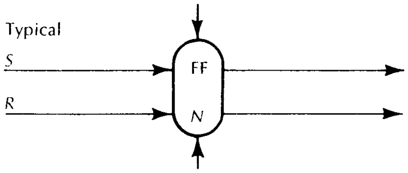
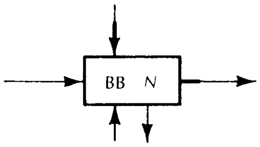
Components should not be numbered unless they are referred to by number in the text.

All amplifier gains should be shown just outside the amplifier at the point where the input enters.

Always apply the test of clarity and simplicity. Ask yourself: "Is this the most understandable way to diagram this *for those unfamiliar with the hardware, and less familiar with the subject, than I?"*

ELEMENT	SYMBOL	FUNCTION	NOTES
Signal flow	Analog  Binary 		Distinction as to kind of signal is made only at beginning and end of line, and then only where there are two or more kinds of signals to be shown on one diagram. Arrowheads should be omitted only if direction of signal flow is obvious.
	Digital word  Mechanical 		
	Connections  No connection 		
	Prohibited  Trunks 		Showing two connections to a line at a single point is prohibited.
Summers	Standard 	$x = -(10a + 5b + c)$	No arrows; shape clearly indicates direction of flow. "N" indicates where number of amplifier should be placed if, *and only if*, referred to in text.
	With one high-gain input 	$x = -(10a + b + \mu c)$	Label high-gain input "SJ" or "G". The point at which c enters (at top or bottom of symbol) indicates a gain of μ, which is a function of the amplifier characteristics and the resistance in the c circuit.
	High-gain 		The curved back indicates that the feedback resistor has been removed.
	Inverting only 	$x = -a$	When amplifiers are used *only* to change signal sign they should be smaller. Orientation should be that which makes for greatest graphical simplicity, *i.e.*, they may be shown in vertical, or in feedback, paths.
Integrators	Standard 	$x = -a - \int_0^t (10b + 5c + d)\, dt$	Standard integrator modes and time scales are those of the basic problem. If they are *not*, the facts should be noted as indicated in the diagram below.
	Three-mode 	MODE H Ø IC 0 0 HOLD 1 0 OPERATE 0 1 HOLD 1 1	Arrows indicating mode control should be shown only when two or more integrators are operating in different modes simultaneously. The mode-control inputs should then be labeled to conform to a truth-table. Because the truth-table will be equipment-peculiar it should be shown somewhere on each patching diagram to which it applies. A typical one is shown. The integrator time-scale, when other than that of the basic problem, should also be as indicated. The amplifier illustrated is integrating at ten times problem rate.
	Track-hold 	TRACK when in IC HOLD when in HOLD (See truth-table)	When used in this way the size of the integrating capacitor may be important, as it affects the tracking lag. In such cases the value should be indicated as shown. The quantity tracked or held is $-(a + b + c)$.

ELEMENT	SYMBOL	FUNCTION	NOTES
Integrators (continued)	Memory pair  		Integrator 1 tracks *a* while integrator 2 holds previous value. A momentary reversal of modes transfers the current value of *a* to *x*. In certain iterative problems *x* is required to have some value for the first iteration; thus an "initial" initial condition must be furnished as shown.
Attenuators	Two-terminal potentiometer  Three-terminal poten-tiometer  	$x = ka$ (load connected) $a = 1, b = 0$ (load connected) ;	The mathematical relationship shown for the three-terminal pot is given only to define *k*. It will not usually hold during problem solution because if *b* were always zero a two-terminal pot could be used. If it is desirable to define *k* in any other way, its meaning should be clearly indicated.
	Servo-set potentiometer  	$x = ka$	This should be used only if the fact that *N* is servo-set is significant to the solution of the problem, as, for instance, when it must be automatically reset after each run of an iterative problem.
	Digital attenuator  	$x = k^{*}a$	As shown, the symbol indicates a digital pot with integral (committed) inverting amplifier. The symbol for non-inverting digital pots should not include the amplifier. *Indicates quantized value.
Function generators	Arbitrary  	$x = f(a)$	The number *N* should be used only when the function generator is referred to by number in the text, or when it is desirable to explain what the function *f* is, either by a footnote, or by a separate graph.
	Mathematical functions (typical)  	$x = \sqrt{a}$	In cases where the function generated can be represented by a standard mathematical symbol, the *f* should be replaced by the symbol.
	Multiplier  	$x = \frac{ab}{\alpha}$ $x = \frac{\alpha a}{b}$	These symbols should be used for all analog multipliers and dividers; if the *kind* is significant to the solution of the problem, it should be so stated in the text and/or noted on the diagram. Because equipment differs, the sign of the output for specified signs of both inputs should be given, otherwise inversion will be assumed. The number *N* should be used only if the component is referenced in the text, or if more than one channel of a multichannel device is used. In the latter case the subscript, in this case *i*, identifies the channel. In many cases, it will be found that a drawing can be simplified and clarified by drawing the same multichannel device in more than one signal flow path. In this case the number would be the same, but the subscript would be different for each channel.
	Divider  	α = some reference voltage, not necessarily the computer reference voltage	
	Resolvers Polar to rectangular  Rectangular to polar  	$x = R \cos \theta$ $y = R \sin \theta$ $\theta = \arctan y/x$ $R = \sqrt{x^2 + y^2}$	If a resolver has additional outputs they should be shown only if used.
	Comparators With binary output  With true and false binary outputs  	$U = 1 \quad (a + b) > 0$ $U = 0 \quad (a + b) < 0$ $\tilde{U} \neq U$	The symbol for a relay comparator can, of course, be made by combining either of these symbols with that of a relay.

ELEMENT	SYMBOL	FUNCTION	NOTES
Relay		$U_x = x \quad (U = 0)$ $\tilde{U}_x = x \quad (U = 1)$	If the diagram is drawn as shown, the designations of arms and contacts are superfluous. However, it is often desirable, for clarity, to show the relay coil in one place on a diagram and the contacts in signal-flow paths elsewhere. In this case, N_{oa} would designate the normally Open $(U = 0)$ contact of channel a, relay N; N_{aa} the Arm of channel a of relay N, etc.
Digital inverter		$x = \tilde{a}$	The tilde (~) is used instead of a bar to indicate negation. This is recommended because of the many other meanings of a bar over a variable.
Digital/analog switch		$U_a = 0 \quad (U = 0)$ $U_a = a \quad (U = 1)$	ON when binary signal, U, is digital "one."
Limiters	Feed-back 	$x = -a \quad (l < a < u)$ $x = -l \quad (a < l)$ $x = -u \quad (a > u)$	u = larger value at which output is limited l = smaller value at which output is limited
	Bridge 	$x = a \quad (l < a < u)$ $x = l \quad (a < l)$ $x = u \quad (a > u)$	
Converters	Analog-to-digital Digital-to-analog Multiplying DAC N a A D x N a D A x N a D A x_m b	$x^* = a$ $x = a^*$ $x_m = a^*b$	*Indicates quantized value.
Diode	Solid state 	$x = a \quad (a > x)$ $x = 0 \quad (a < x)$	
Logic gates	AND 	AND $x = a \cdot b \cdot c$	Output is high (logic 1) if and only if all inputs are high.
	NAND 	NAND $x \neq a \cdot b \cdot c$	Output is low (logic 0) if and only if all inputs are high.
	OR 	OR $x = a + b + c$	Output is high (logic 1) if and only if at least one input is high (*i.e.*, any or all of the inputs are high).
	NOR 	NOR $x \neq a + b + c$	Output is low (logic 0) if and only if at least one input is high (*i.e.*, any or all of the inputs are high).
Flip-flop	Typical 		As there are many kinds of flip-flops, symbolic representation of the operation of any but the simplest is not advised; if the operation is not obvious, an explanation in the text or a footnote with the diagram is recommended.
Black box		This is Black Box number N to be used if no symbol is given here, *or in case of doubt!* It can have any number of inputs and outputs, but the nature of the signals should be indicated and all signal-flow directions should be designated by arrowheads. Here there are three analog inputs giving rise to one analog and one binary output.	The function of a "black box" can often be made obvious by properly labeling the inputs and outputs. However, if it is not obvious, the function should be explained in the text or by a footnote with the diagram. If internal details are of interest, they should be shown in an appropriately labeled auxiliary diagram.

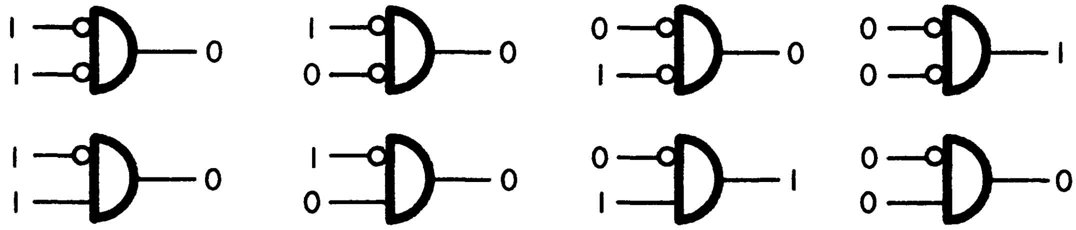
4.1.2 Additional Standards

Emitter-Follower Override: Defined in Chapter 3, Section 3.0.

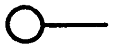
****

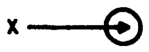
Negative AND Gate

Some gates are structured to negate one or more inputs. These inputs are indicated by circles.

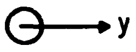


****

 internal connection to patchboard hole.



patchcord with signal "x" flowing into the hole.



patchcord with signal "y" flowing out of the hole.

The concept of signal flow direction should not be confused with voltage or current polarities. Signals flow from sources to destinations without regard to polarity or logical state.

****

Unpatched inputs to logic computing devices may be regarded as a logic zero. Unpatched inputs to logic control devices, however, such as integrator controls, may not automatically be considered to behave as if they were logic zero, and often need to be patched to a logic zero in order to obtain the desired action of the various devices.

****

The tilde (~) is not used to indicate negation (complementation) of logic variables in this manual as recommended by SCi. The reason is that the tilde is not available in standard Le Roy lettering guides except in 12 point size. A bar above a logic variable ($\overline{A}$) is used instead, and is exclusively reserved for this purpose.

****

4.2 COMPUTING ELEMENTS

Section 4.2 describes the patching configurations necessary to implement operation of the individual elements of the AD/FIVE. The purpose of this section is to describe the signals which are patched into and out of each patchboard hole on the AD/FIVE. The section is therefore not a manual of analog programming, but is instead a description of the patching unique to the AD/FIVE. Section 4.2.1 describes the patching of the analog computing elements and the analog/logic computing elements. Section 4.2.2 describes patching of the logic computing elements. Section 4.2.3 describes the patching of miscellaneous computing elements. Section 4.2.4 gives a resume of control patching locations and of all other patchholes for analog usage which are not covered in the previous sections of this chapter. Patchholes reserved for hybrid usage are discussed in the AD/FIVE manual for the hybrid use of the computer system. The patching directions for each component are accompanied by an insert showing the standard patchboard locations of the terminations for the component shaded in solid black, and showing optional locations for the component shaded in diagonal hatching. There is also an illustration showing the actual patchboard holes of each component and the input and output connections to these holes. Finally, there is an illustration of the standard flow diagram symbol for the component, and any necessary additional information such as mathematical formulas, etc. Descriptions of the internal circuitry of the components have been deliberately avoided in this chapter in order to simplify the task of the programmer by giving him only that information which he needs in order to properly patch any analog program. The details of the internal circuitry are important for maintenance purposes and can be found in the volumes of the AD/FIVE Service Manual.

4.2.1 Analog Computing Elements

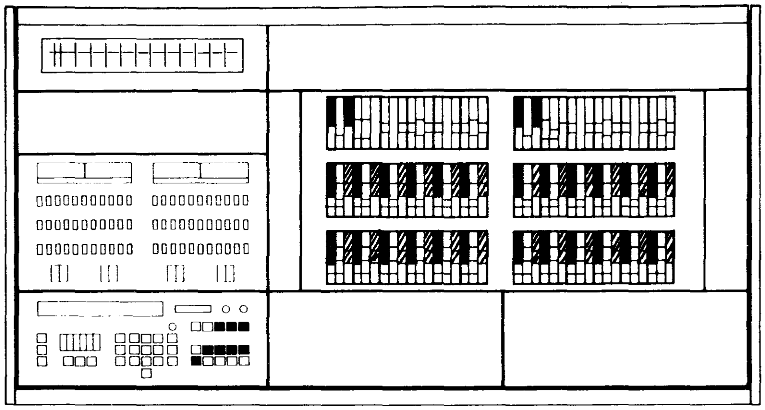
Analog computing elements are those components whose signal terminations are found in the lower two thirds of the patchboard. Analog/logic computing elements have, in addition to terminations in the lower two thirds of the patchboard, control terminations for each component in the upper third of the patchboard, which contains terminations for the logic computing components.

4.2.1.1 Integrator/Summers

There are normally two integrator/summers in each area of the AD/FIVE. Two expansion integrators may be added to each area, making a maximum total of four integrators possible per area. The two integrators in the standard locations have individual overload indicators in the Overload Indicator Panel, and overriding individual mode and gain controls. The two expansion integrators have individual overload indicators and individual X10 gain controls. They do not, however, have individual mode controls or individual X100 gain control. They are instead slaved automatically to the integrators already present in the same horizontal row in their area. For example, the logic control address 05 controls the integrator with the number 5, and the expansion integrator number 8 in area 0.

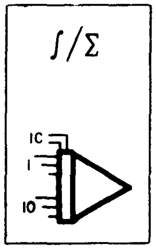
The integrator/summers are configured as integrators by patching C to CJ. In the integrator configuration the SJ terminal cannot be used as an input, since it is not switched by the integrator mode controls. (See the AD/FIVE Service Manual for circuit details.) The high-speed electronic mode-control switch is normally controlled by the console operate and hold busses, but may be controlled individually for each integrator by patching the individual H and OP patchboard input terminals. Integrator gain is normally controlled by the console X10 and X100 busses, but may also be individually controlled for each integrator by patching the individual X10 and X100 input terminals on the patchboard. All individual control of integrators is achieved by means of emitter-follower override of the console busses. Whenever the individual mode or time scale control is used, the individually patched integrator is no longer under console control. If the operator uses the X10 or X100 pushbuttons to speed up the solution, then he must be aware that any integrator which he has patched for individual X10 or X100 control to achieve required gains will not normally respond to the pushbutton control. (However, see Chapter 5, Section 5.2.1.1 for a method of patching an individual integrator to overcome this restriction.) The following section describes the various means of using the time scale pushbuttons or individual integrator patching to achieve time scale/gain control.

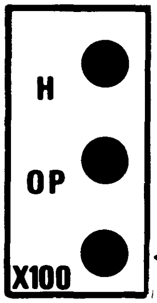
* See Chapter 5, Section 5.2.1.1, for an interesting variation.



LOGIC ANALOG

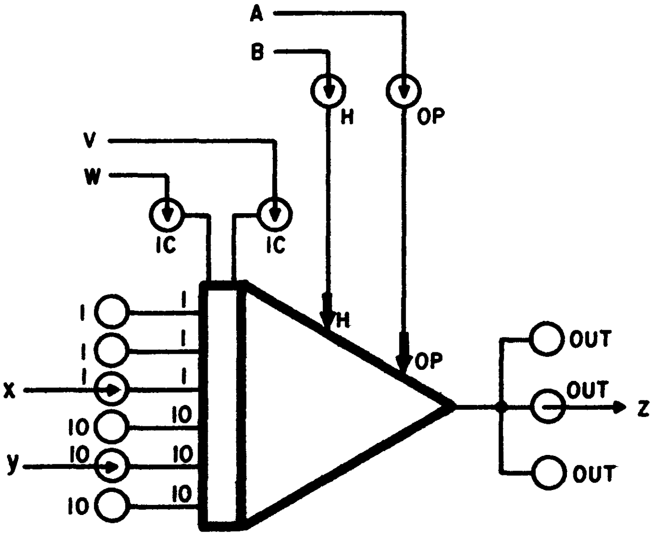
INTEGRATOR GAIN AND MODE CONTROLS



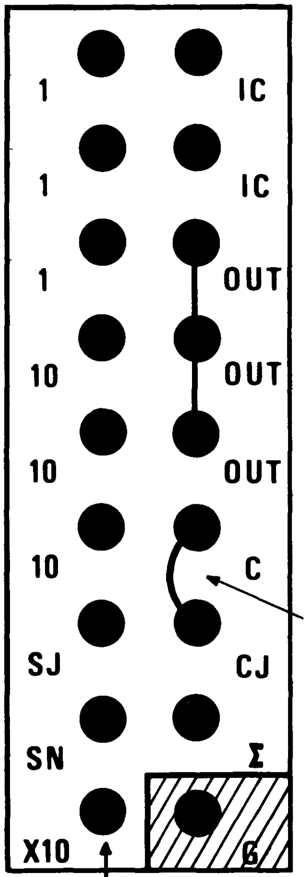


To lock an individual integrator at gain one so that it will be independent of console gain controls, patch X100 to a logic zero, and X10 to a logic zero or plus reference

Patch logic one to multiply the below gains by 100



When individual integrator mode control is patched, then the mode is determined by the inputs to A and B. It will often be necessary to patch A or B to a logic zero in order to override console control of the integrator. It cannot be assumed that if A or B are unpatched that they will behave as if patched to logic zero. If A is patched to a logic one, for example, but console mode is Hold, then the integrator will not operate until B is patched to a logic zero.



Patch as shown

Patch logic one (or ground) to multiply the above gains by ten

MODE	Value of Z
IC	$-(V + W)\ ;\ (T = 0)$
OPER	$-\int_0^t (X + 10\,Y)dt - (V + W)$ ← *
HOLD	value at time of entry into HOLD mode

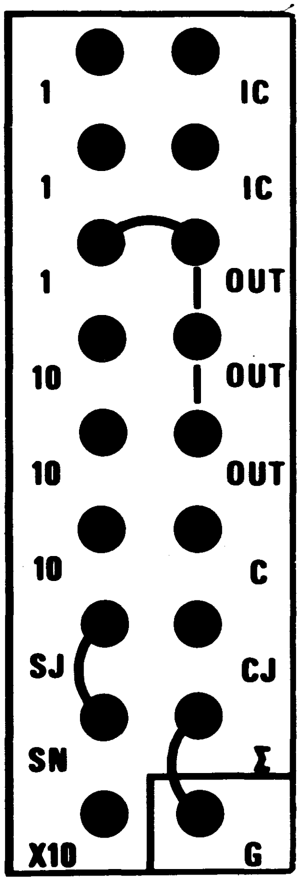
* If X10 is patched $Z = -10 \int_0^t (X + 10Y)dt - (V + W)$

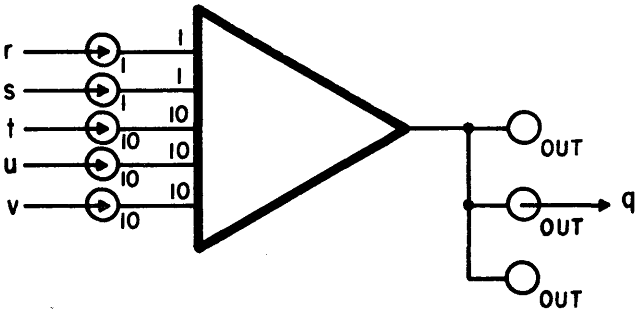
If X100 is patched $Z = -100 \int_0^t (X + 10Y)dt - (V + W)$

If X10 and X100 are patched $Z = -1000 \int_0^t (X + 10Y)dt - (V + W)$

Patching a Summer/Integrator as an Integrator

Figure 4-1





$$q = -(r + s + 10t + 10u + 10v)$$

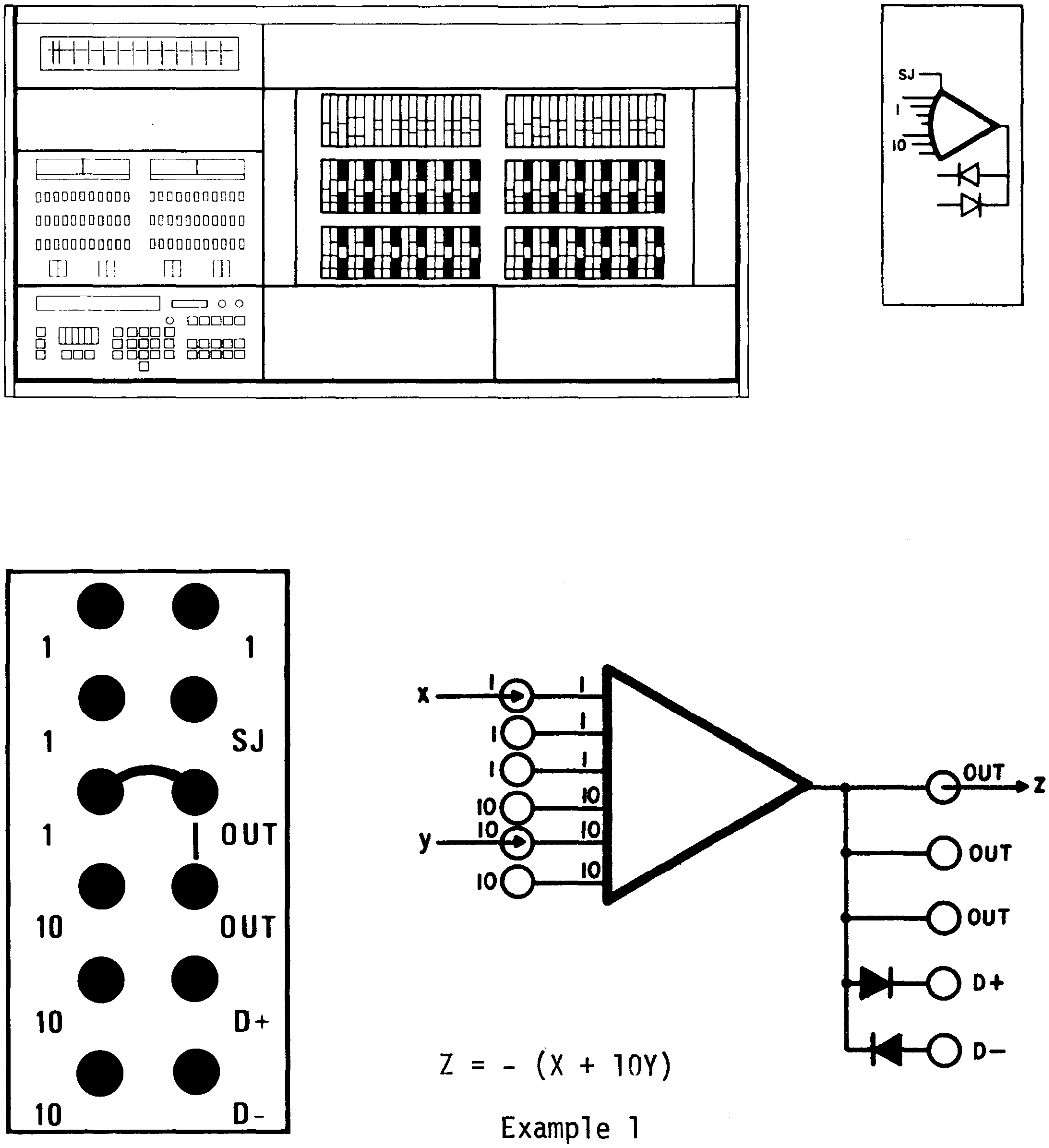
Note: The actual forward gain of each input on the amplifier is the reciprocal of the sum of the patched amplifier feedback gains multiplied by the labelled gain of that input. See Fiqure 4-3, Examples 1, 2, and 3 for details. See also Chapter 5, Sections 5.1.4.1 and 5.1.4.2 for discussions of amplifier gain and feedback.

Patching a Summer/Integrator as a Summer

Fiqure 4-2

4.2.1.2 Summer

Each AD/FIVE area may contain four summers. Each summer has an individual overload light, inputs and outputs, a summing junction connection, and two diodes attached to the outputs. Several examples in Chapter 5 illustrate the use of these diodes. Note that the amplifier is terminated as a high gain operational amplifier. One or more of the inputs is connected to the output to configure the amplifier as a summer. The remaining inputs will have gains which are the labelled gains multiplied by the reciprocal of the sum of the feedback gains. Three examples are shown in Figure 4-3.

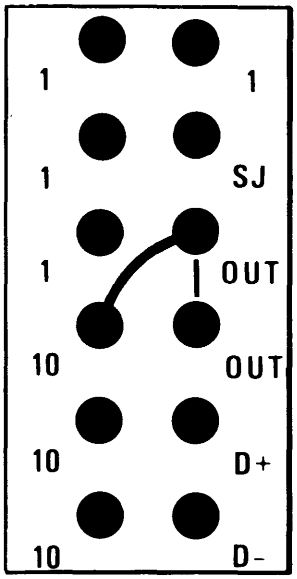


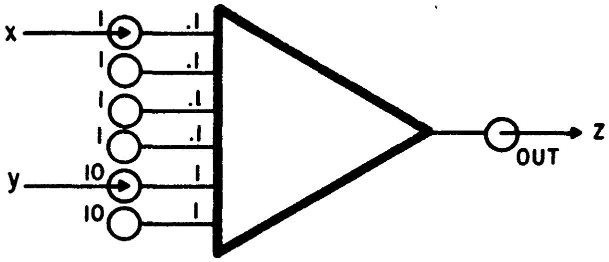
$Z = -(X + 10Y)$

Example 1

Patching a Summer

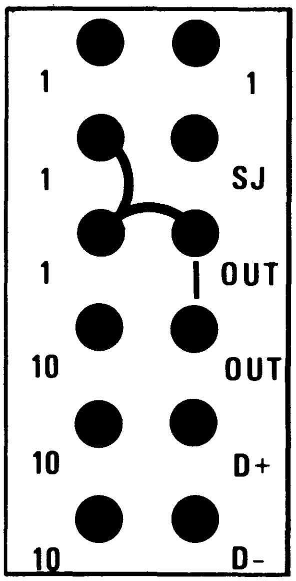
Figure 4-3

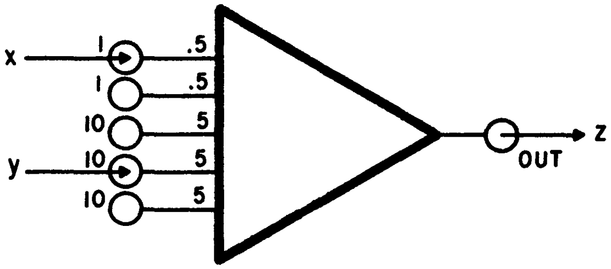




$Z = -(.1X + Y)$

Example 2





$Z = -(.5X + 5Y)$

Example 3

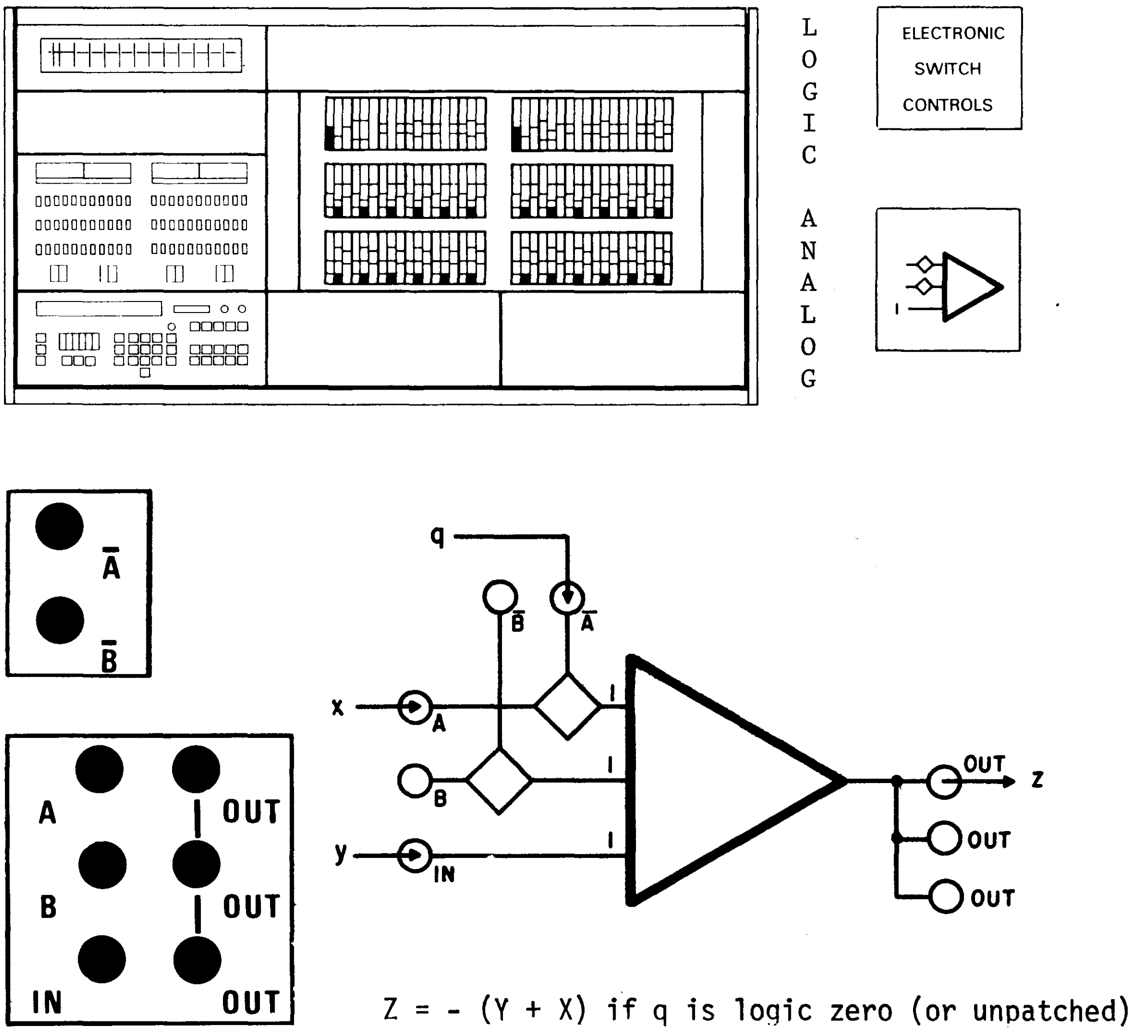
Patching a Summer

Figure 4-3

4.2.1.3 Switch/Summer

Each AD/FIVE area contains two Switch/Summers. Each Switch/Summer has two logic-controlled electronically switched inputs. A third, unswitched gain one input is provided. An individual overload indication is provided for the amplifier.

Note that the logic control holes are labelled with the complement of the control signals. This means that the switches in the A and B inputs are "on" unless turned "off" with a logic one. Therefore, all three gain-one inputs are useful with no logic patching required at all. Logic control is patched only when needed.



Z = - (Y + X) if q is logic zero (or unpatched)

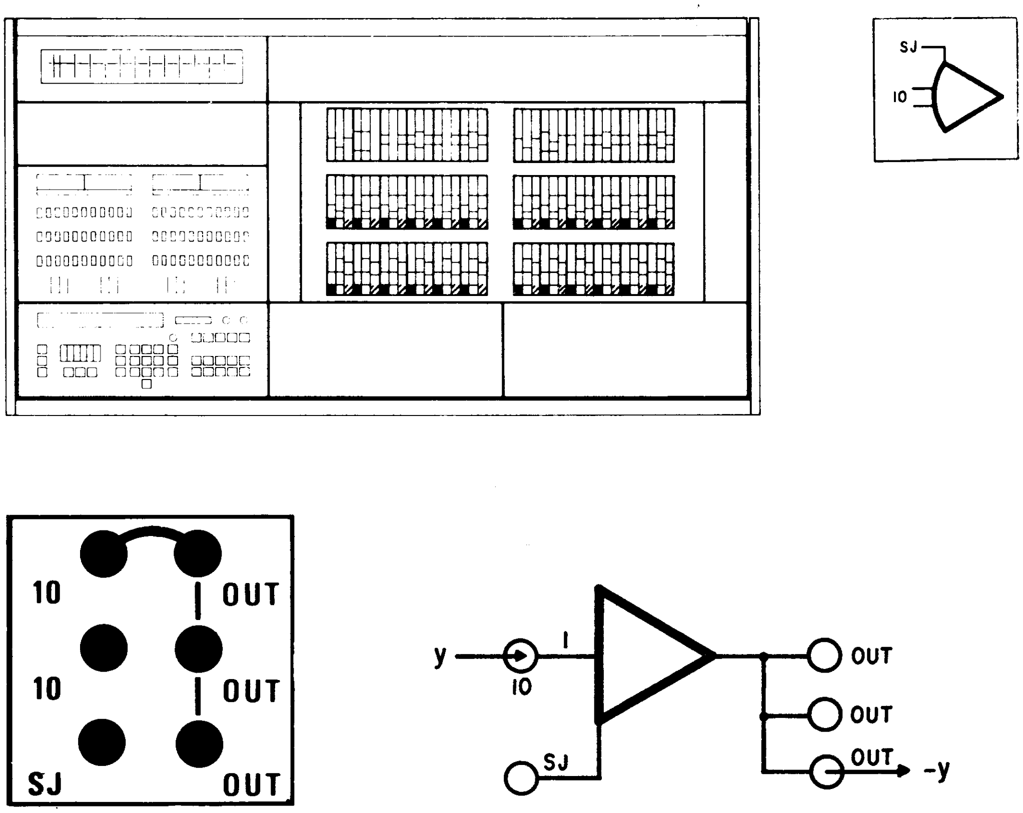
Z = -Y if q is a logic one

Patching a Switch/Summer

Figure 4-4

4.2.1.4 Inverter

Inverters are located beneath the integrators. Gain 10 inputs are used for better dynamic performance in the inverter configuration.



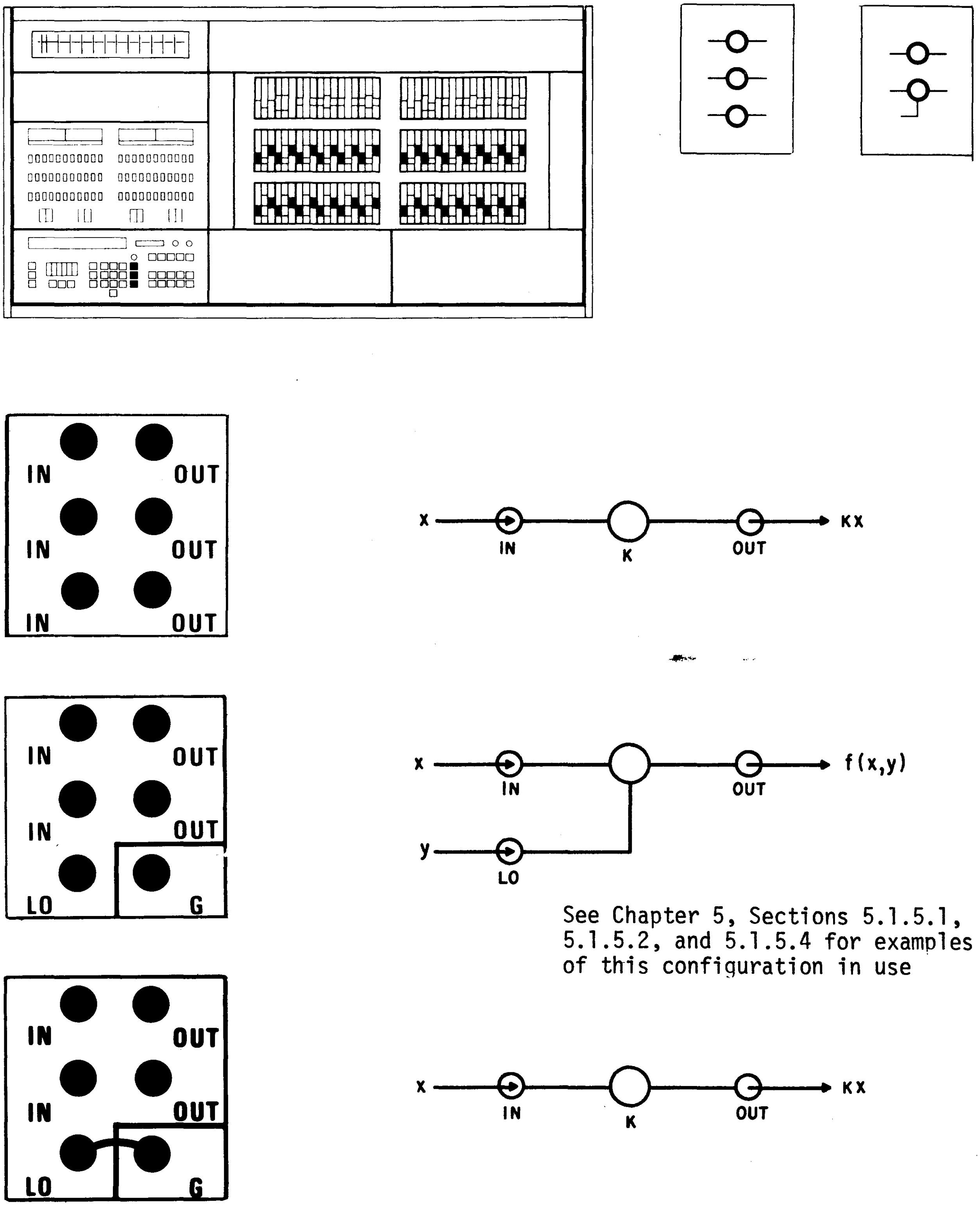
Patching an Inverter

Figure 4-5

4.2.1.5 Potentiometer

Ten potentiometers are provided in each area of the AD/FIVE. Various mixtures of hand-set and servo-set pots are provided. Two of every ten pots are terminated at the patchboard as ungrounded pots.

Potentiometers are devices which multiply an input value by a constant. The setting of the value of this constant is discussed in Chapter 3.



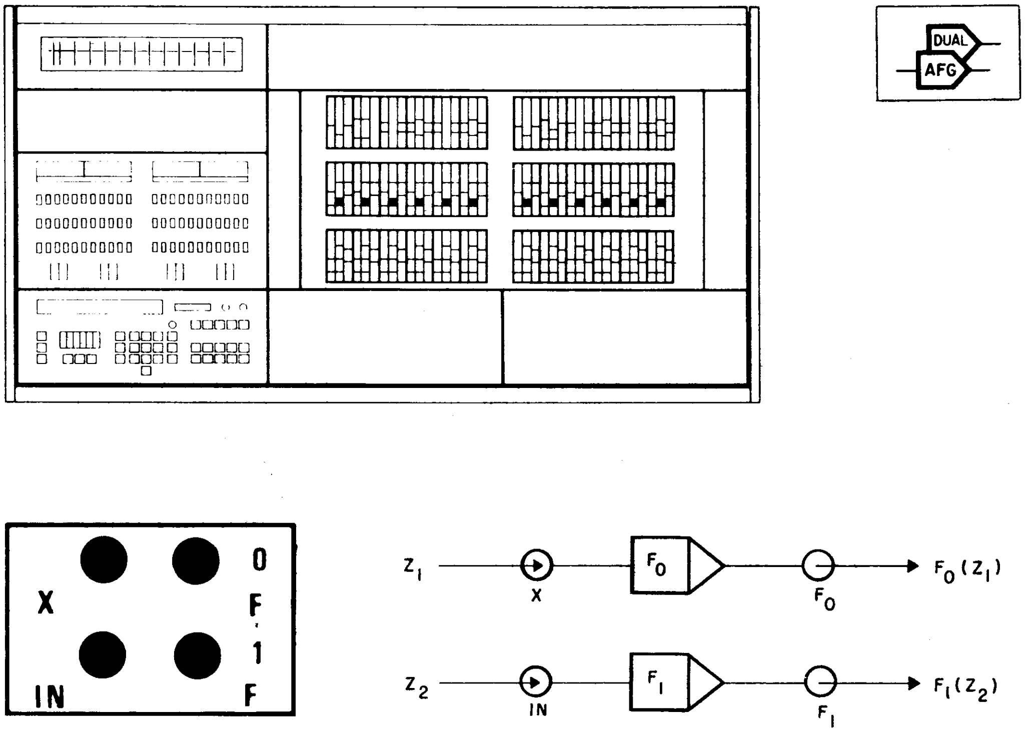
See Chapter 5, Sections 5.1.5.1, 5.1.5.2, and 5.1.5.4 for examples of this configuration in use

Patching a Potentiometer

Figure 4-6

4.2.1.6 Arbitrary Function Generators

Each AD/FIVE area may contain one dual Arbitrary Function Generator. The device is complex to set up, compared to all of the other devices discussed in this chapter. Hence, the description of the various options associated with the Arbitrary Function Generator is to be found in an appendix to this manual.



Patching an Arbitrary Function Generator

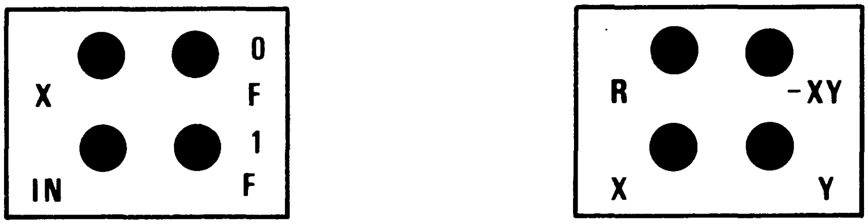
Figure 4-7

4.2.1.7 Convertible Multiplier

Each analog area contains the terminations for one Convertible Multiplier. This multiplier may be converted into a pair of square or square-root function generators. A separately addressable internal amplifier, which is required in order to provide a pair of such function generators when converted, is released when the device is used to multiply or divide, as shown in Figure 4-9.

Note: Non-convertible multipliers (Section 4.2.1.8) may be installed in the slot used for the convertible multiplier. If this is done, all use of the IN, OUT, and CVT terminals shown in Figure 4-9 is ignored. Of course, only the multiply and divide configurations can then be achieved.

Note: Either the convertible or the non-convertible multipliers may be installed in the Arbitrary Function Generator card slots as an option to increase the multiplier complement. If this is done, then the terminations will be as shown in Figure 4-8.



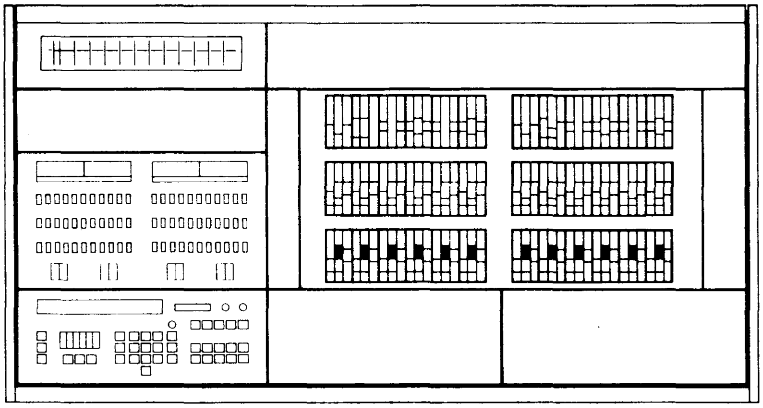
Labelling of Function Generator Area

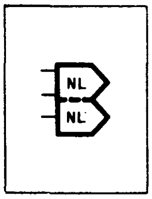
Equivalent Labelling for Multiplier

Multiplier Terminated in a Function Generator Slot

Figure 4-8

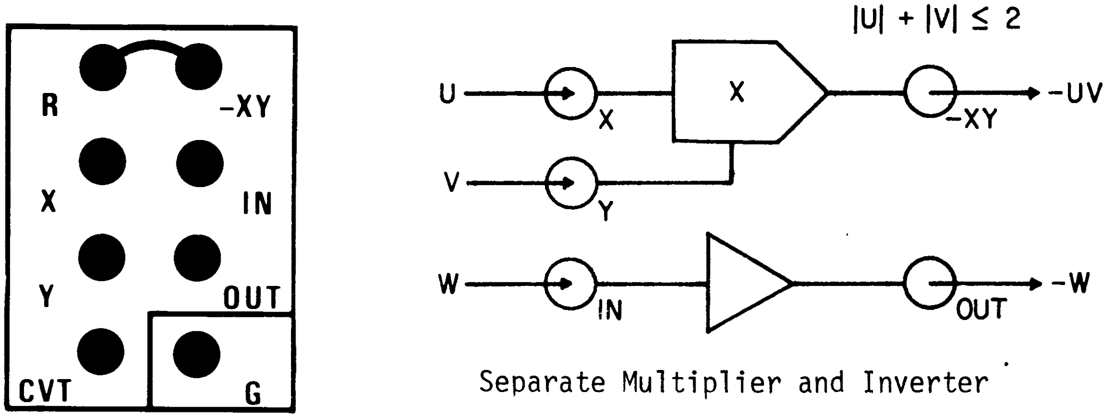
Note that the equivalent terminations for a multiplier in an Arbitrary Function Generator slot are identical in position to the terminations of a standard non-convertible multiplier slot. It is thus not necessary to refer to this figure in order to patch the multiplier. The operator may simply look at the labelling of a standard multiplier slot and patch the multiplier in the arbitrary function generator slot in the same configuration, using a one-to-one correspondence.





CAUTION: The following restrictions must be observed when patching the convertible multiplier in order to obtain meaningful results:

$|U| + |V| \leq 2$

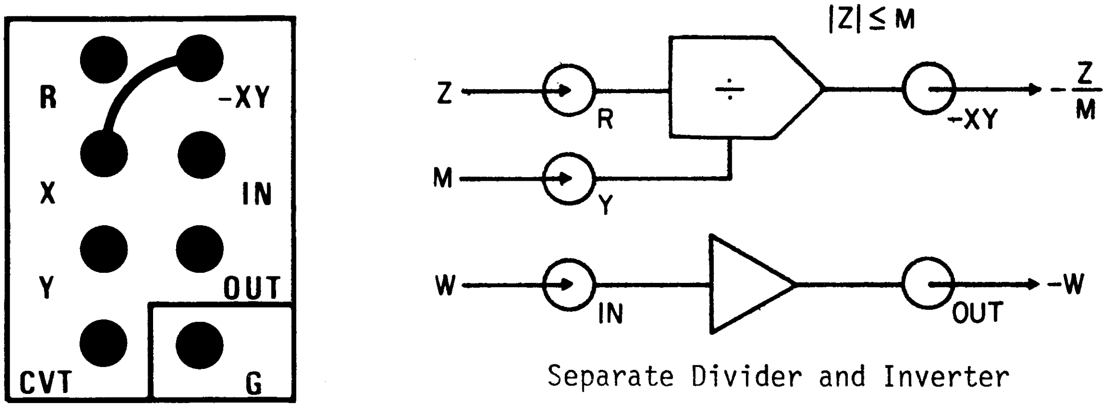


Separate Multiplier and Inverter

CAUTION: The following restrictions must be observed when dividing in order to obtain the proper results:

$M > 0$

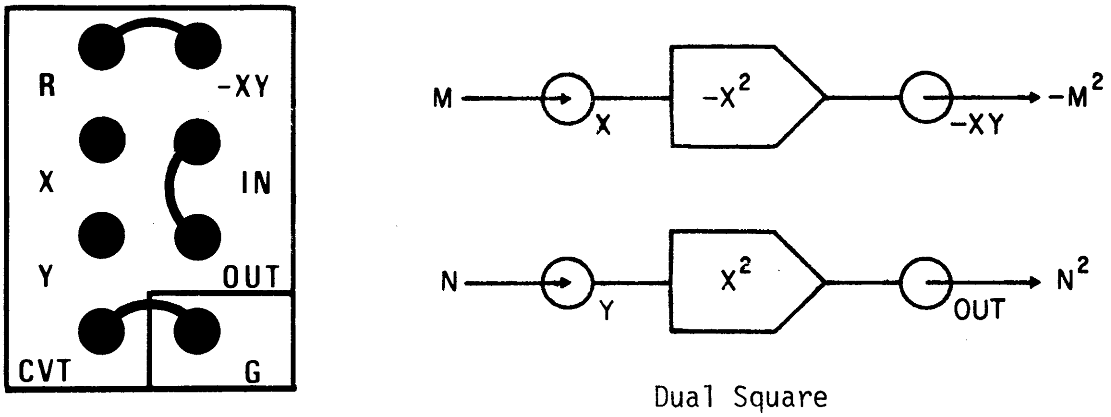
$|Z| \leq M$



Separate Divider and Inverter

Care must be taken to avoid values of Z and M too near zero, as inaccurate results will be obtained due to inherent sensitivity of the circuit to noise for very small input signals.

(Some overrange is allowed. The amount varies with the individual unit.)

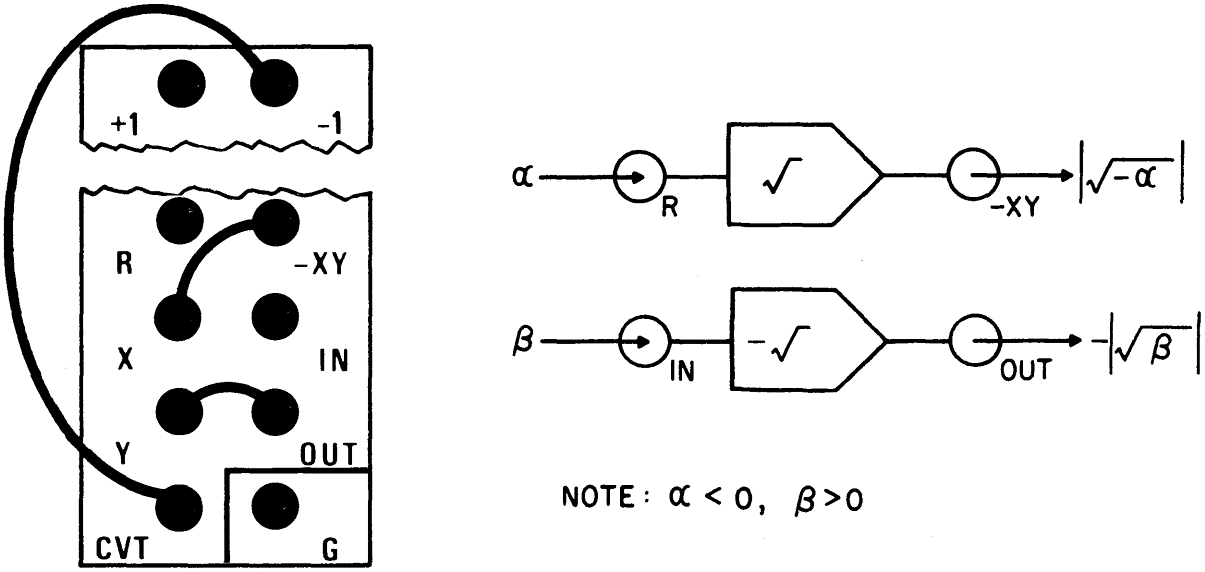


Dual Square

Patching a Convertible Multiplier

Figure 4-9

CAUTION: Care must be taken when patching for dual square root to observe that:



NOTE: $\alpha < 0, \ \beta > 0$

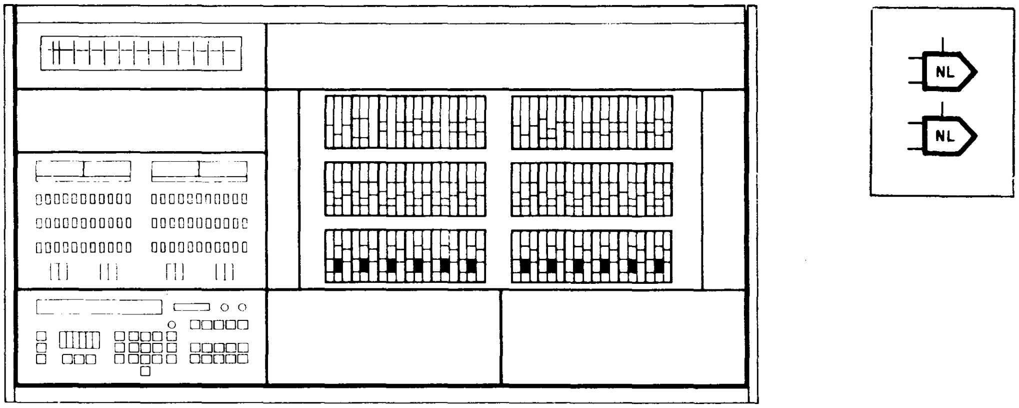
Dual Square Root

Patching a Convertible Multiplier

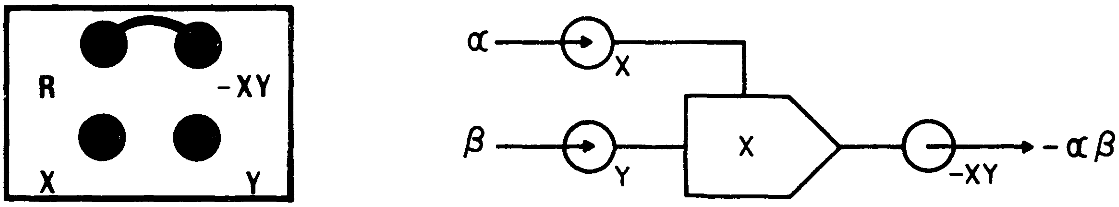
Figure 4-9

4.2.1.8 Multiplier

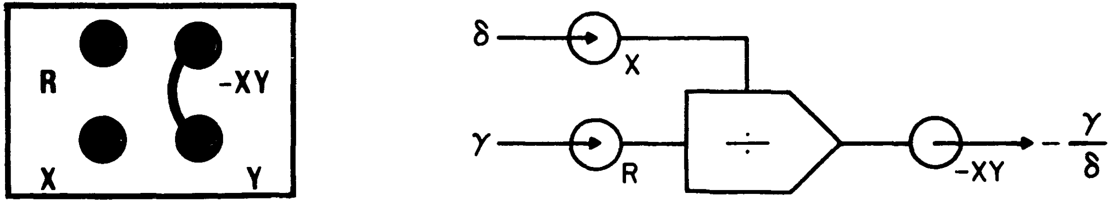
Each AD/FIVE area has provision for up to four non-convertible multipliers. Patching for multiplication and division is shown in Figure 4-10 for the two normally assigned slots in each area. This multiplier may be substituted for the convertible multiplier. If this is done, see the note in Section 4.2.1.7. In addition, this multiplier may be substituted for an Arbitrary Function Generator. If this is done, see the note in Section 4.2.1.7, and Figure 4-8.



CAUTION: $|\alpha|+|\beta| \leq 2$ must be observed when patching the multiply configuration.



$\delta \geq 0$
$|\gamma| \leq \delta$ must be observed when patching the divide configuration. Care must be taken to avoid values of D and E near zero.



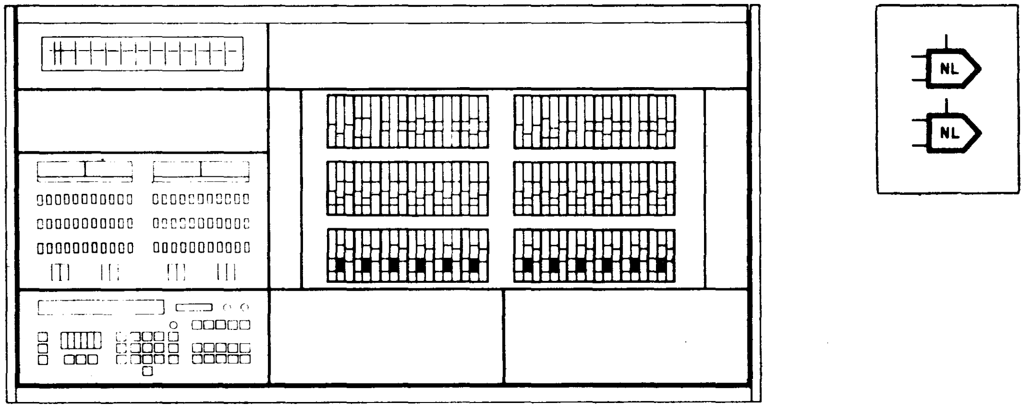
(Some overrange is allowed. The amount varies with the individual unit.)

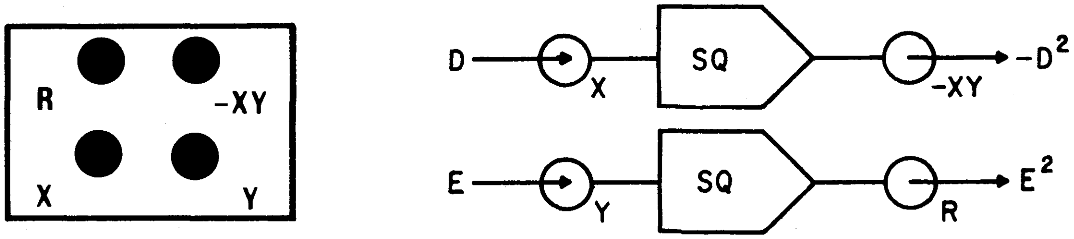
Patching a Multiplier

Figure 4-10

4.2.1.9 Dual Square Function Generator

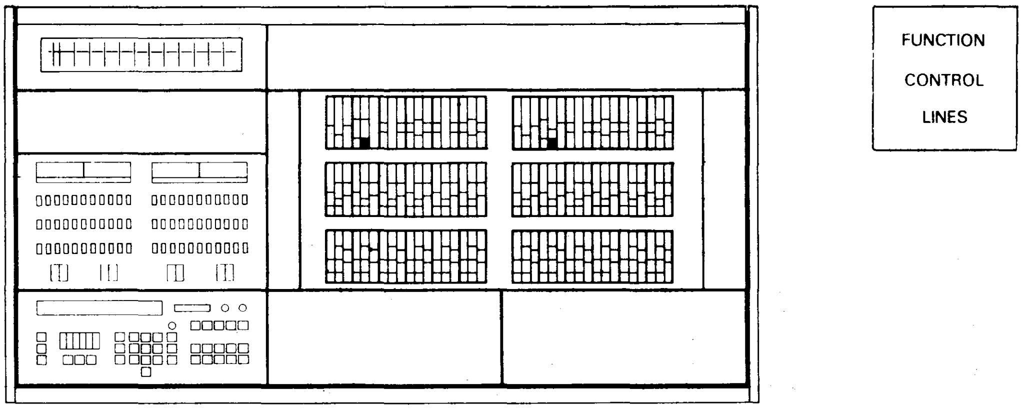
A pair of fixed function generators, whose outputs are the square of their inputs, can be substituted for the multiplier. Patching is as shown in Figure 4-11 and 4-12.

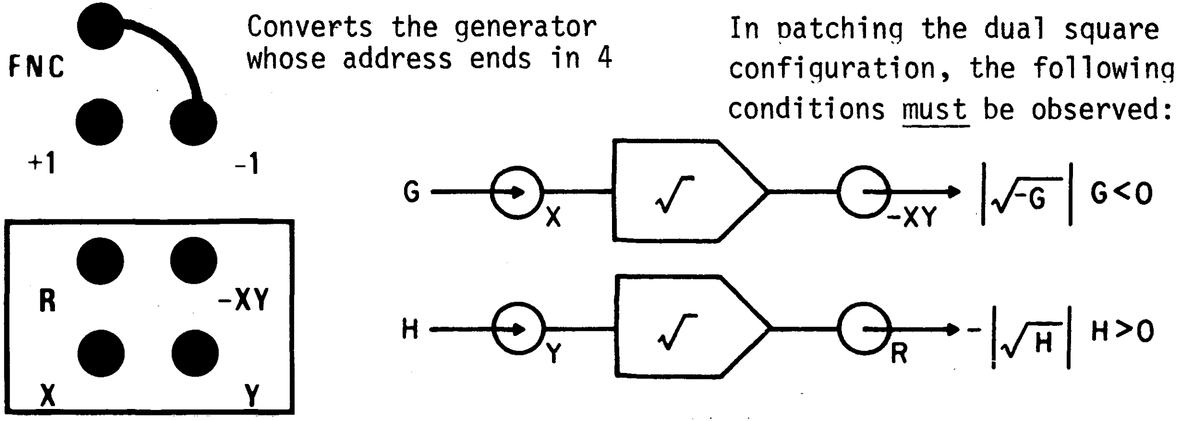




Patching a Dual Square Function Generator

Figure 4-11




FNC
+1
-1
R
-XY
X
Y
G
X
$\sqrt{\ }$
-XY
$\left|\sqrt{-G}\right|$ $G<0$
H
Y
$\sqrt{\ }$
R
$-\left|\sqrt{H}\right|$ $H>0$

Converts the generator whose address ends in 4

CAUTION:
In patching the dual square root configuration, the following conditions <u>must</u> be observed:

Patching the Dual Square Function Generator for Square Root

Figure 4-12

Note: This function generator can also be used in the convertible multiplier or Arbitrary Function Generator slots. If used in the Convertible Multiplier slot, the IN, OUT, and CVT terminations are ignored. If used in the Arbitrary Function Generator slot, see Figure 4-8 for patching equivalence.

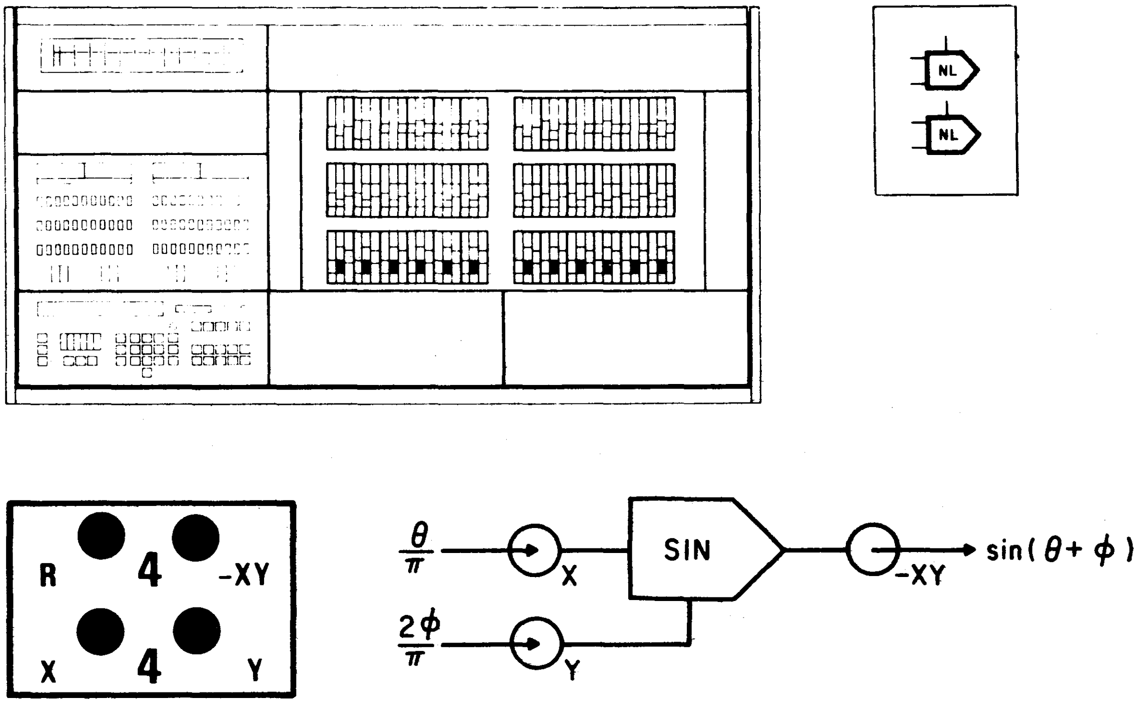
Note: To address the output at the -XY termination, the non-linear slot is addressed as labelled. To address the output at the R termination, the non-linear slot as labelled PLUS ONE is addressed.

4.2.1.10 Sine Cosine Function Generator

The Sine Cosine Function Generator is a fixed function generator employed to generate trigonometric functions of a dependent variable or variables. Patching is as shown in Figure 4-13.

CAUTION: The following conditions must be observed when patching the Sine Cosine Function Generator: $-\pi \leq \theta \leq \pi$

$$-\pi/2 \leq \phi \leq \pi/2$$



θ and ϕ are expressed in radians

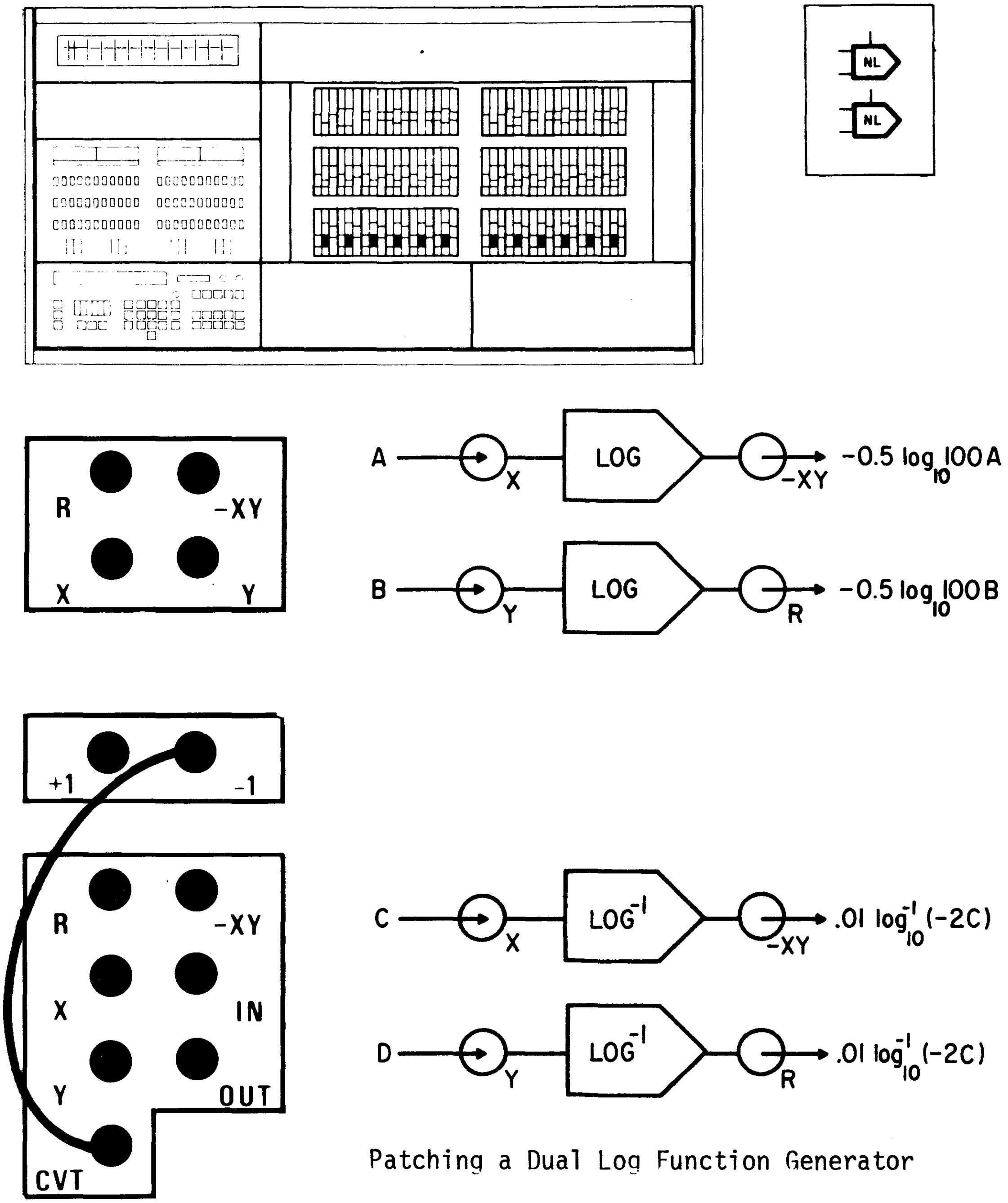
Patching a Sine Cosine Function Generator

Figure 4-13

Note: The Sine Cosine Function Generator can be mounted in place of the Arbitrary Function Generator or the Convertible Multiplier. In the case of the Convertible Multiplier slot the IN, OUT, and CVT holes are ignored. See Figure 4-8 for the equivalent patching of the Sine Cosine Function Generator in the Arbitrary Function Generator slot.

4.2.1.11 Log Function Generator

The Log Function Generator is a four-decade fixed function generator. It can be substituted for any of the multipliers or the Arbitrary Function Generator. Patching and a mathematical definition are given in Figure 4-14.



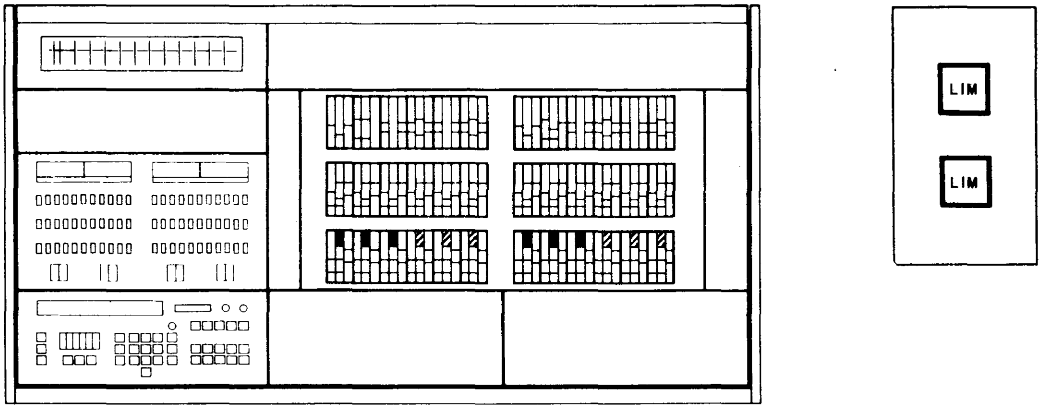
Patching a Dual Log Function Generator

Figure 4-14

If the log generator is mounted in a convertible multiplier slot, the anti-log function may be obtained as shown, by patching CVT to -1. Otherwise the R, X, Y, -XY terminations mean the same for either mounting slot. If mounted in the Arbitrary Function Generator slot, patching follows the equivalent terminations depicted in Figure 4-8. The log cannot be patched in the feedback of a high-gain amplifier to get the anti-log function. The anti-log function can only be obtained by use of the CVT to -1 patching.

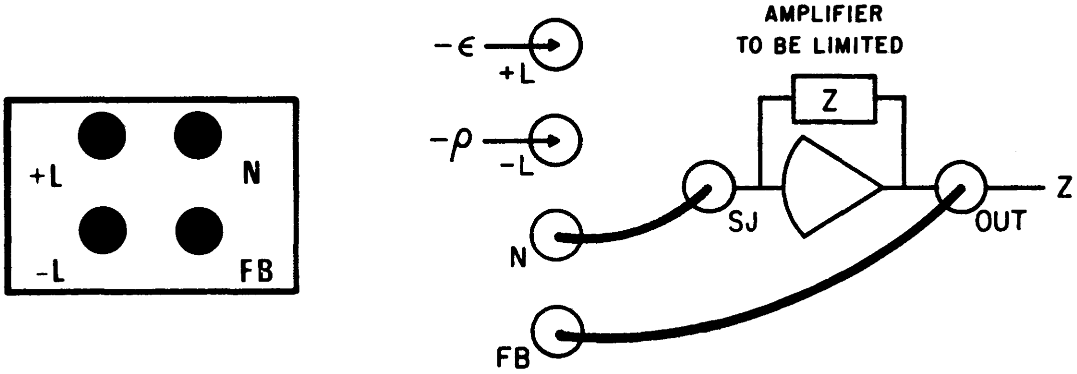
4.2.1.12 Limiter

The purpose of the Limiter is to limit the output of an operational amplifier within preset upper and lower values, and then to hold those values even for large input value changes which would normally command output values in excess of the preset limits. The Limiter is therefore a device which is "attached" to an operational amplifier whose output is to be held between desired programmable limits. Six Limiters per field are the normal AD/FIVE complement, but six additional Limiters may be added in the optional areas indicated by the hatched shading in Figure 4-15, making a total of twelve Limiters per field.



Note; X is defined as the normal value of the problem variable input to the limited amplifier as held between the limits of ε and ρ. Both the upper and lower limit terminations +L and -L must be patched in order for the Limiter to be operational. The following conditions must also be observed: $\rho \leq Z \leq \epsilon$

$$\rho < \epsilon$$



Patching a Limiter

Figure 4-15

Note: Limiters are not intended for use with integrators. It is doubtful if there is any physical system which is correctly simulated by a limited integrator. To appreciate this, consider an integrator whose output represents displacement. The input would represent velocity. Limiting the integrator to simulate a mechanical limit stop would restrict the maximum displacement but would leave the acceleration and velocity unaffected, which is clearly erroneous.

Therefore, the dynamic performance of this limiter has been optimized for use with the summers. Alternate circuits are provided in Chapter 5.

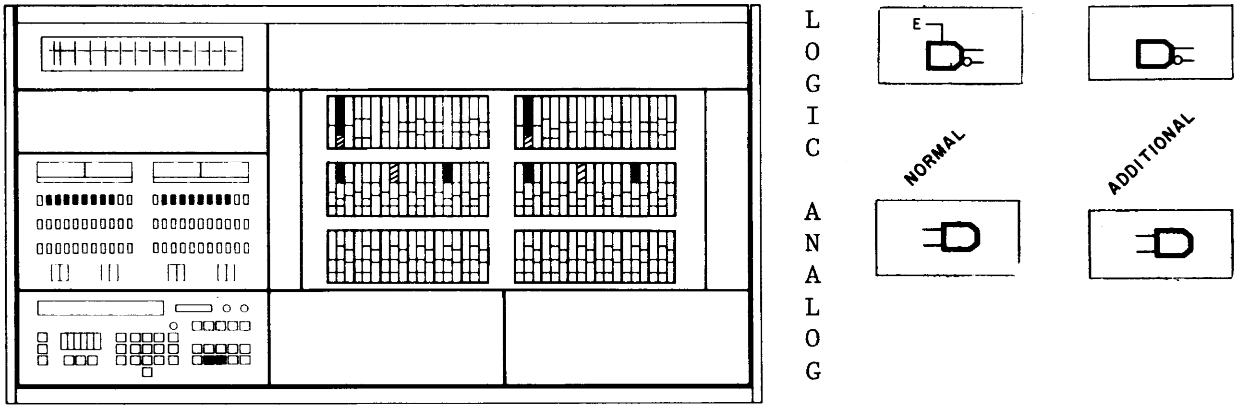
4.2.1.13 Comparator

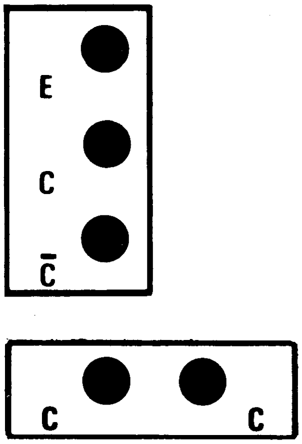
Each AD/FIVE field normally contains eight comparators. Each comparator has two analog inputs, a logic output and its complement, a logic enable control input, and a single-pole triple throw toggle switch and indicator lamp located on the logic control panel (See Section 4.2.2). A comparator compares two analog quantities and provides a logic one output if the sum of the two quantities is greater than zero, a logic zero output if the sum of the two quantities is less than zero. A logic zero patched into the enable input allows the programmer to latch (freeze) the output when desired. The three position toggle switch allows the operator to select an arbitrary output of logic one or logic zero. Note that the toggle switch overrides the operation of the comparator at all times, whether or not the comparator is enabled. For normal operation of the comparator, the toggle switch should always be centered. Moving the toggle switch up will provide a logic one output from the comparator; returning the switch to center position will not change the output value. If the switch is left up, however, the comparator output will remain at logic one regardless of input changes. Likewise, moving the switch down will make the comparator output a logic zero, and the output will remain a logic zero as long as the switch is in the down position; moving the switch back to center will not change the output, but will free the comparator output to respond to input changes.

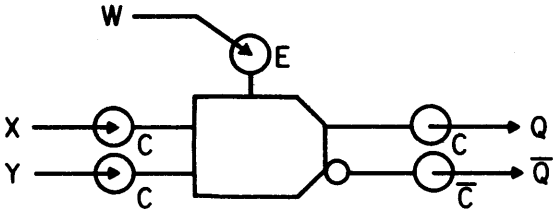
Four additional comparators may be added to each AD/FIVE field in place of the Track/Store Networks. These comparators behave in the same way as the standard complement of comparators, except that they are always enabled and do not have indicators or switches on the control panel.

If both analog inputs are zero at the beginning of a problem, the logic output value Q is indeterminate. To provide a determined (or even a false) value at t = 0 the following procedure is employed:

1. Arrange to have W = 0 prior to t = 0
2. Move the selector switch up for a value of logic one or down for a value of logic zero
3. Move the selector switch to the center position; the value selected will not change
4. Arrange to have W = 1 at t = 0; Q will immediately depend upon X and Y







SWITCH IS UP Q=1
SWITCH IS DOWN Q=0

iF SWITCH IS CENTERED AND W = 1

$(X + Y) > 0$ $Q = 1$
$(X + Y) < 0$ $Q = 0$

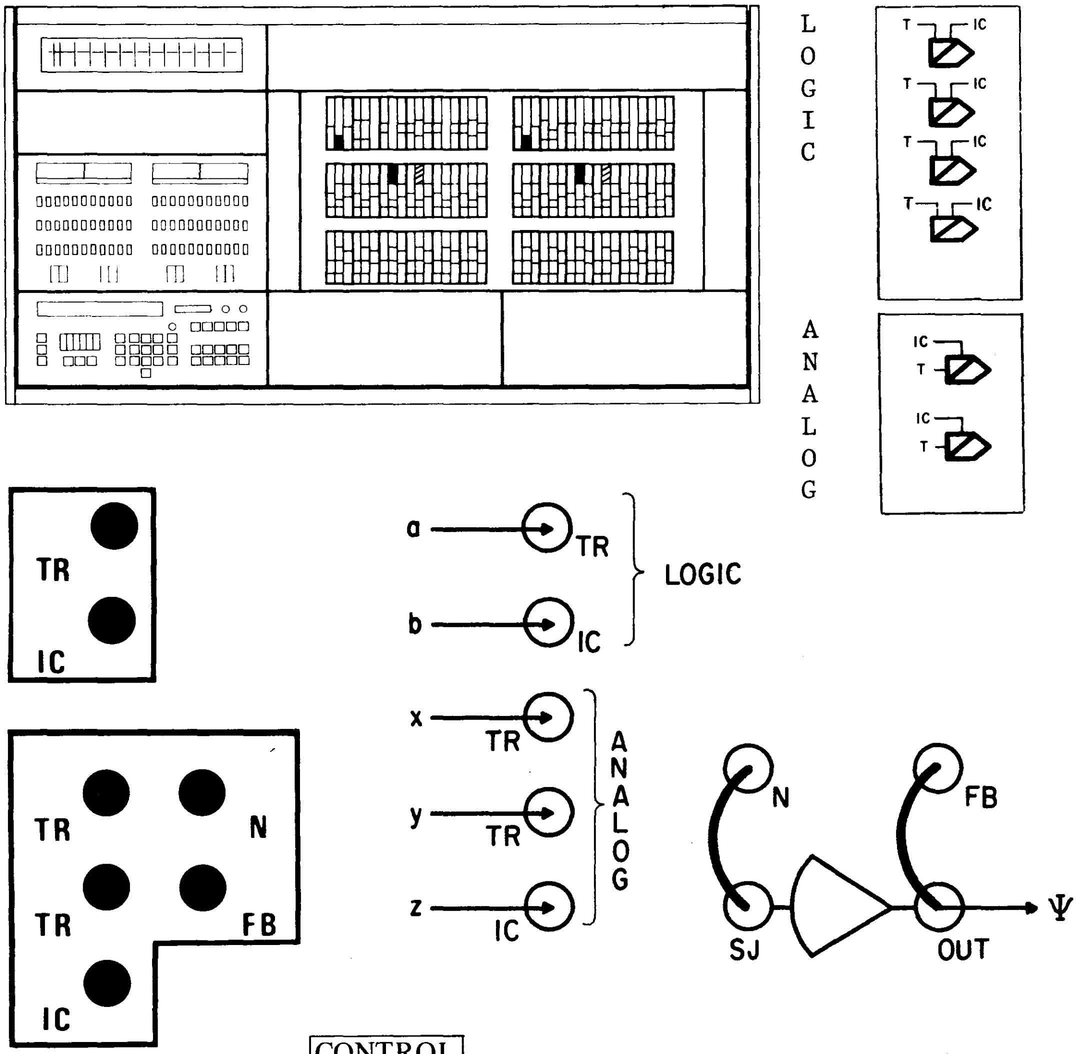
Output Q freezes if W=0

Patching a Comparator

Fiqure 4-16

4.2.1.14 Track/Store Network

Each AD/FIVE field may contain four Track/Store Networks. These networks are "attached" to a high gain operational amplifier to provide the track-store capability. Initial condition inputs and controls are provided.



CONTROL		
a	b	VALUE OF Ψ
0	0	"Frozen" (store last value)
0	1	-z ("initial" initial condition)
1	0	-y -x (track)
1	1	"Frozen" (store last value)

Patching a Track/Store Network

Figure 4-17

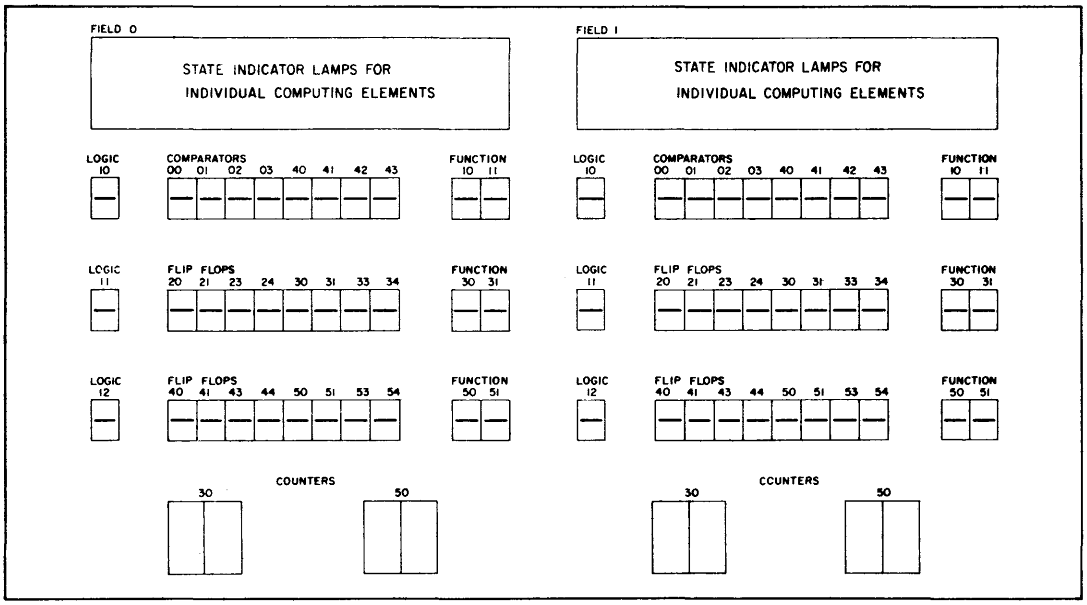
4.2.2 Logic Computing Elements

This section describes the patching of the Logic Computing Elements. All of these elements are to be found in the upper third of the patchboard. These logic elements have inputs and outputs which can exist at only one of two possible states, logic one or logic zero. Logic one and logic zero are implemented on the computer by two voltages (nominally, logic one: zero volts; logic zero: +5 volts). When operating the AD/FIVE, the important properties of logic one and logic zero are the following: logic one at an input causes the labelled action of that input, e.g., logic one at the input of an OR gate causes the output of the gate to be logic one. Or, as a further example, logic one at the input H of an integrator control causes that integrator to assume the HOLD mode. Outputs are patched to inputs.

Outputs are a logic one or a logic zero. Most outputs have an indicator lamp. If the lamp is lit, the state of the device is logic one. Most outputs have built-in logic inverters (complementors) with the complement output terminated at $\overline{\text{OUT}}$. Hence the computer equipment complement excludes free logic inverters. For the few cases where complementation is required, a NOR gate ($\overline{\text{OUT}}$ termination of an OR gate) is used. In general, it is desirable to have readily available the complement of any give signal, as the complement can readily be AND'ed with some other signal as shown in Figure 4-20.

Inadvertent patching of analog reference to any logic device will cause a protect circuit to temporarily activate. No physical harm will be done. Recovery is automatic upon removing the cause.

To operate and monitor certain logic devices a control panel is located immediately above the main control panel. Figure 4-18 depicts this panel. Descriptions of the individual indicator lamps and switches will be found in the appropriate sections which describe the individual computing elements.

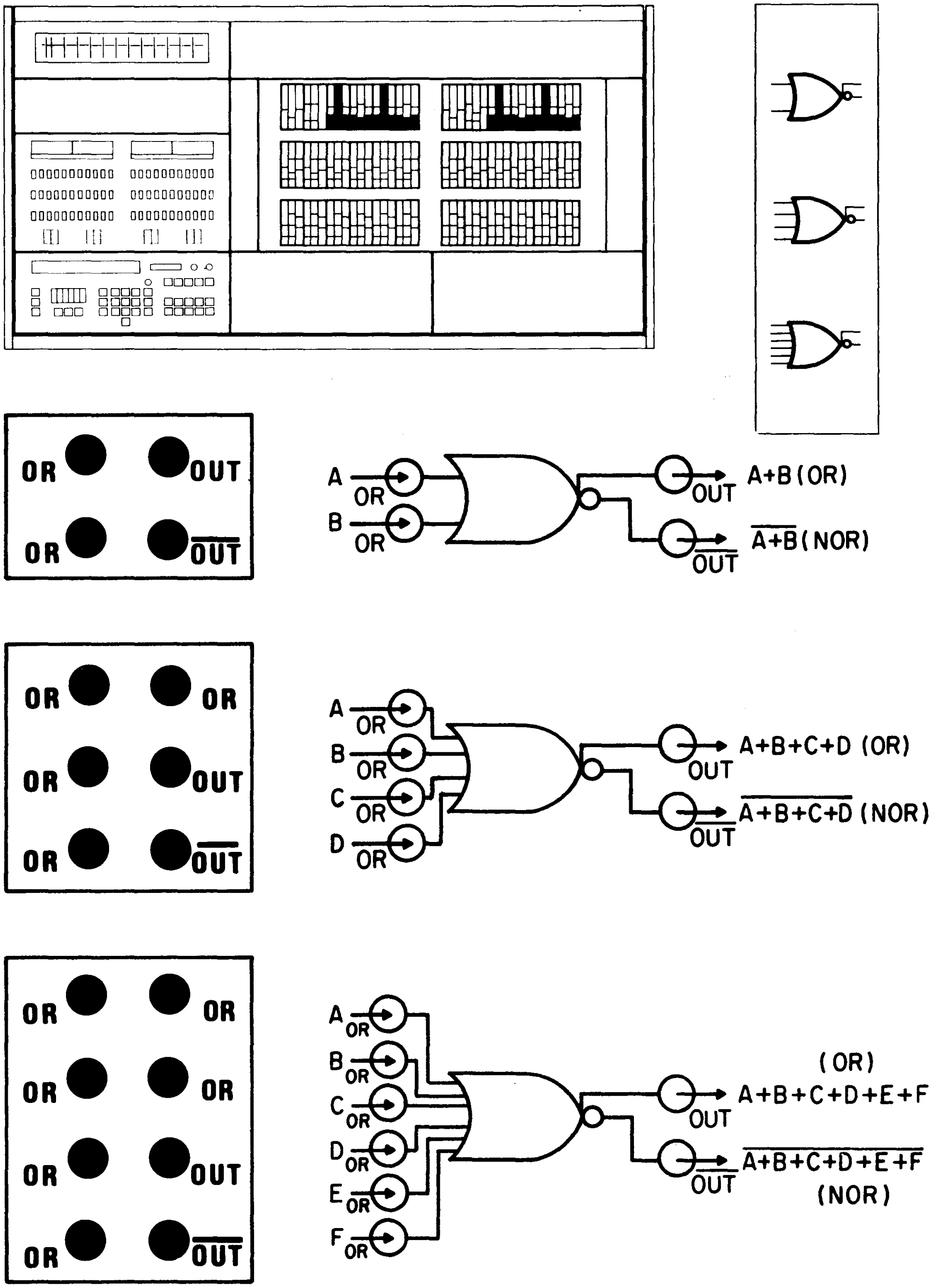

FIELD 0
STATE INDICATOR LAMPS FOR INDIVIDUAL COMPUTING ELEMENTS
LOGIC 10
COMPARATORS 00 01 02 03 40 41 42 43
FUNCTION 10 11
LOGIC 11
FLIP FLOPS 20 21 23 24 30 31 33 34
FUNCTION 30 31
LOGIC 12
FLIP FLOPS 40 41 43 44 50 51 53 54
FUNCTION 50 51
COUNTERS
30
50
FIELD I
STATE INDICATOR LAMPS FOR INDIVIDUAL COMPUTING ELEMENTS
LOGIC 10
COMPARATORS 00 01 02 03 40 41 42 43
FUNCTION 10 11
LOGIC 11
FLIP FLOPS 20 21 23 24 30 31 33 34
FUNCTION 30 31
LOGIC 12
FLIP FLOPS 40 41 43 44 50 51 53 54
FUNCTION 50 51
CCUNTERS
30
50

Logic Control Panel

Figure 4-18

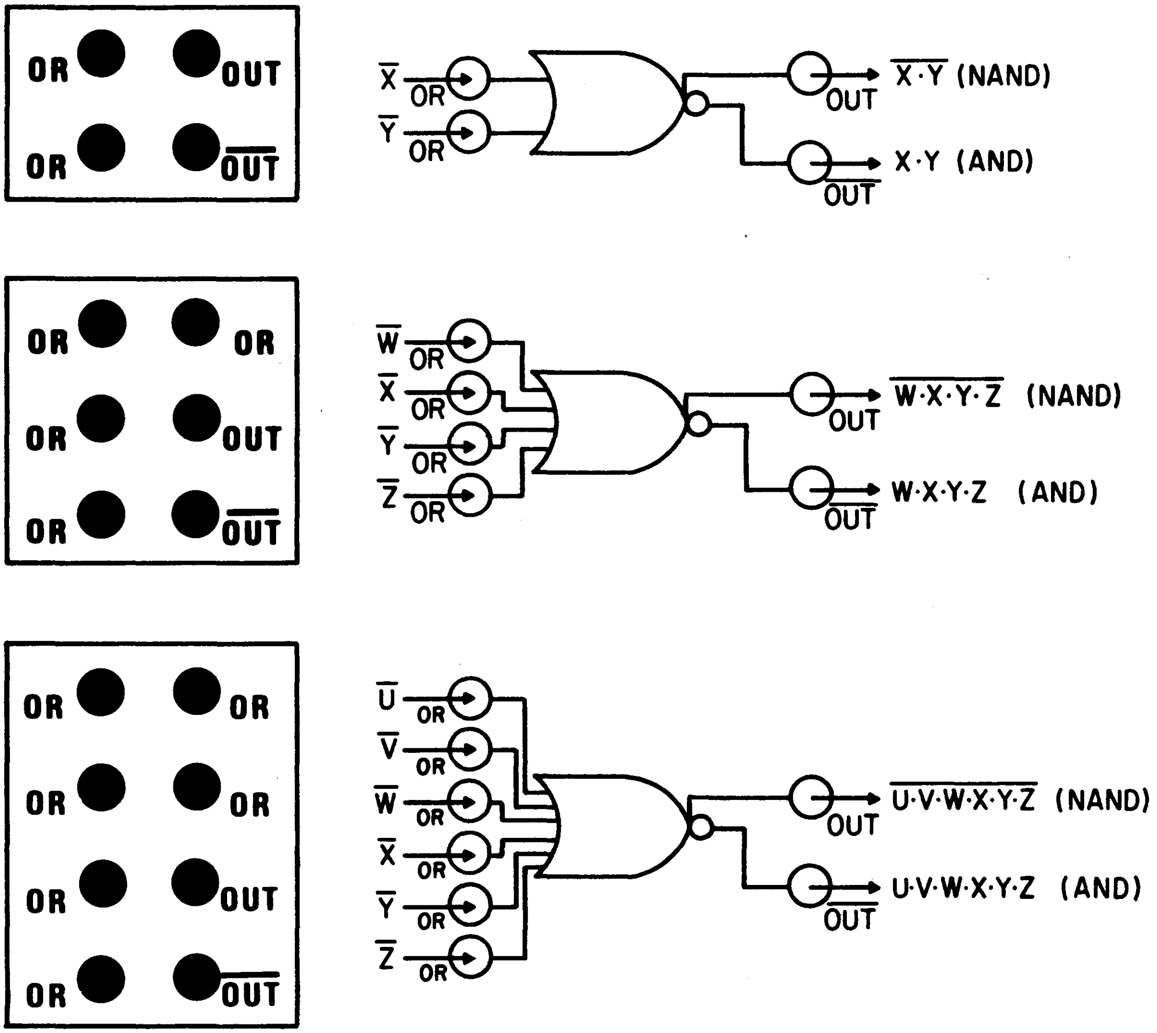
4.2.2.1 Gates

AD/FIVE gates are terminated as OR/NOR gates. Two, four, and six-input variations are available. According to DeMoraan's Theorem AND/NAND functions may be performed by OR/NOR gates. This is shown is Fiqure 4-20.



Patching Gates for the OR/NOR Function

Figure 4-19



Note: It is not necessary to patch all OR inputs to a gate. Only the number of signals needed is patched to gate inputs. To obtain the AND of three values, for example, a four or six input gate may be used, and only the three input signals are patched. The remaining input(s) are ignored.

Patching gates for the AND/NAND Function

Figure 4-20

4.2.2.2 Dual Flip-Flop

The AD/FIVE flip-flops are designed to be used independently or as individual cells in binary up or down counters. Each flip-flop pair has common ENABLE and LOAD inputs, but has individual TRIGGER, SET, CLEAR, LOAD SET, and OUT and O̅U̅T̅ terminations. Except for receiving common enable and load signals, the flip-flops in a pair are completely independent and any desired interaction between them (such as shiftings or counting) is programmed by the operator. Another pair of terminals, CO and $\overline{\text{CO}}$ (Carry Out and its complement), is provided to simplify up and down counter programming. Chapter 5 presents several examples of counter, shift register, and individual flip-flop programming.

In addition to its logic patchboard terminals, each flip-flop has an indicator lamp and a single-pole triple throw spring return toggle switch on the logic switch and indicator panel. The following paragraphs give a description of flip-flop operation in the three logic modes. All transitions are synchronous.

LOAD A flip-flop may be in LOAD because the computer logic mode is LOAD, or it may be commanded to LOAD individually by having a logic one on its LD input. In either case, the flip-flop output will be determined by either the toggle switch or the LDS input on the logic patchboard. When it is patched from some other logic device, the LDS input overrides the toggle switch in determining the flip-flop load state output.

RUN A flip-flop will be in the RUN mode whenever the logic mode is RUN unless a logic one is patched into its LD input. In RUN, flip-flops respond to changes in the logic levels on their inputs (trigger, set, and clear) at a rate determined by the basic computer clock and the signal patched into the flip-flop enable input. Changes in output state may occur only when there is coincidence between the basic computer clock and the enable input. These coincidences are referred to as enabled clock periods.

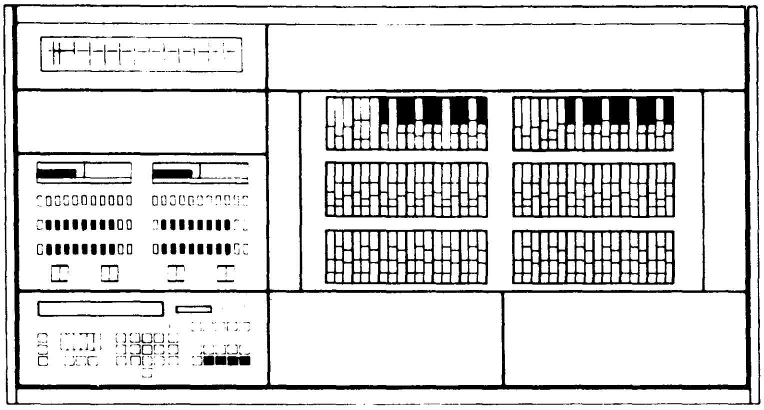
SET: When SET is logic one and CLR and T are zero, OUT will be logic one at the end of the next enabled clock period.

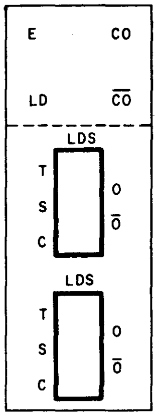
CLR: When CLR is logic one and SET and T are zero, OUT will be logic zero at the end of the next enabled clock period.

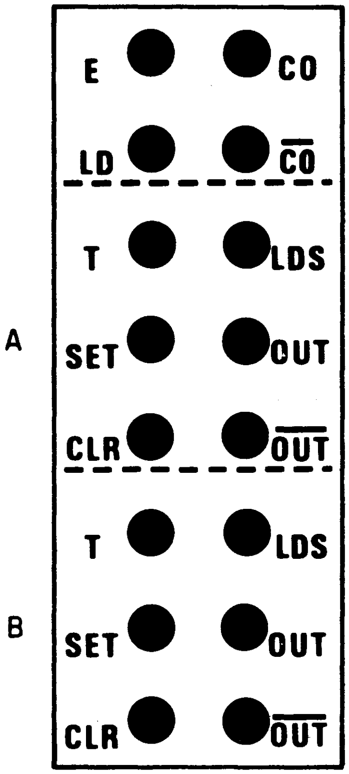
T: When T is logic one, OUT will change its state at the end of each enabled clock period. A logic one on both SET and CLR has exactly the same effect as a logic one on T.

STOP A flip-flop will be in the STOP mode whenever the logic mode is STOP unless a logic one is patched into its LD input. In STOP, the flip-flops store the last state they had before the transition to STOP mode.

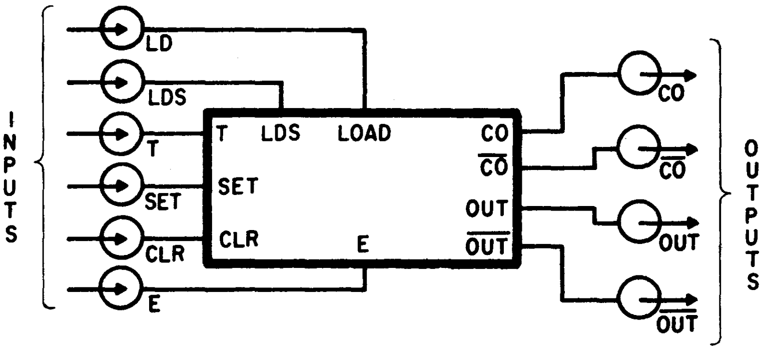
There are three conditions, however, that will allow a flip-flop to change state during the STOP mode. Logic Step allows the advance of the logic program for one clock period upon pushing the LOGIC STEP button while in the STOP mode. Advance Step allows the logic program to go to RUN for a predetermined number of clock periods when the AST terminal is patched to a logic one. If any enabled clock periods occur during use of either of these two features, the flip-flop may change states. The third condition occurs when the flip-flop toggle switch is depressed while the mode is STOP. This complements the output for the individual flip-flop. The switch is spring returned to center from the down position, and the original state of the flip-flop may be resumed by depressing the switch a second time while the mode is STOP.







Two FLIP-FLOPS with a common ENABLE and LOAD (MODE CONTROL) input



CO = logic one if OUT_A = logic one

and OUT_B = logic one

and T_A = logic one

	SET		CLEAR		TRIGGER			
SET	1	1	0	0	1	1	1 or 0	1 or 0
CLR	0	0	1	1	1	1	1 or 0	1 or 0
T	0	0	0	0	0	0	1	1
OUT	1	0	1	0	1	0	1	0
$\overline{OUT}$	0	1	0	1	0	1	0	1

Before enabled clock pulse

OUT	1	1	0	0	0	1	0	1
$\overline{OUT}$	0	0	1	1	1	0	1	0

After enabled clock pulse

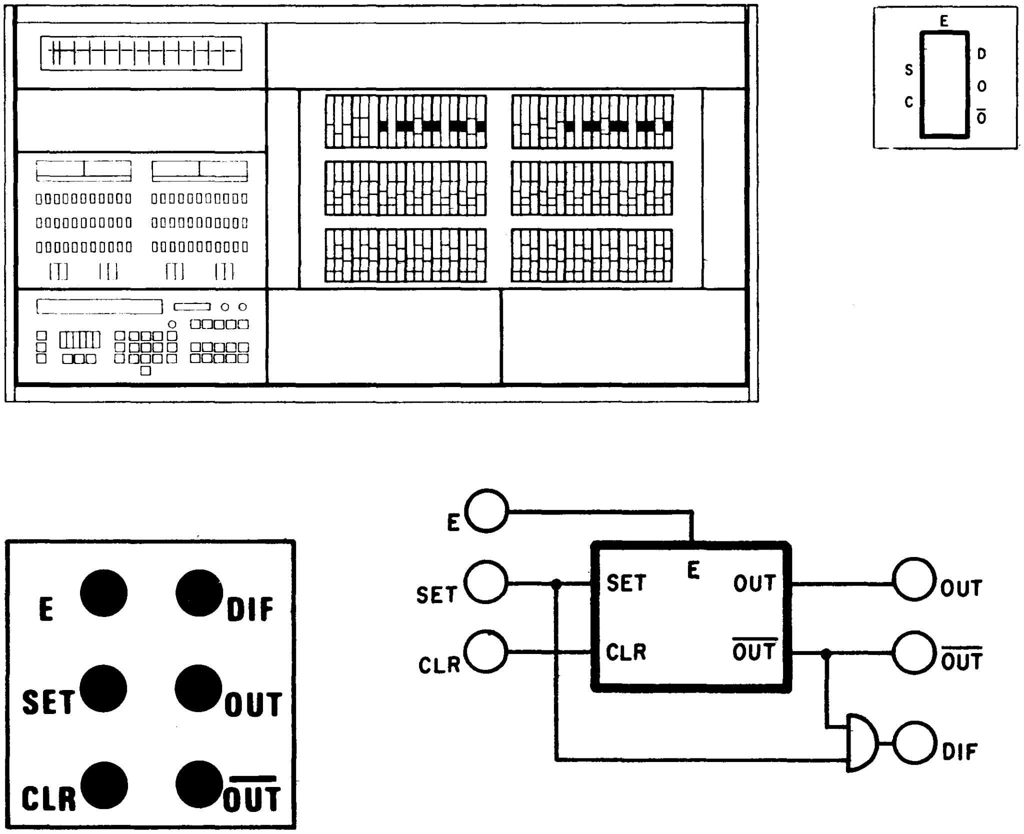
Patching a Dual Flip-Flop

Figure 4-21

4.2.2.3 Flip-Flop/Differentiator

Immediately below each dual flip-flop is a flip-flop differentiator. The flip-flop is a normal synchronous flip-flop with ENABLE, SET, CLEAR inputs and OUT and $\overline{\text{OUT}}$ outputs. There is no switch on the logic control panel. An extra gate is built in to provide convenient patching of a differentiator circuit. The gate output is the DIF termination.

A careful study of the truth table in Figure 4-22 reveals that the DIF output is a signal which predicts the logic zero to one transition of the output of the associated flip-flop. The DIF output signal is of programmable duration, and the length of the DIF signal and its frequency of occurrence may both be controlled by the operator. If, for example, a logic one reference is patched into both SET and CLR, and E is a V1S signal, then the duration of the DIF signal is one second, and the DIF signal is present for one second, absent for the next second, then present for the next second, etc.; if E is a V100M signal, then the duration of DIF is .1 second, and the DIF signal is present for .1 second, absent for .1 second, present for .1 second, etc. The DIF signal lasts as long as SET is a logic one and OUT is a logic zero. Thus, disabling the flip-flop differentiator when the flip-flop is set, but before it changes state, will cause the DIF output to be a logic level one. See Chapter 5, Sections 5.2.3, 5.2.3.1, and 5.2.3.2 for details on applications of this useful computing component.



Patching a Flip-Flop Differentiator

Figure 4-22

	MEMORY		SET		CLEAR		TRIGGER	
SET	0	0	1	1	0	0	1	1
CLR	0	0	0	0	1	1	1	1
OUT	1	0	1	0	1	0	1	0
$\overline{\text{OUT}}$	0	1	0	1	0	1	0	1

before enabled
clock pulse

OUT	1	0	1	1	0	0	0	1
$\overline{\text{OUT}}$	0	1	0	0	1	1	1	0

after enabled
clock pulse

SET	0	1	0	1
OUT	0	0	1	1
DIF	0	1	0	0

DIF is a function of SET and OUT as shown

Patching a Flip-Flop Differentiator

Figure 4-22

4.2.2.4 Two Decade Variable Carry-Out BCD Counter

A fully expanded AD/FIVE contains four Variable Carry-Out BCD Counters. These counters normally count from 0 to 99 (100 counts) repetitively, and provide an additional carry-out (CO VAR) at an intermediate count value determined by the setting of a two decade thumbwheel switch. This carry-out occurs (yields a logic one) when the following conditions are true:

A. The count is equal to one less than the value set on the corresponding thumbwheel.
B. The carry-in (CI) is a logic one.

Another terminal, CO 100, provides a carry-out when: the count is 99 and CI is a logic one.

A powerful adjunct to the thumbwheels used to set the variable carry-out is a set of patchboard inputs which allow this variable carry-out to be set by patched logic values. To achieve this control a logic one signal is patched to the inhibit switch (ISW) input hole. The thumbwheel is now ignored and patched BCD inputs determine the variable carry-out value. Only valid BCD values will work. Valid and non-valid BCD values are tabulated below.

VALID BCD VALUES

DECIMAL VALUE	BCD VALUE
0	0
1	1
2	2
3	2 1
4	4
5	4 1
6	4 2
7	4 2 1
8	8
9	8 1
10	10
20	20
30	20 10
40	40
50	40 10
60	40 20
70	40 20 10
80	80
90	80 10

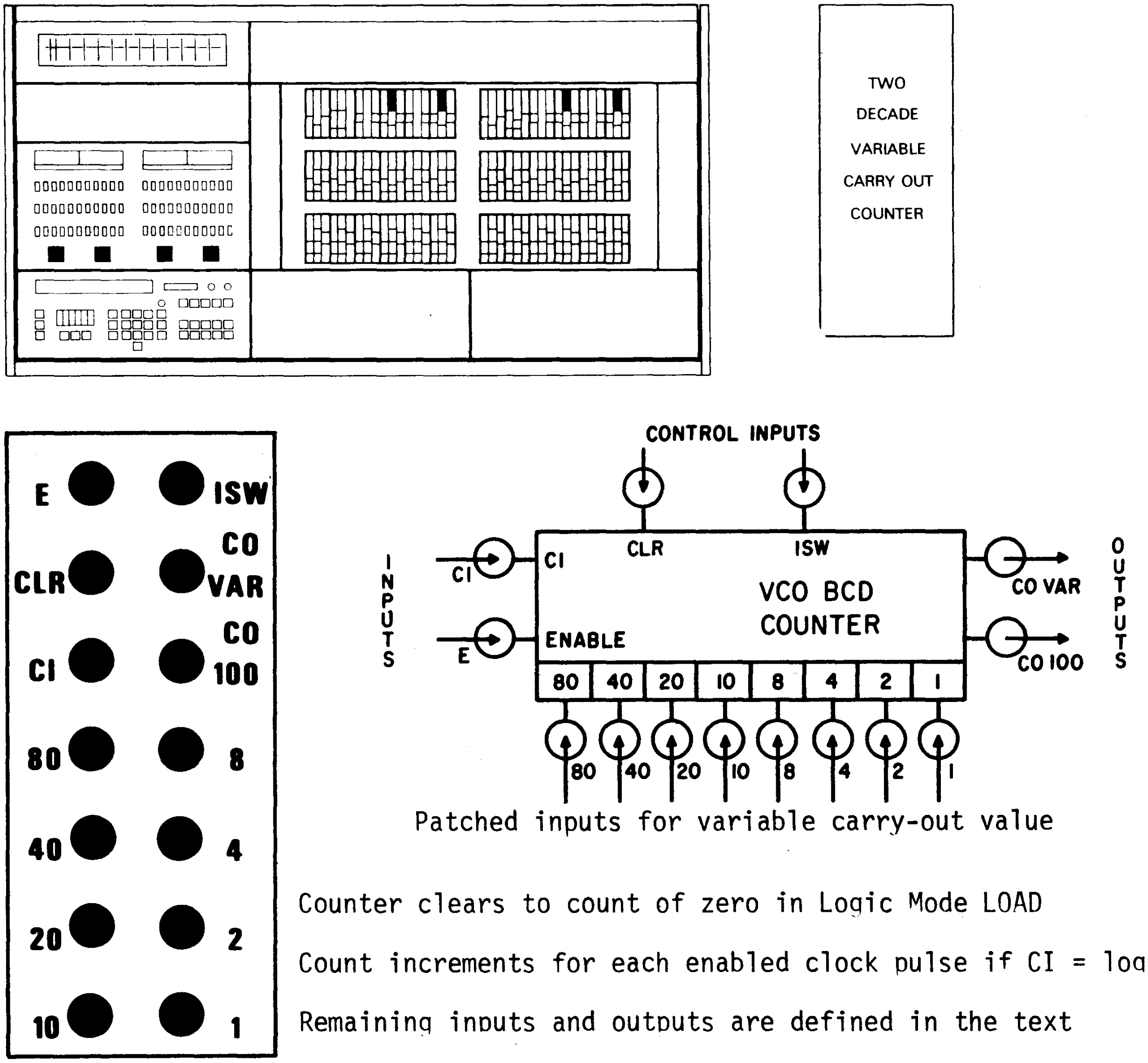
EXAMPLES

73	40 20 10 2 1
49	40 8 1
88	80 8
11	10 1

NON-VALID VALUES

Any combination of:	80	and	40
	80	and	20
	8	and	4
	8	and	2

Indicator lamps are provided on the Loaic Control Panel to monitor the count in the counter. The lamps are BCD coded (80, 40, 20, 10, 8, 4, 2, 1). The clear (CLR) input is synchronous, i.e., it clears the count in the counter to zero upon receiving the next clock pulse, but the pulse does not need to be enabled, i.e., the CLR input will clear the counter whether or not the E input is enabled.



Patched inputs for variable carry-out value

Counter clears to count of zero in Logic Mode LOAD

Count increments for each enabled clock pulse if CI = loaic one

Remaining inputs and outputs are defined in the text

Patching a Variable Carry-Out BCD Counter

Fiqure 4-23

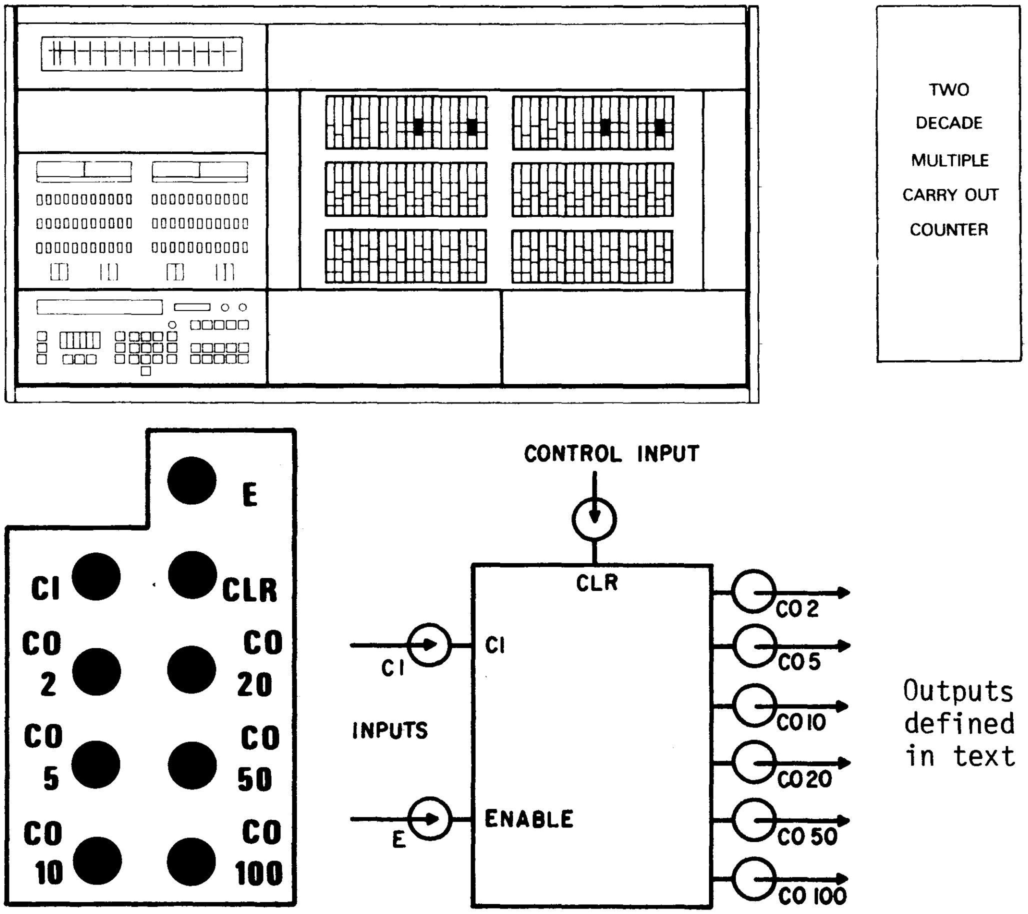
4.2.2.5 Multiple Carry-Out BCD Counter

A fully expanded AD/FIVE contains four Multiple Carry-Out BCD Counters. These counters normally count from 0 to 99 (100 counts) repetitively and provide a number of carry-outs from several terminations at various intermediate count values. These carry-outs are tabulated below. Carry-out are a logic one only if the carry-in (CI) is a logic one.

Output Terminal	Value of the count in the counter which will provide a logic one at the output.									
CO 2	1,	3,	5,	7,	9,	11,	----	95,	97,	99
CO 5	4,	9,	24,	19,	14,	29,	----	89,	94,	99
CO 10	9,	19,	29,	39,	49,	59,	69,	79,	89,	99
CO 20		19		39		59		79		99
CO 50					49					99
CO 100										99

Other terminations (E, CI, CLR) function in the same way as the similar terminations of the VCO BCD Counter. CLR does not need to be enabled.

There are no indicator lamps. The fact that the counter is built internally in BCD format is presented for information only, since there are no indicator lamps or BCD coded terminations.



Outputs defined in text

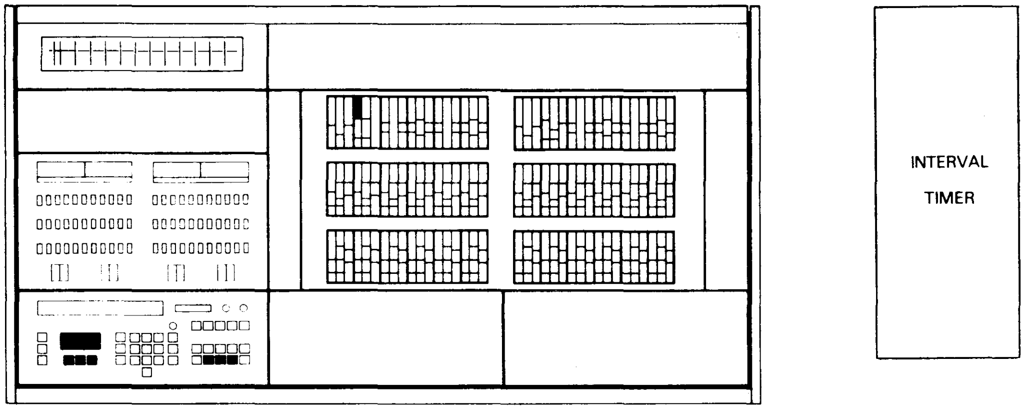
Count clears to zero in Logic Mode LOAD
Clear input is synchronous

Patching a Multiple Carry-Out BCD Counter

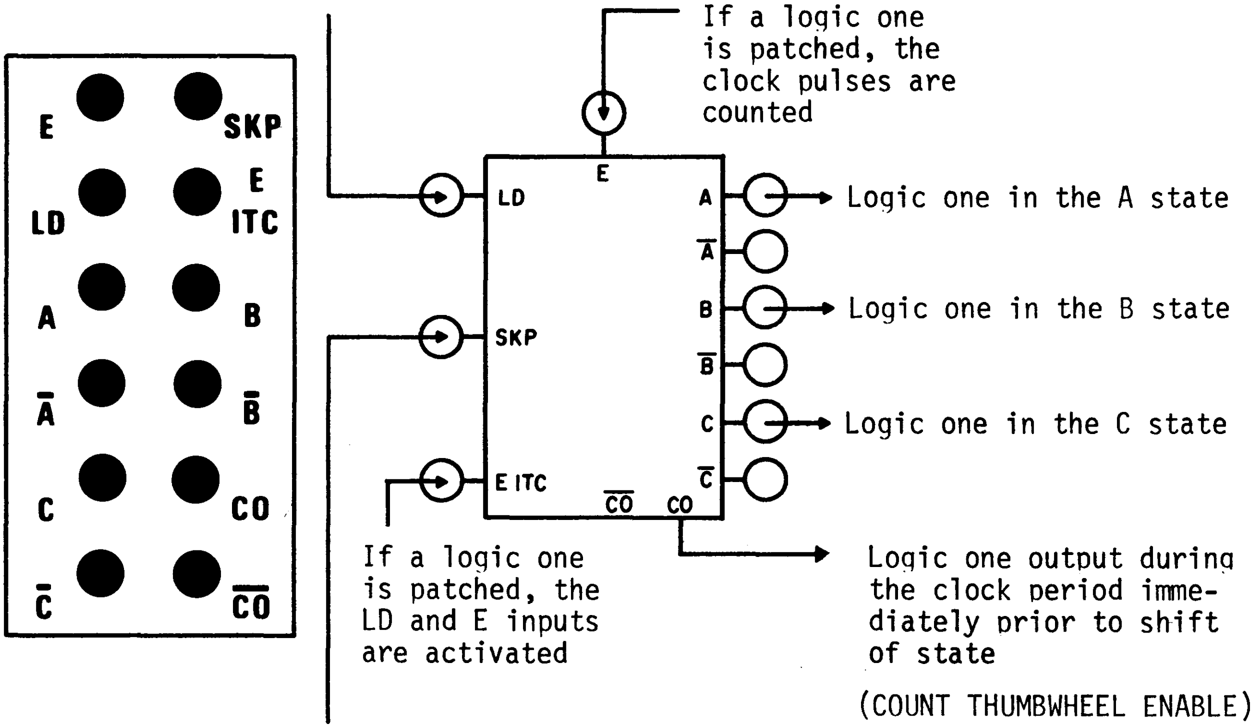
Figure 4-24

4.2.2.6 Interval Timer

The operation of the Interval Timer under pushbutton control is described in Chapter 3. There is a set of patchboard holes which allow the Interval Timer to be used as a logic element. When used as such, it is a three-bit ring-counter with thumbwheel settable variable counters determining the counts which cause the ring-counter to shift states. Such logic patchboard use does not interfere with the previously discussed use of the IT as an analog mode controller under Logic Exec control. Patching of the Interval Timer is shown in Figure 4-25.



LD is an asynchronous input. Patching a logic one into LD causes the IT to immediately return to the "A" state at a count of zero.



If a logic one is patched, the clock pulses are counted

Logic one in the A state

Logic one in the B state

Logic one in the C state

If a logic one is patched, the LD and E inputs are activated

Logic one output during the clock period immediately prior to shift of state

(COUNT THUMBWHEEL ENABLE)

The SKP input is always active, whether or not E is a logic one. A logic one patched into SKP cancels the present state of the interval timer and advances the timer to the next state at the next enabled clock pulse. If the count has been reached and the state is ready to change when SKP goes to a logic one, then the entire next state is skipped at the next enabled clock pulse.

Patching the Interval Timer

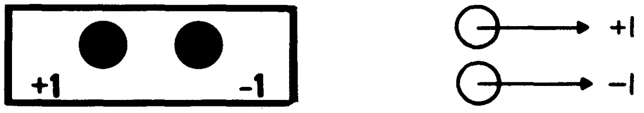
Figure 4-25

4.2.3 Miscellaneous Computing Elements

This section describes miscellaneous computing elements used in analog usage of the AD/FIVE. Holes which are devoted exclusively to hybrid applications are described in the AD/FIVE hybrid manual.

4.2.3.1 Analog Reference

The analog reference is terminated in holes located throughout the lower two-thirds of the patchboard and labelled +1 and -1 as shown in Figure 4-26

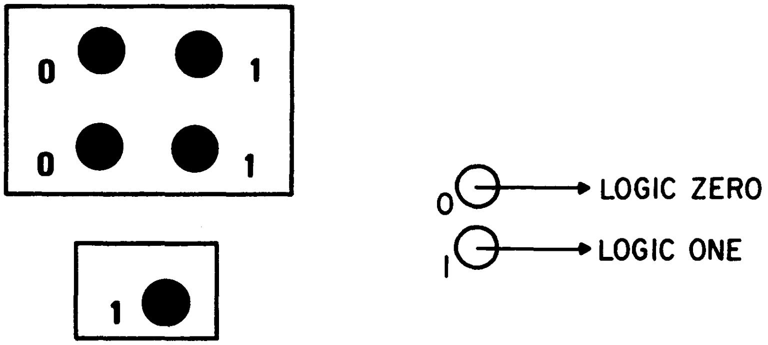


Patching Analog Reference

Figure 4-26

4.2.3.2 Logic Reference

The logic reference is terminated in holes located in the upper third of the patchboard and labelled 0 and 1 as shown in Figure 4-27.

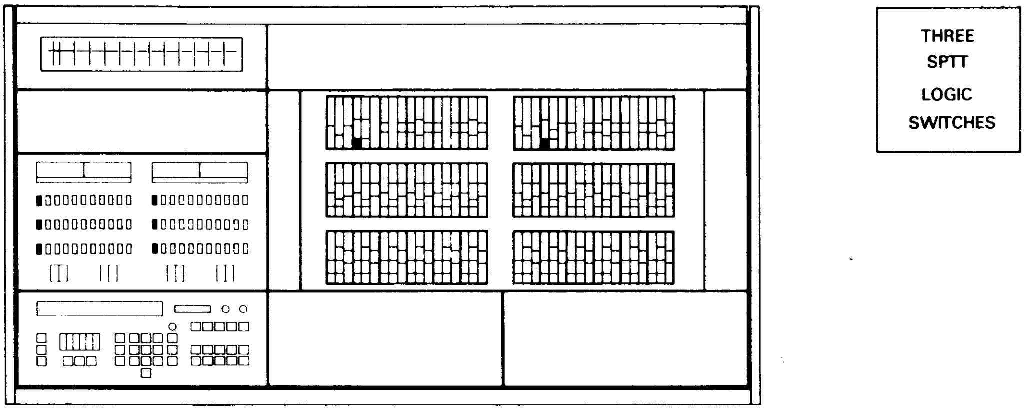


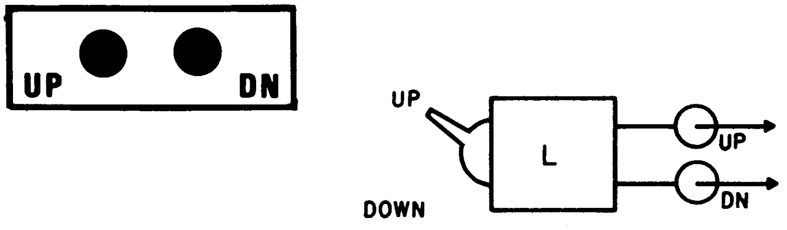
Patching the Logic Reference

Figure 4-27

4.2.3.3 Switches

The AD/FIVE Logic Panel contains two toggle switches not yet discussed. These are the Logic and Function switches. See Figures 4-28 and 4-29 for patching of these switches. Terminations for the switches are obtained as the computer is expanded. Logic outputs of the switches are suitably buffered to prevent multiple triggering of logic devices due to mechanical bounce of the switch contact.

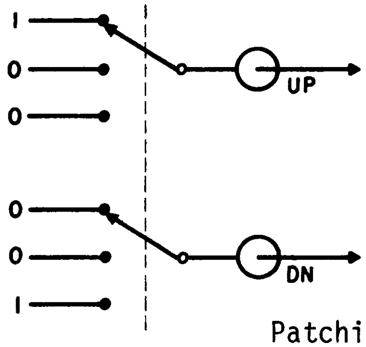




logic one if the switch position is "UP"

logic one if the switch position is "DOWN"

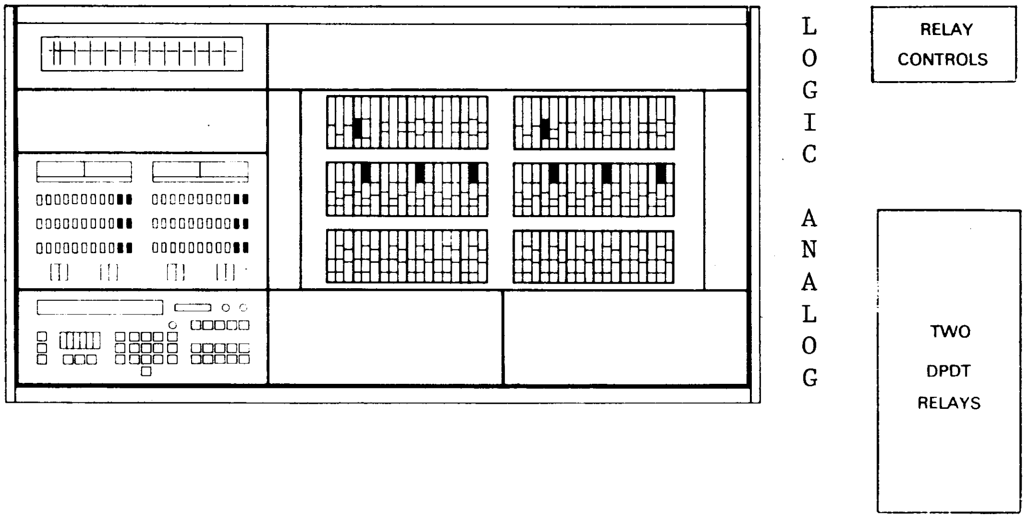
Both outputs are logic zero, if the switch is in the center.

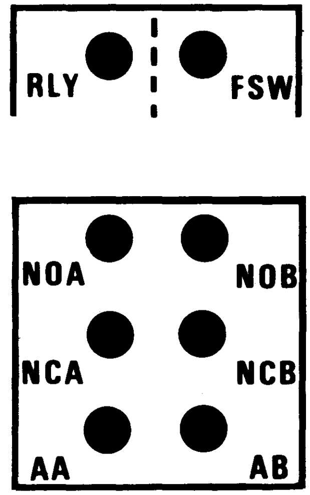


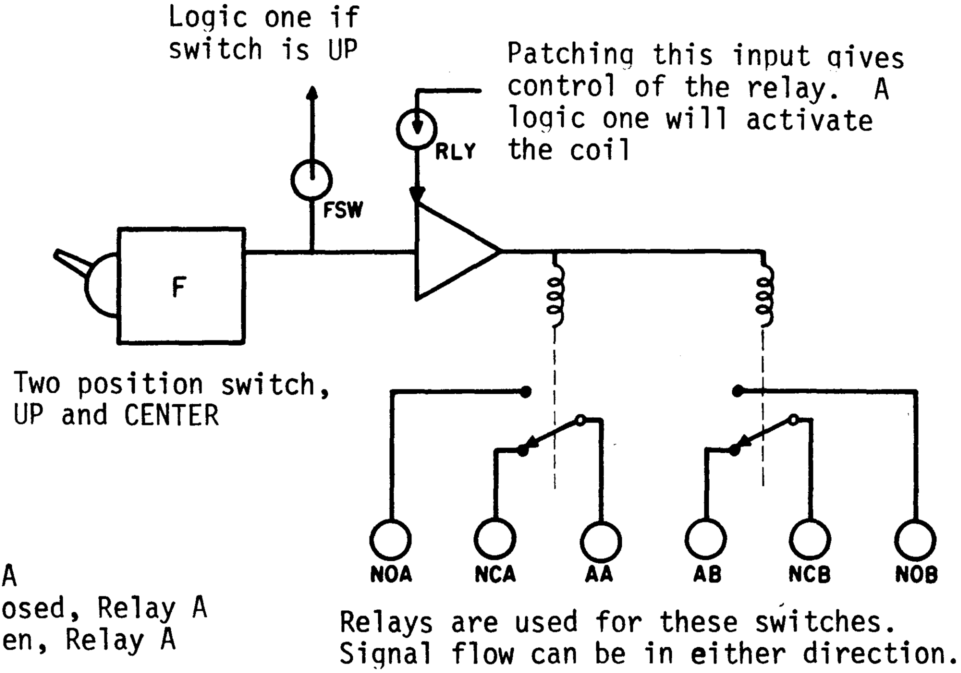
Equivalent Electrical Diagram

Patching a Logic Switch

Figure 4-28







Logic one if
switch is UP

Patching this input gives
control of the relay. A
logic one will activate
the coil

Two position switch,
UP and CENTER

Relays are used for these switches.
Signal flow can be in either direction.

AA	ARM, Relay A
NCA	Normally closed, Relay A
NOA	Normally open, Relay A
AB	ARM, Relay B
NCB	Normally closed, Relay B
NOB	Normally open, Relay B

Patching a Function Switch

Figure 4-29

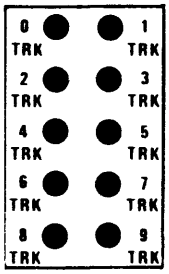
4.2.3.4 Trunks

Trunks provide patchboard terminations from external sources connected to the wiring plane, and are used for slaving, control of external devices, etc. Each analog area except 00 can have ten trunks. Trunks 0-4 are addressable.

Logic trunks are not addressable. Logic trunks may be located as follows:

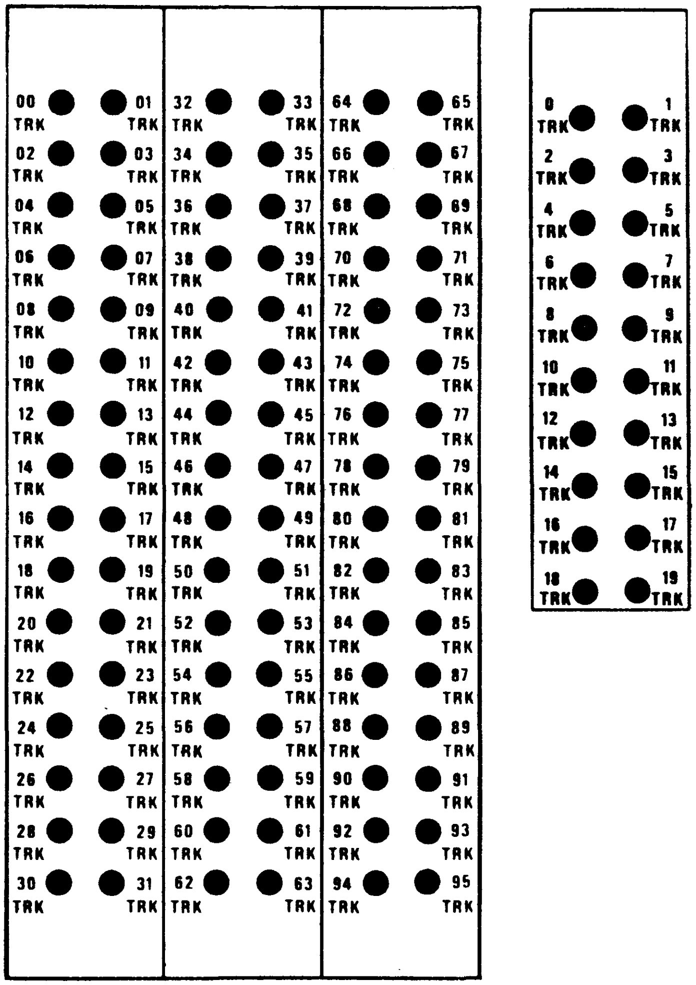
Twenty trunks in area 11
Groups of up to ninety-six trunks in field 1

Analog Trunk Terminations



CAUTION: Addressable trunks 0 to 4 exhibit a ground connection when the AD/FIVE power switch is turned off.

Logic Trunk Terminations

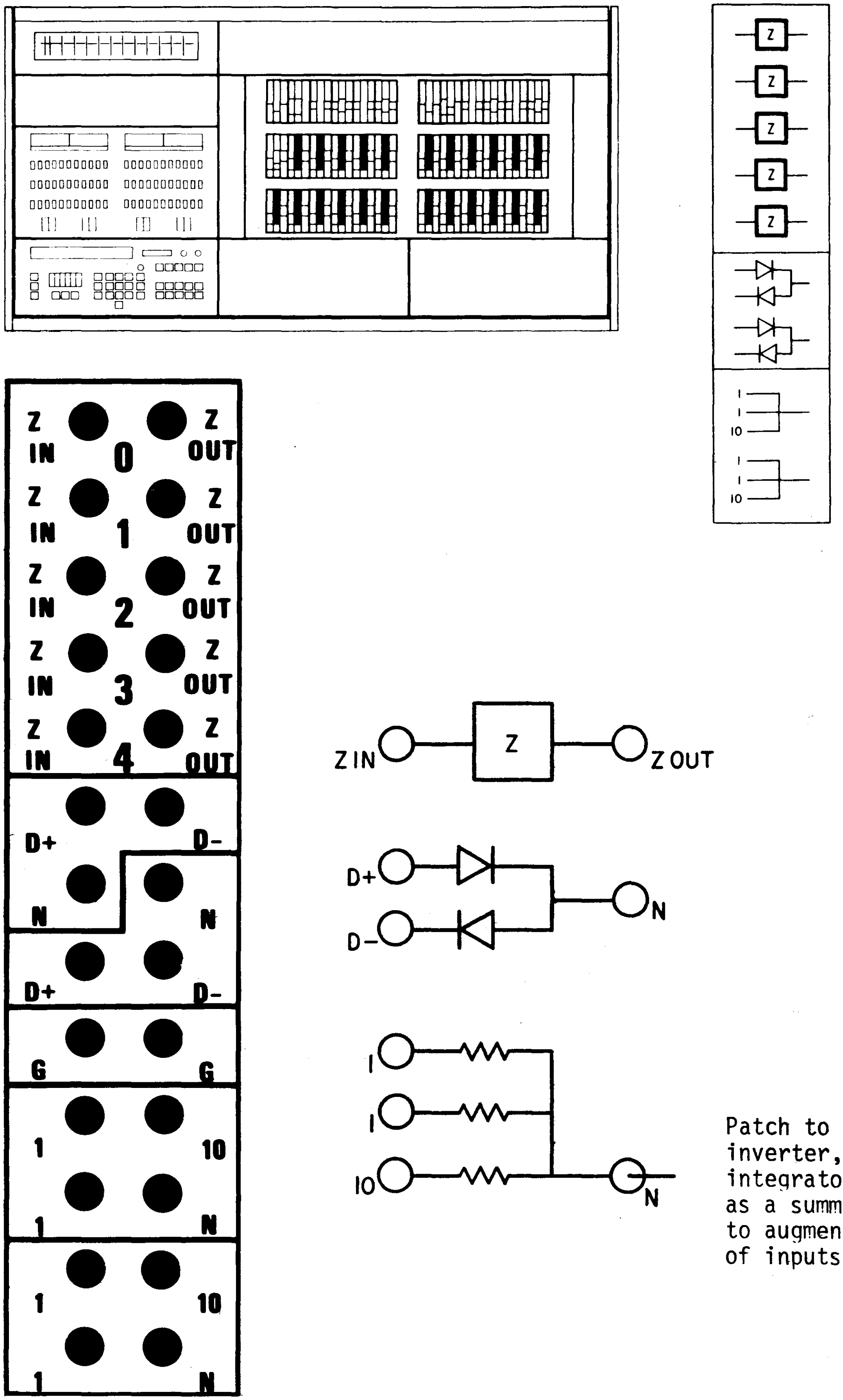


Trunks

Figure 4-30

4.2.3.5 Free Impedance

Special Free Impedance cards may be installed in the center slot in any analog area except Field 0, Upper Analog Area 0. These cards contain free resistors, free diodes, and special "do it yourself" impedance networks. The user may install whatever networks he wishes on the upper portion of this card.



Patch to SJ of summer, inverter, or summer/ integrator configured as a summer, in order to augment the number of inputs

Patching the Free Impedance Network

Figure 4-31

4.2.4 Control and Miscellaneous Patchboard Terminations

The following section of this chapter gives a resume of control and miscellaneous terminations and references to previous discussions of these terminations as they are given in the Reference Manual. Chapter 4 thus gives a complete descriptive catalog of all patchboard terminations used in analog applications. Patchboard terminations for hybrid usage are discussed separately in the section of the Reference Manual on hybrid usage of the AD/FIVE.

4.2.4.1 Analog Mode Control

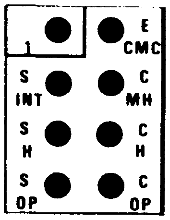


Figure 4-32

Analog Mode Control: System Outputs and Enabled Console Inputs. Chapter 3, Sections 3.1.5, 3.2.1, and 3.2.4.2.

S INT: System Interrupt
S H: System Hold
S OP: System Operate
E CMC: Enable Console Mode Control
C MH: Console Master Hold
C H: Console Hold
C OP: Console Operate

Located in field 0, logic area 1.

4.2.4.2 Logic Mode Control

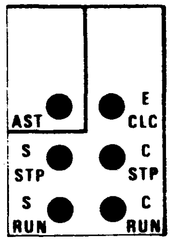


Figure 4-33

Logic Mode Control: System Outputs and Enabled Console Inputs.

AST: Advance Step Terminal. Chapter 3, Section 3.1.2.5.
S STP: System Stop
S RUN: System Run
Chapter 3, Sections 3.1.5 and 3.2.2.
E CLC: Enable Console Logic Control
C STP: Console Stop
C RUN: Console Run
Chapter 3, Sections 3.2.2 and 3.2.4.2.

Located in field 0, logic area 1.

4.2.4.3 Console Mode Control Outputs

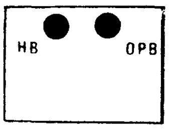


Figure 4-34

Console level mode control outputs from the Hold Bus and Operate Bus. These terminals provide a logic output for console level mode control, and reflect the console mode as controlled by the control panel mode control pushbuttons, the interval timer mode control, or the patched console mode control. HB provides a logic one output when the console mode is Hold; OPB provides a logic one output when the console mode is Operate.

Located in fields 0 and 1, logic area 0.

4.2.4.4 Overload Signals

Overload Logic Output Signal Terminations. Chapter 2, Section 2.2.4, and Chapter 3, Section 3.2.4.2.

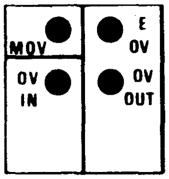


Figure 4-35

M OV: Master Overload. This output is useful in slaving, and provides logic signal output of any overload in the entire system when OV OUT of the slave console is enabled by a logic one in E OV and patched to OV IN of the master console. M OV on the master console then functions as a system overload logic output. On the slave console, M OV functions as a console overload logic output.
OV IN: Overload In
E OV: Enable Overload
OV OUT: Overload Out

Located in field 0, logic area 1.

4.2.4.5 Time Scale Control

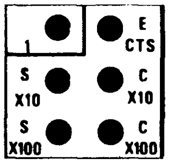


Figure 4-36

System Outputs for X10 and X100; Enabled Console Inputs for X10 and X100. Chapter 3, Sections 3.2.3 and 3.2.4.2.

Located in field 0, logic area 1.

4.2.4.6 Time Control Signals

P Signals: Reference Timer Outputs. Chapter 3, Section 3.1.3.2. A **pulse** is a 1μs duration signal.

P1S = 1 pulse every second
P100M = 1 pulse every 100 milliseconds
P10M = 1 pulse every 10 milliseconds
P1M = 1 pulse every millisecond
P100μ = 1 pulse every 100 microseconds
P10μ = 1 pulse every 10 microseconds

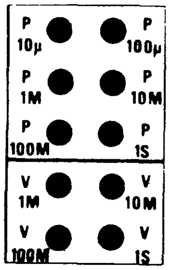


Figure 4-37

V Signals: Variable Time Signals. Chapter 3, Section 3.1.3.2.

V1S = 1 pulse every second when time scale is X1
V100M = 1 pulse every 100 milliseconds when time scale is X1
V10M = 1 pulse every 10 milliseconds when time scale is X1
V1M = 1 pulse every millisecond when time scale is X1

V Signals may be multiplied in decade steps by use of the X10 and X100 time scale pushbuttons, up to 1 pulse every microsecond from the V1M output when the time scale is X1000.

Located in field 0, logic area 1. (V signals may also be located in field 1, logic area 1.

4.2.4.7 Clock Selection and Output

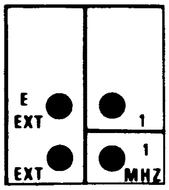


Figure 4-38

Chapter 3, Section 3.2.4.1.

E EXT: Enable External Clock Selection
EXT: External Clock
1MHZ: AD/FIVE Clock Output

Located in field 0, logic area 1.

4.2.4.8 System Lines

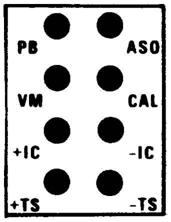


Figure 4-39

PB Patchboard. Chapter 3, Section 3.1.6.4
ASO Address Selector Output. Chapter 3, Section 2.2.7.5
VM Voltmeter. Chapter 2, Section 2.2.7.5
CAL VDFG Calibration. See Appendix on VDFG operation
+IC
-IC
+TS
-TS
Problem Verify Terminals. Chapter 3, Section 3.1.1.4

Located in field 0, upper analog area 0.

4.2.4.9 Slaving Terminations

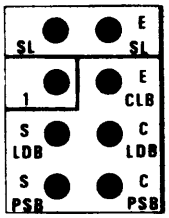


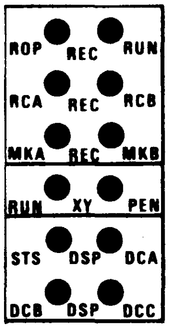
Figure 4-40

Chapter 3, Section 3.2.4.2.

SL: Slave Pushbutton Output
E SL: Enable Slave Pushbutton Output
S LDB: System Load Bus
S PSB: System Pulse Signal Bus
E CLB: Enable Clock and Load Bus Controls
C LDB: Console Load Bus
C PSB: Console Pulse Signal Bus

Located in field 0, logic area 1.

4.2.4.10 Recorder and Display Controls



See Chapter 3, Section 3.2.4.3.
For information on the use of these terminations with the standard F6.18 Display Option offerred with the AD/FIVE, see the Display Option Manual.

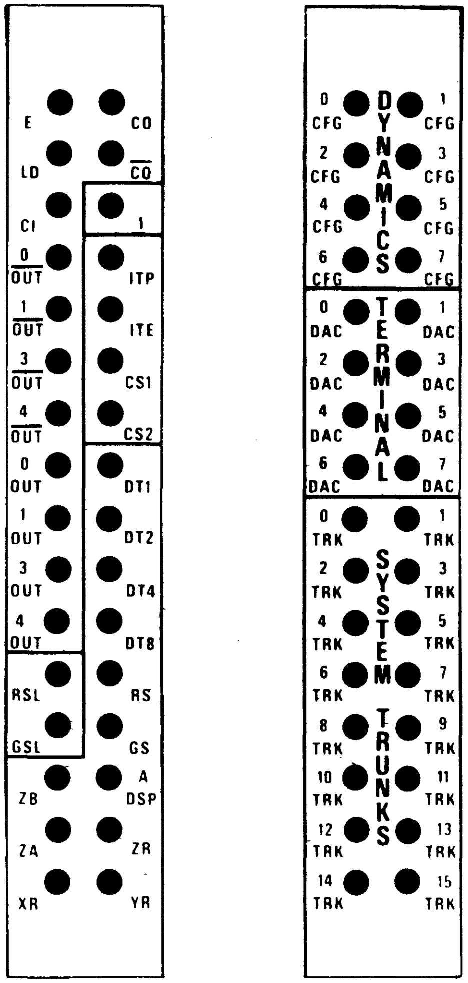
These terminations are located in field 0, logic area 1.

Figure 4-41

4.2.4.11 Dynamics Terminal Controls

For patching of these terminations, see the Dynamics Terminal Maintenance Manual.

The Dynamics Terminal Control terminations are located in field 0, logic area 5.

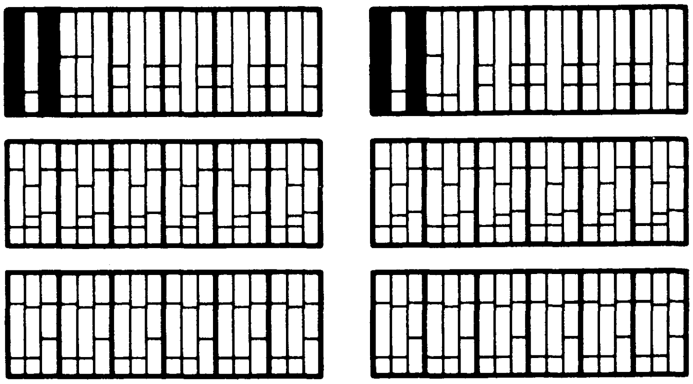


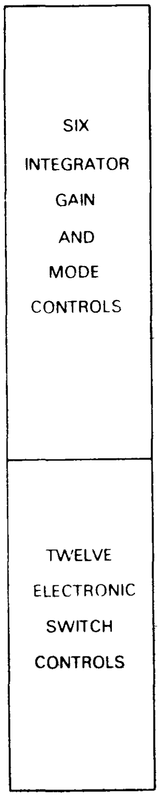
Fiqure 4-42

4.3 CARD CATALOG

This section presents a description of each of the cards which mount directly behind the patchboard. Valid mounting slots are indicated, standard locations in black shading and optional locations in cross-hatching.

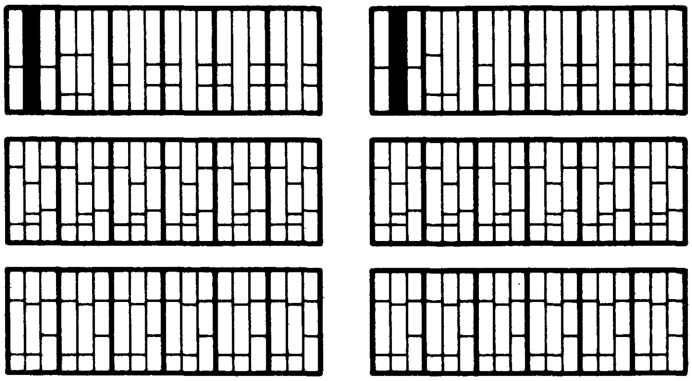
D5.140 Integrator and Switch Control

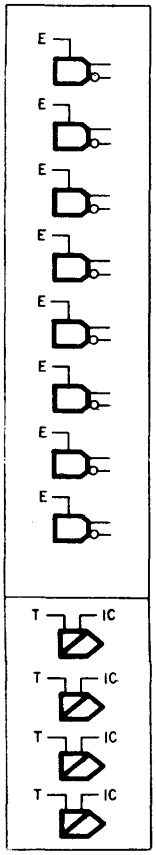




The D5.140 contains mode and time scale controls for the integrators as well as control for the electronic switches in three areas or one fourth of the AD/FIVE. When expansion Integrators/Summers are used, each set of integrator controls acts on both Integrator/Summers in a given area and row. (See D1.35 Integrator/Summer.)

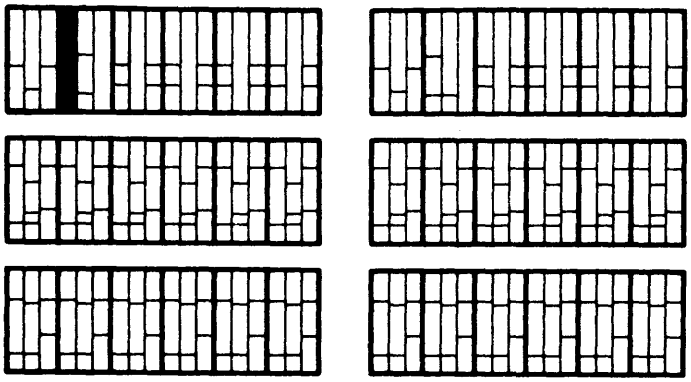
D5.141 Comparator and Track/Store Logic

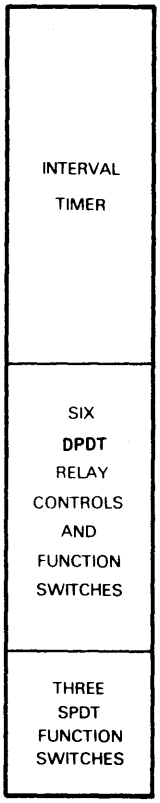




The D5.141 contains the logic terminals for the comparators and track/store networks for one field or one half of the AD/FIVE.

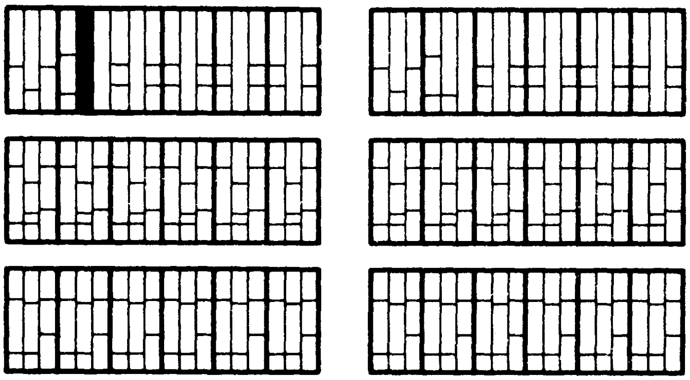
D5.142 Interval Timer/Relay Control Card

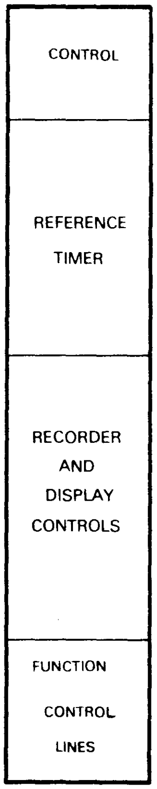




The D5.142 contains logic override terminations for the Interval Timer; outputs and logic overrides for six function switches found on the Logic Panel; and outputs from three logic switches on the Logic Panel.

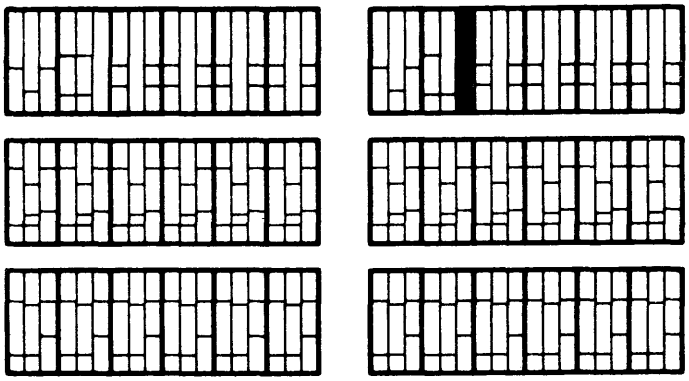
D5.143 Reference Timer/DSP Control Card





The D5.143 contains a 1MHz crystal oscillator and the circuit to select either this oscillator or an external oscillator as the logic system frequency generator; the card also contains circuits to generate the P and V signals which appear on the patchboard; finally, recorder control signals and a block of FNC control lines are contained on this circuit card.

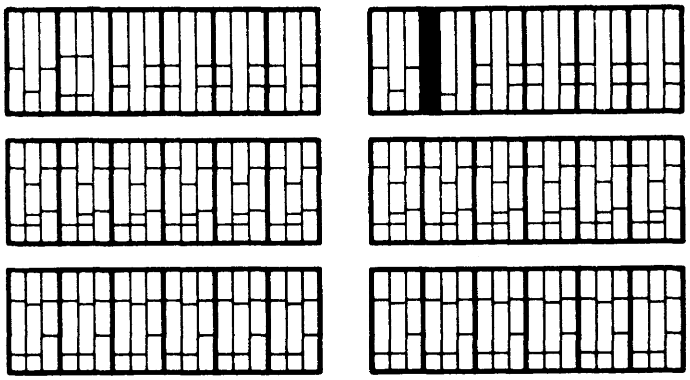
D5.172 DTS Logic Trunk Buffer



CONTROL
CIRCUITS
FOR
DYNAMICS
TERMINALS

The D5.172 contains address control, Interval Timer control, and logic trunk circuitry for coupling up to sixteen Dynamics Terminals to the AD/FIVE.

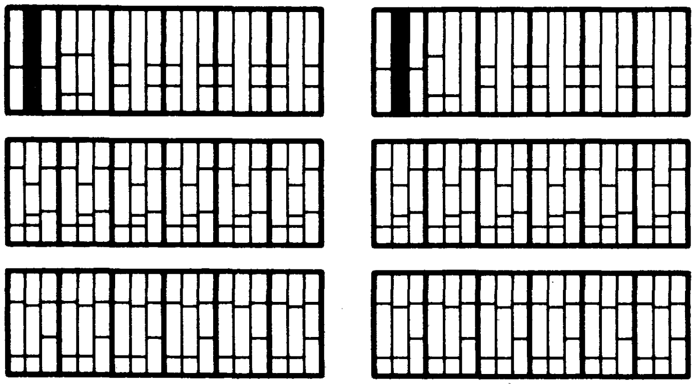
D9.162 DTS Analog Trunk Buffer

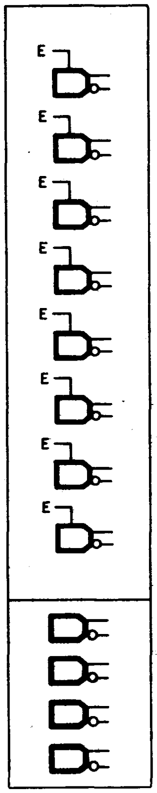


TRUNKS
TO
DYNAMICS
TERMINALS

The D9.162 provides eight trunks for Dynamics Terminal System DAC outputs, sixteen trunks for analog signals, and eight protected logic buffer circuits for the DTS configuration switch outputs.

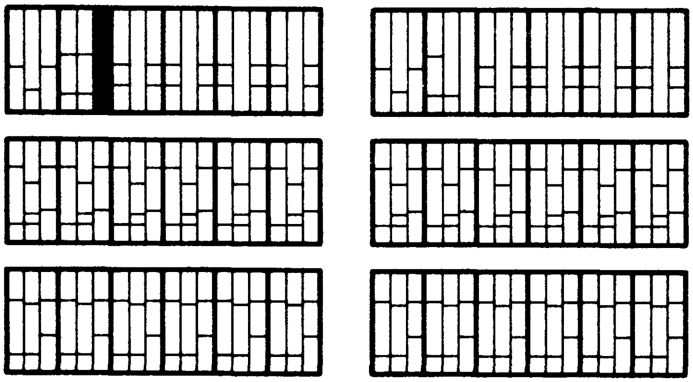
D5.166 Comparator Logic

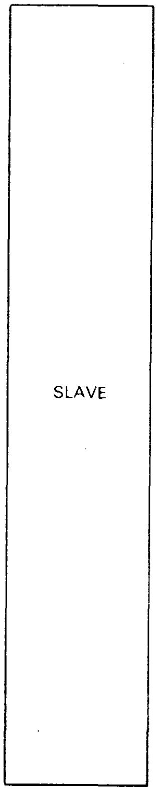




The D5.166 contains logic terminals for twelve comparators in one AD/FIVE field. The four additional comparators (beyond the normal eight) replace track/store networks, and are always enabled.

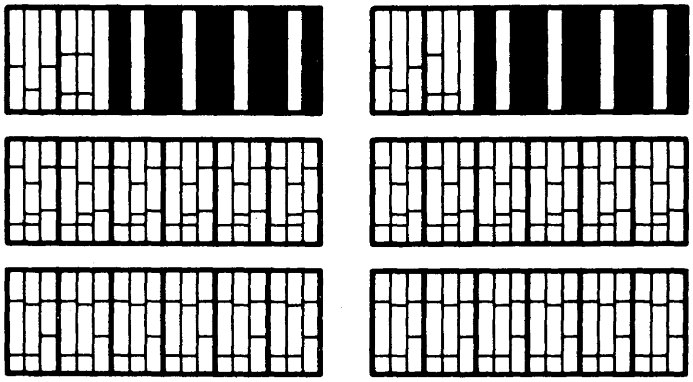
D5.144 Slave Control

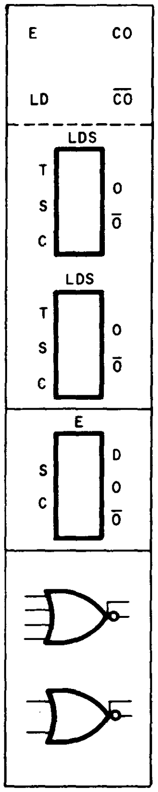




The D5.144 contains the control lines necessary for slaving. All terminals are provided for both master and slave operation of the AD/FIVE with another AD/FIVE.

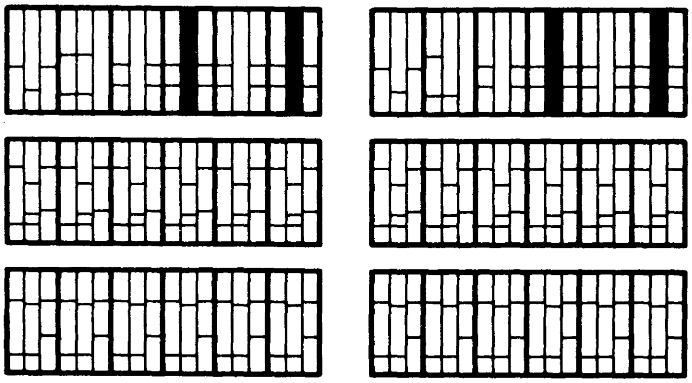
D5.145 Flip-Flops, Differentiators, and Gates

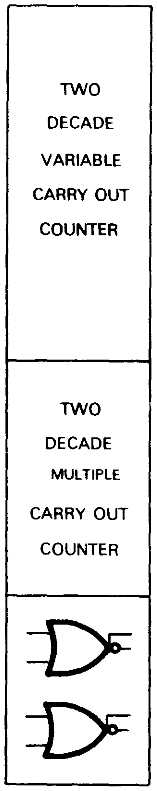




The D5.145 contains two flip-flops with load inputs and a special carry-out output for use as a counter, one clocked flip-flop differentiator (pulser), one four-input OR gate, and one two-input OR gate.

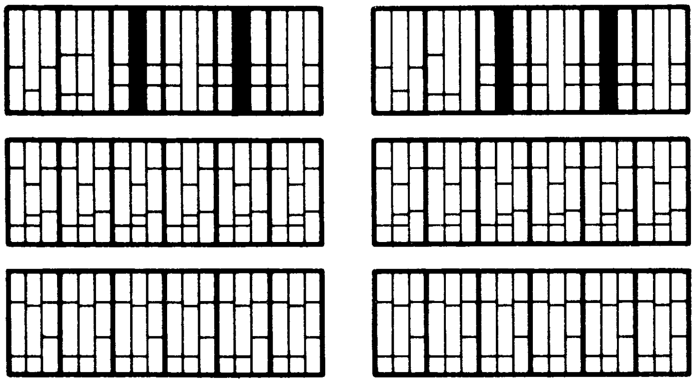
D5.146 BCD Counters and Gates

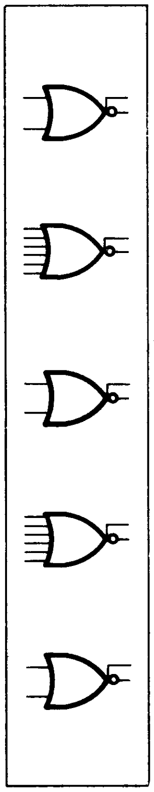




The D5.146 contains two two-decade BCD counters and two two-input OR gates. One of the BCD counters has a program or switch controlled carry-out and the other BCD counter has multiple carry-outs.

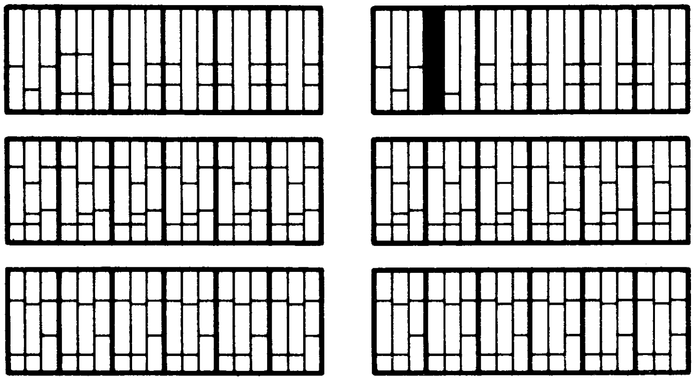
D5.147 Gates

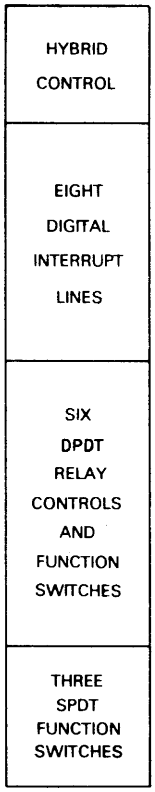




The D5.147 contains two six-input OR gates and three two-input OR gates.

D5.165 Hybrid Access and Switches





HYBRID CONTROL

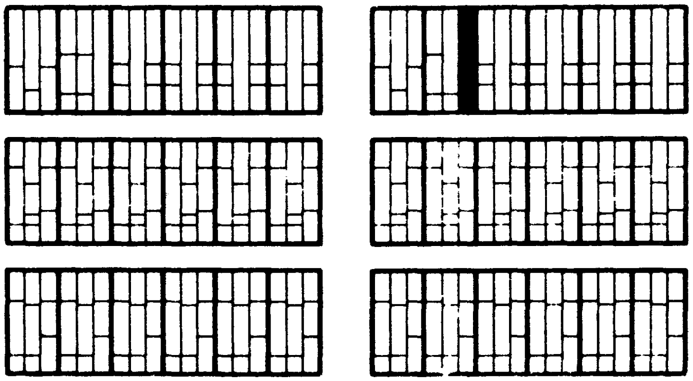
EIGHT DIGITAL INTERRUPT LINES

SIX DPDT RELAY CONTROLS AND FUNCTION SWITCHES

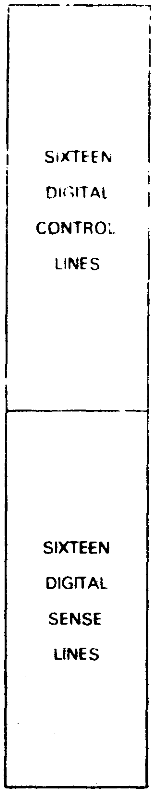
THREE SPDT FUNCTION SWITCHES

The D5.165 contains hybrid controls, eight digital interrupt lines, relay switch controls, and function switches.

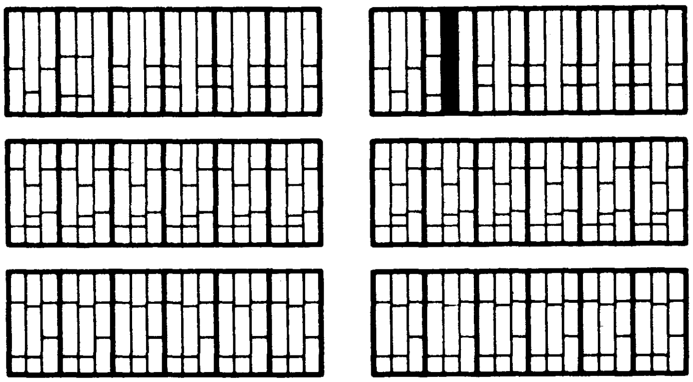
D9.136 Sense and Control Lines



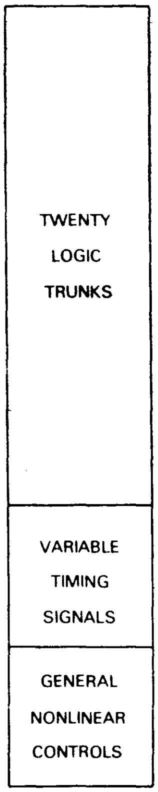
The D9.136 contains 16 digital sense lines and 16 digital control lines.



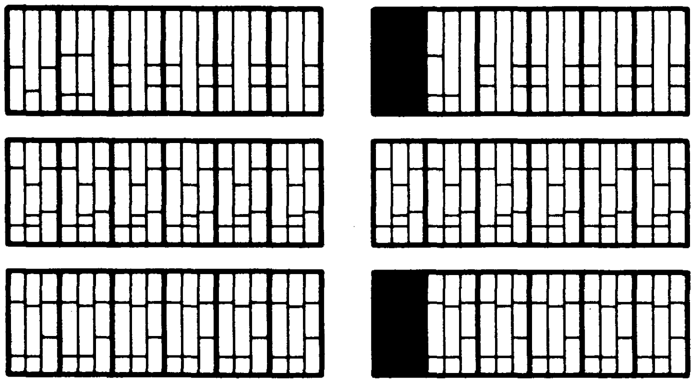
D9.137 Logic Trunks

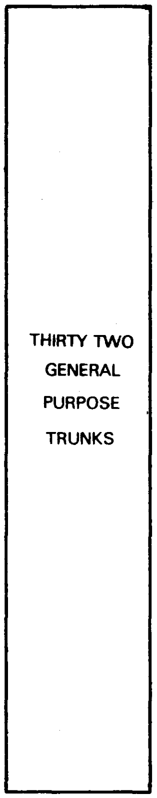


The D9.137 contains twenty logic trunks, access to the variable timing signals, and nonlinear component control lines for one AD/FIVE field.



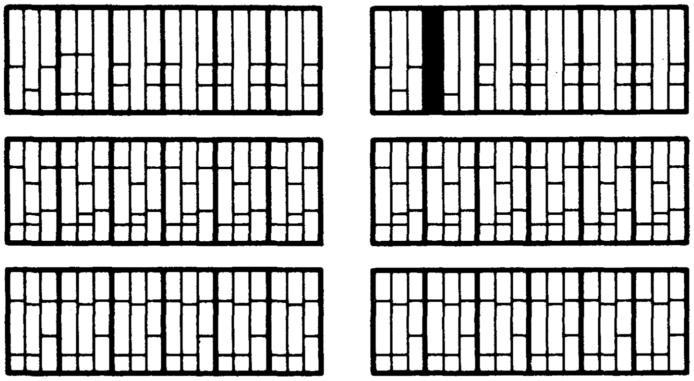
D9.143 General Purpose Trunks

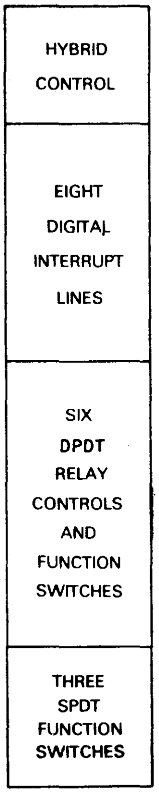




The D9.143 contains 32 general purpose, non-addressable trunks for use in an unexpanded AD/FIVE field.

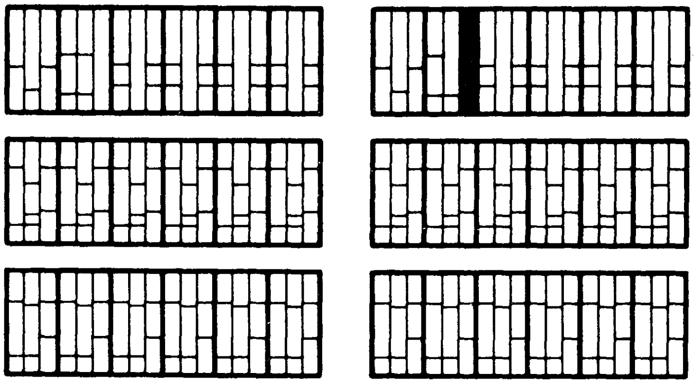
D9.142 Special Hybrid Access

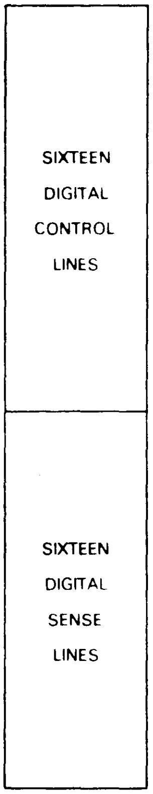




The D9.142 contains hybrid controls, and eight digital interrupt lines for use in an unexpanded AD/FIVE field.

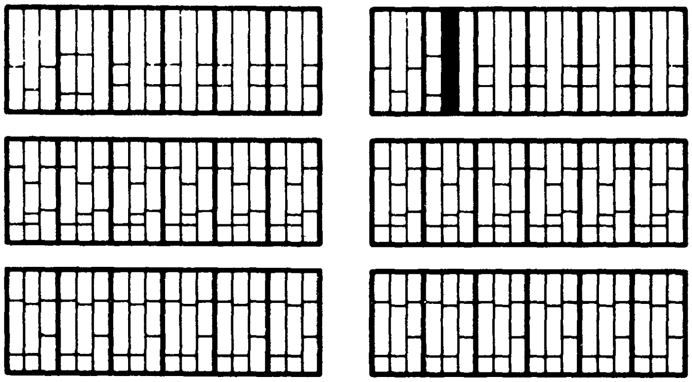
D9.141 Special Sense and Control Lines

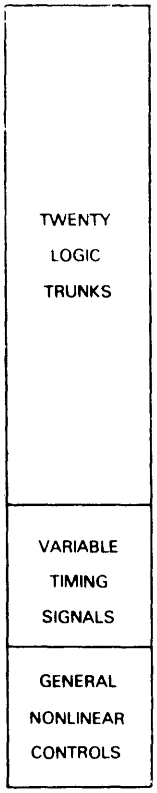




The D9.141 contains 16 digital sense lines and 16 digital control lines for use in an unexpanded AD/FIVE field.

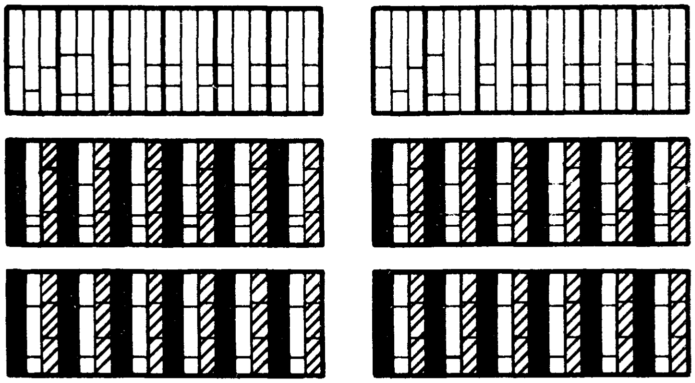
D9.140 Special Logic Trunks



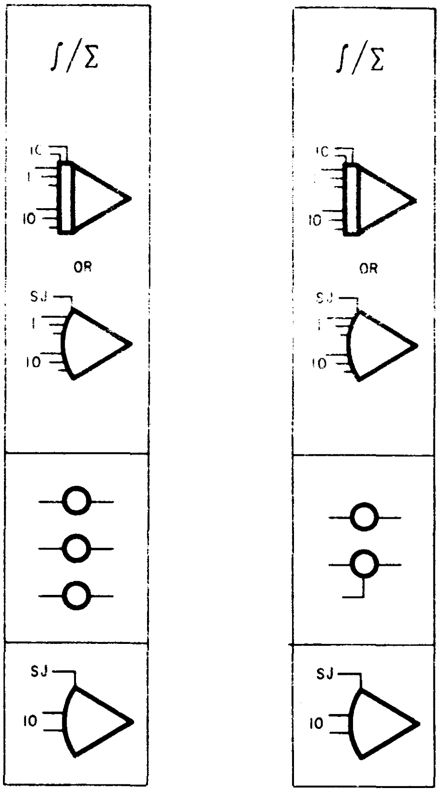


The D9.140 contains twenty logic trunks and access to the variable timing signals for use in an unexpanded AD/FIVE field.

D1.35 Integrator/Summer



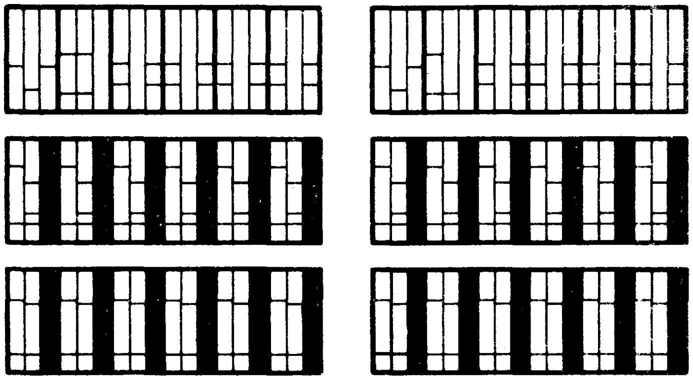
*

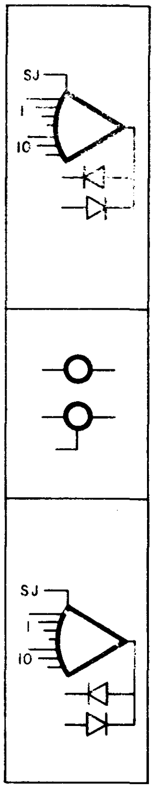


The D1.35 contains a combination integrator/summer, access to an inverter, and access for three grounded pots or one grounded pot and one ungrounded pot.

* Replaces the D1.36 Summer Card

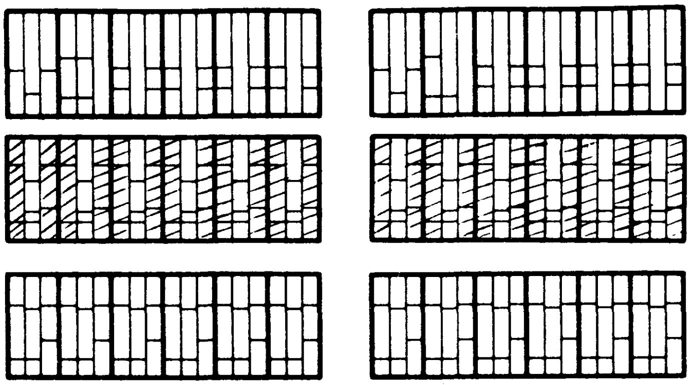
D1.36 Summers

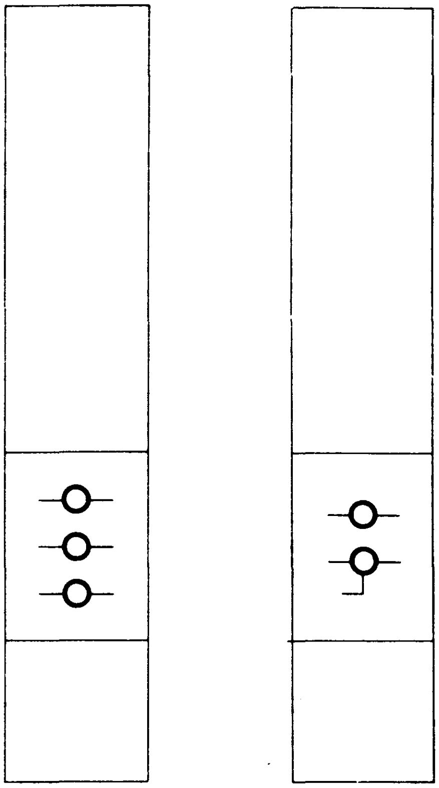




The D1.36 contains two summing amplifiers and access to one grounded potentiometer and one ungrounded potentiometer.

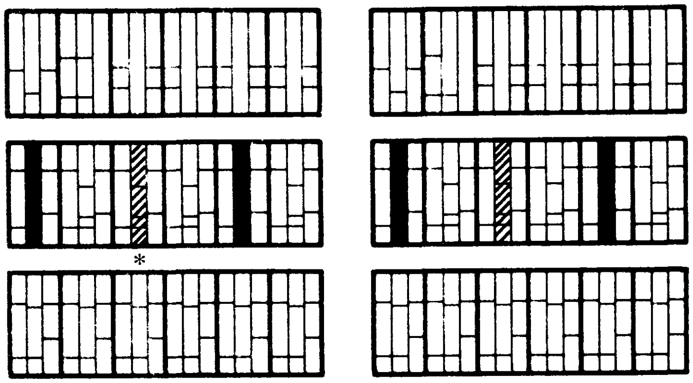
D9.146 Pot Access Card

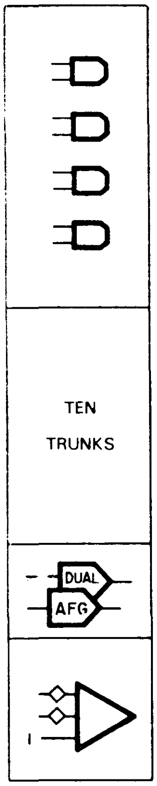




The D9.146 contains access to three grounded pots or two grounded and one ungrounded pot. The D9.146 replaces either the D1.35 or D1.36.

Di.37 Comparators

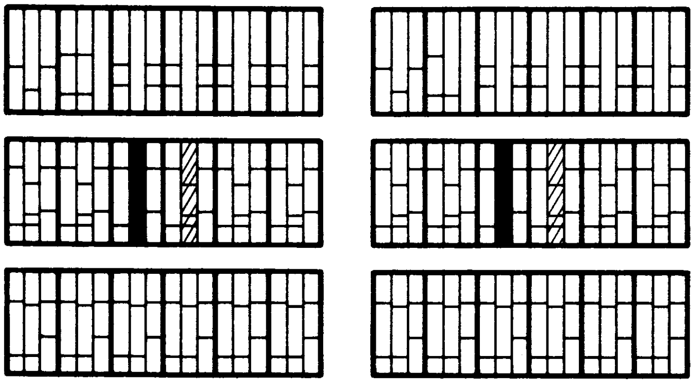


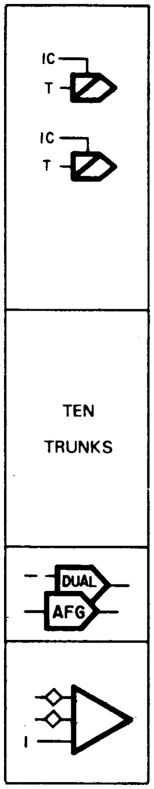


The D1.37 contains four comparators, access for one dual arbitrary function generator, two electronic switches terminated in a summer, and provision for ten analog trunks. In field 0, area 0, the trunks are replaced with system lines such as ASO, VM, ±IC, ±T, etc.

* Replaces the D1.38 Track/Store Network Card

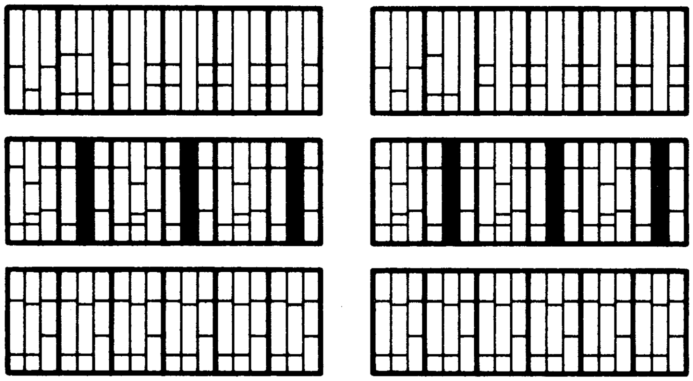
D1.38 Track/Store Networks

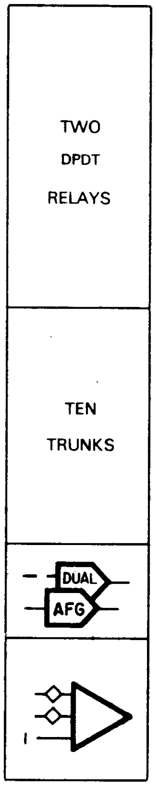




The D1.38 contains two track/store networks, access for a dual arbitrary function generator, two electronic switches terminated in a summer, and ten analog trunks.

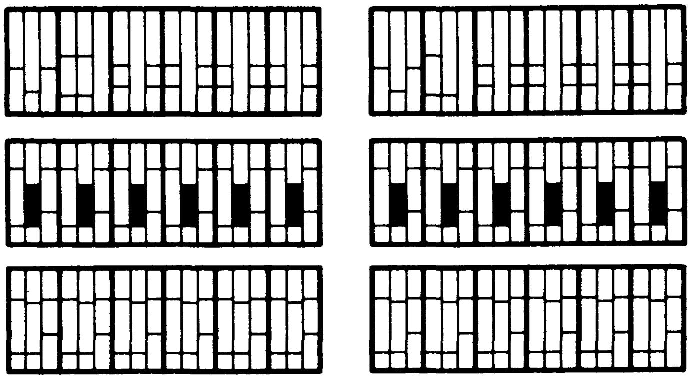
D1.40 Relays

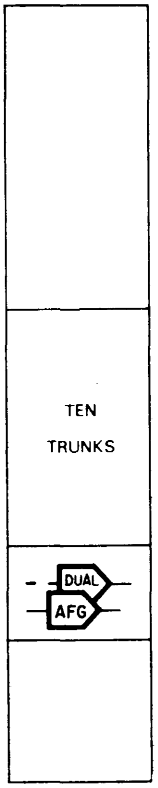




The D1.40 contains two relay switches, access for one arbitrary function generator, two electronic switches terminated in a summer, and ten analog trunks.

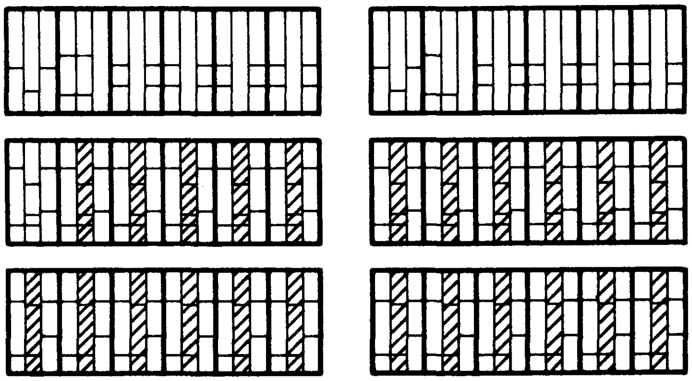
D9.147 DFG, Trunk Access

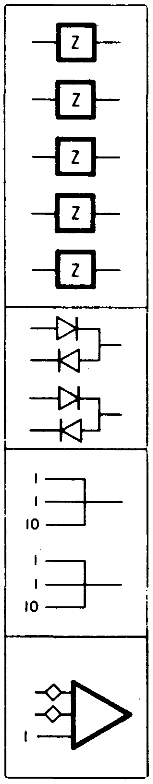




The D9.147 provides access to ten trunks and the arbitrary function generator in that area. No addressing is provided on this simple feed-through card.

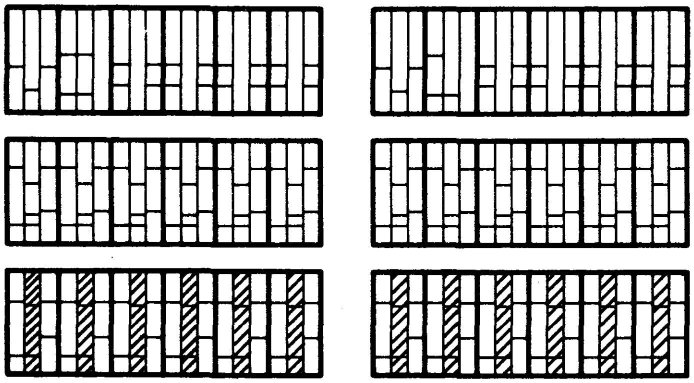
D2.18 Free Networks

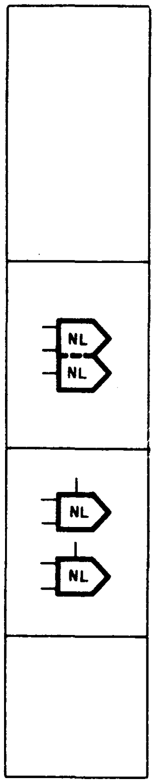




The D2.18 contains two electronic switches terminated in a summer, two free diode pairs, two free resistor networks, and provision for the customer to mount his own free impedance networks. The D2.18 replaces any center analog card except Field 0 Area 0 where there are System Lines.

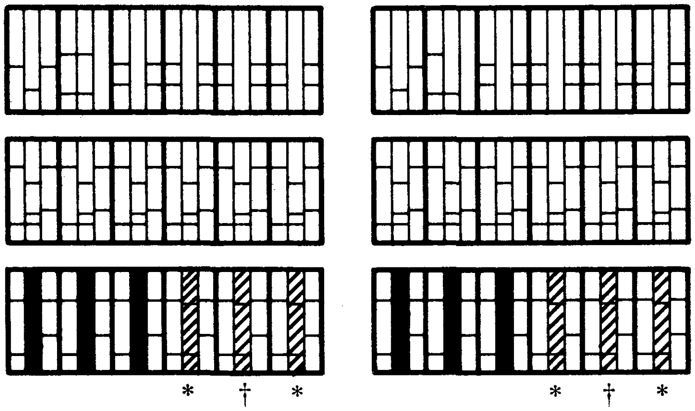
D9.138 Nonlinear Access

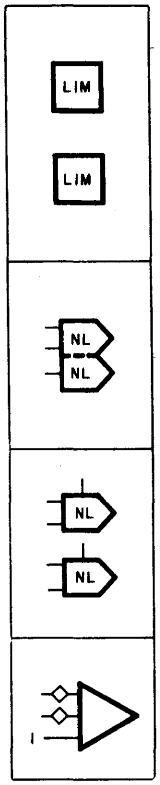




The D9.138 contains access routes for one convertible nonlinear element and two non-convertible nonlinear elements. The D9.138 replaces any center card in the Lower Analog area.

D1.39 Limiters



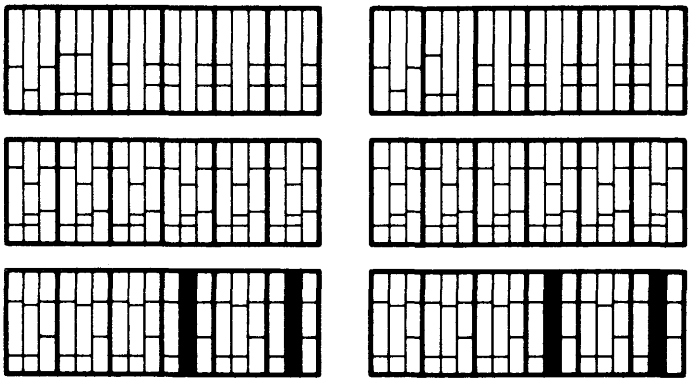


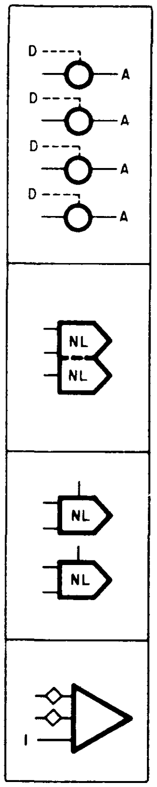
The D1.39 contains two variable hard limiters, one access route for a convertible nonlinear element, two access routes for non-convertible nonlinear elements, and two electronic switches terminated in a summer. A non-convertible, nonlinear element may be used in a convertible area.

* Replaces the D4.46 Digital to Analog converter card.

† Replaces the D4.47 Analog to Digital Lines.

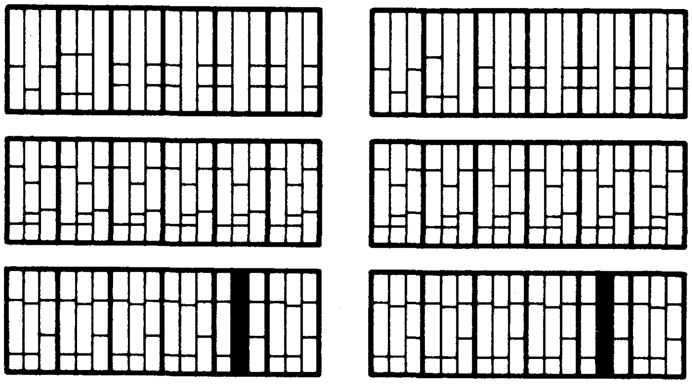
D4.46 Digital to Analog Converters

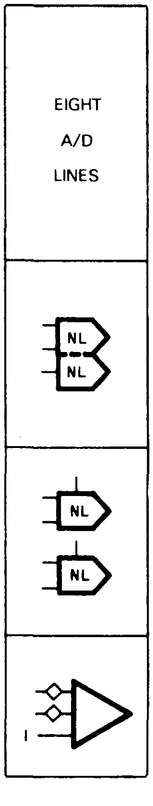




The D4.46 contains access routes for four multiplying digital to analog converters, one access route for a convertible nonlinear element, two access routes for non-convertible nonlinear elements, and two electronic switches terminated in a summer. A non-convertible nonlinear element may be used in a convertible area.

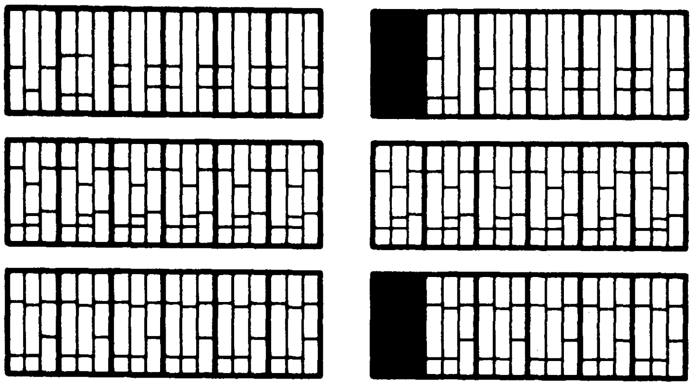
D4.47 Analog to Digital Conversion

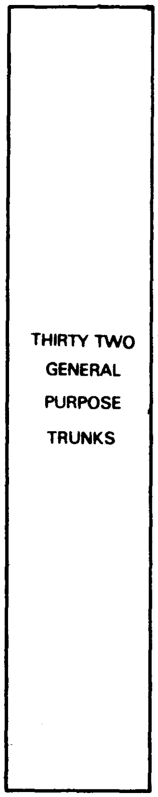




The D4.47 contains eight access routes for analog to digital conversion, one access route for a convertible nonlinear, two access routes for non-convertible nonlinear elements, and two electronic switches terminated in a summer. A non-convertible nonlinear element may be used in a convertible area.

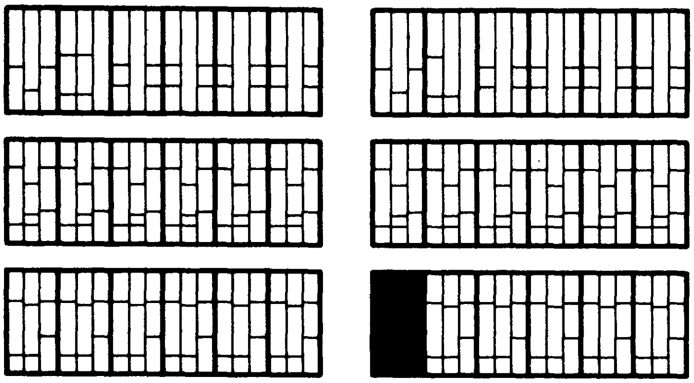
D9.143 General Purpose Trunks

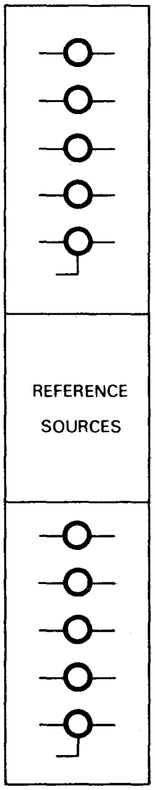




The D9.143 contains 32 general purpose, non-addressable trunks for use in an unexpanded AD/FIVE field.

D9.139 External Pots

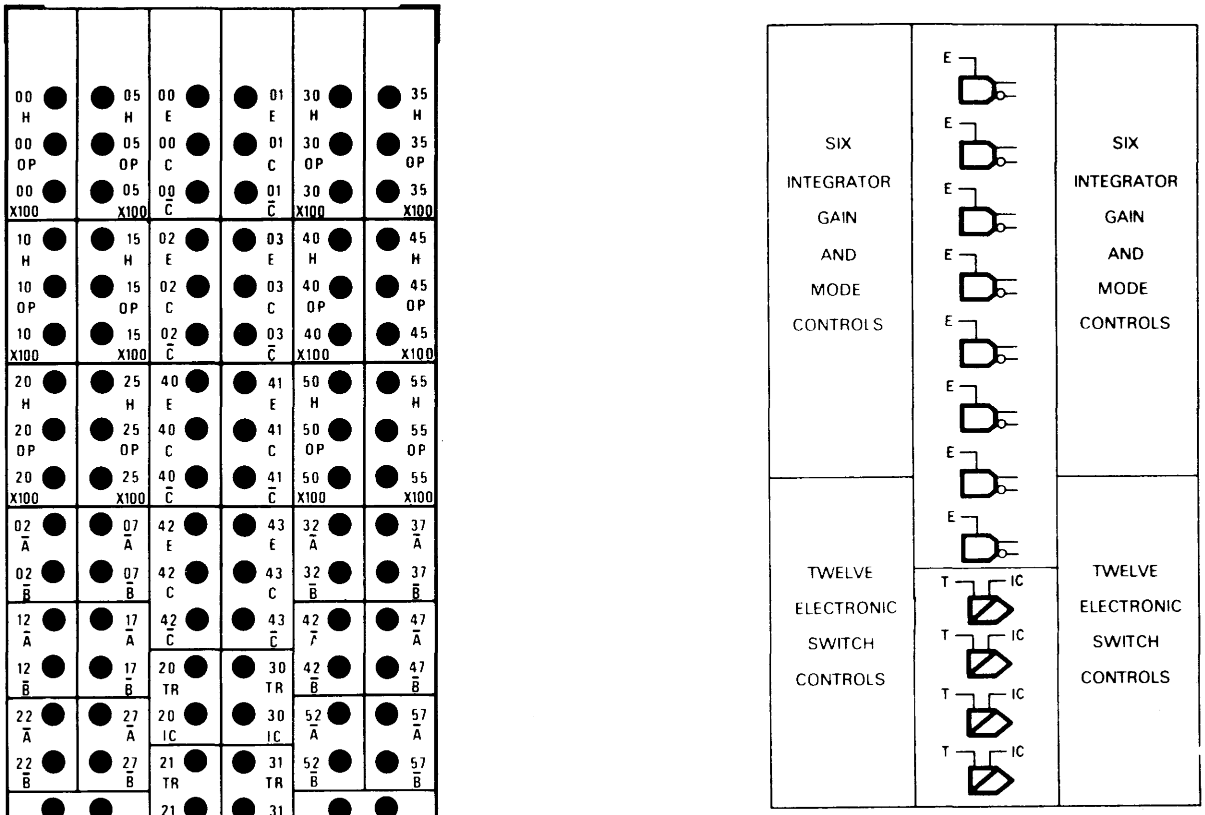




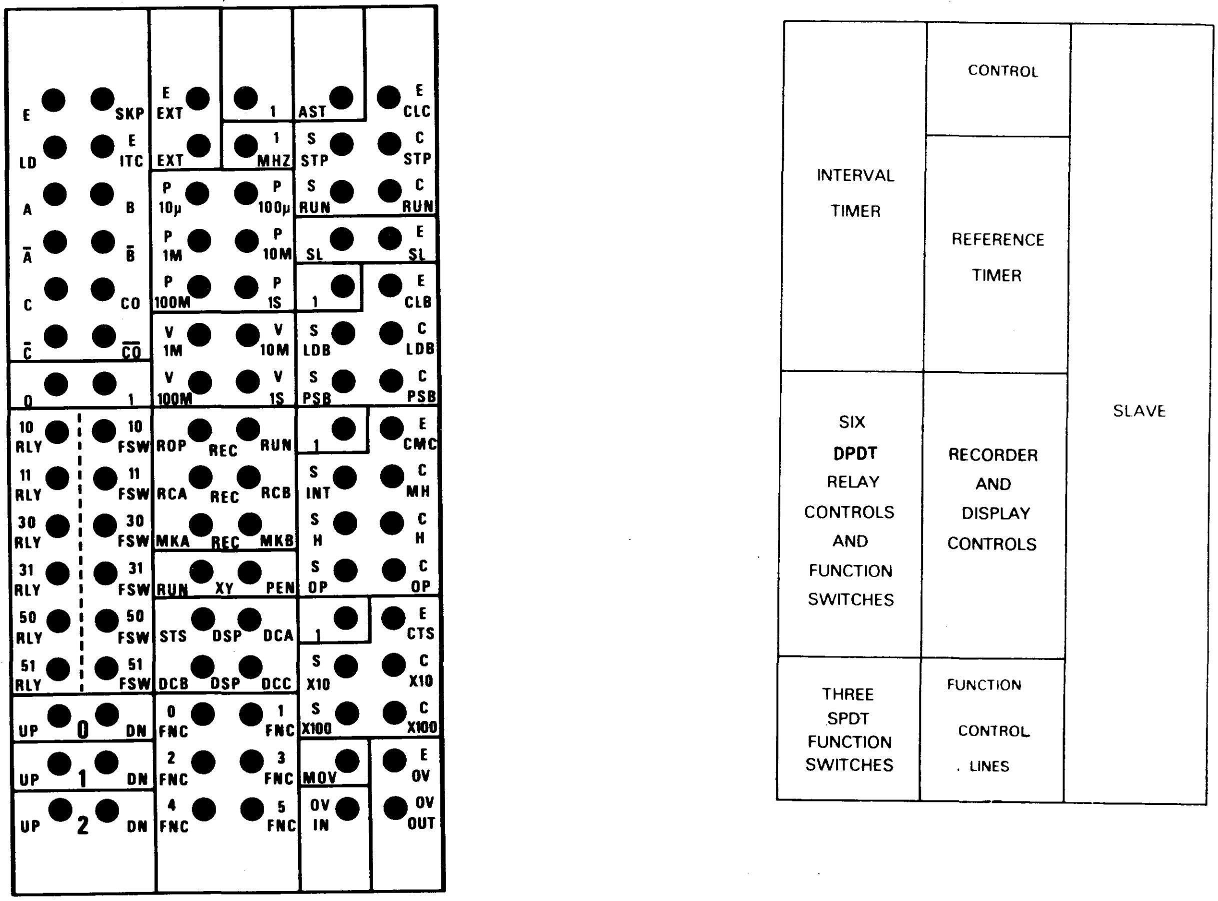
The D9.139 contains reference sources and access routes for 10 external potentiometers to be terminated in an unexpanded AD/FIVE field.

4.4 PATCHBOARD MODULES

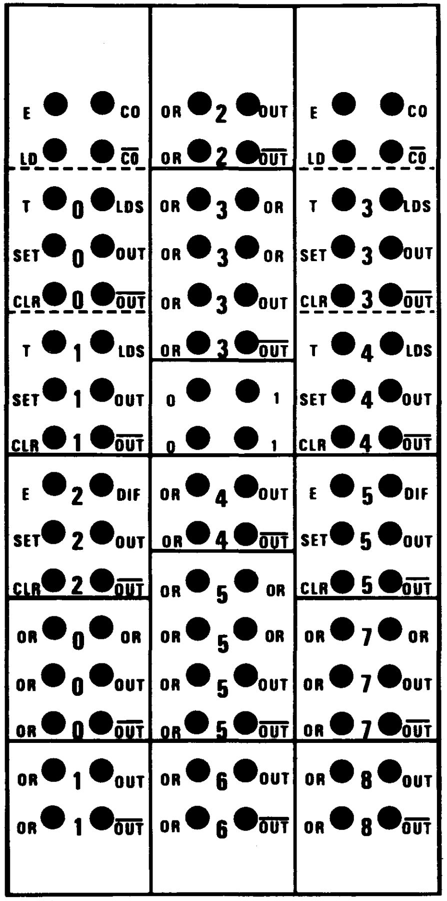
This section is a catalog of available patchboard modules. Each of the modules provides patching to three of the printed circuit cards.

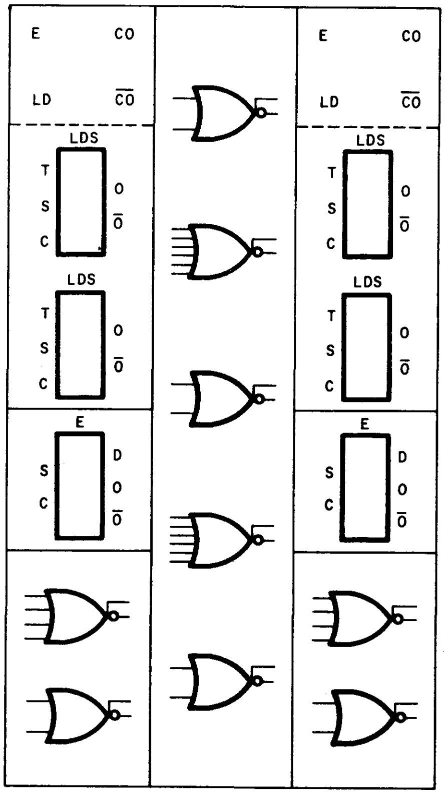


MODULE 1 C7.86

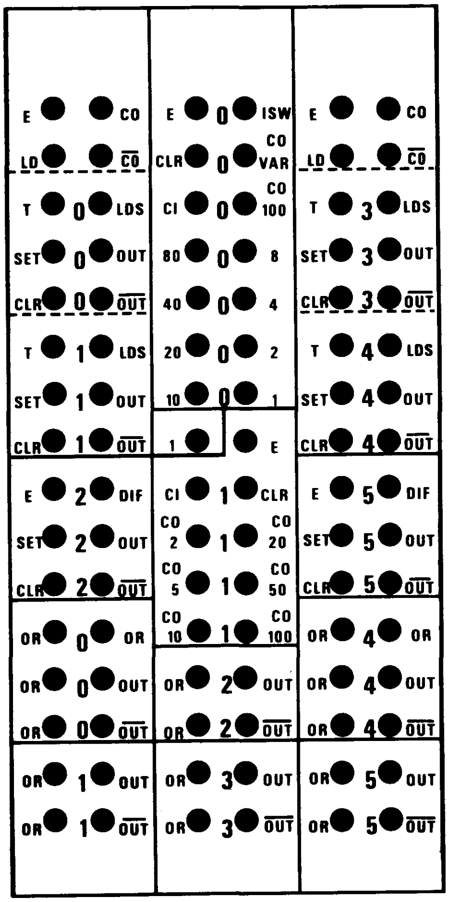


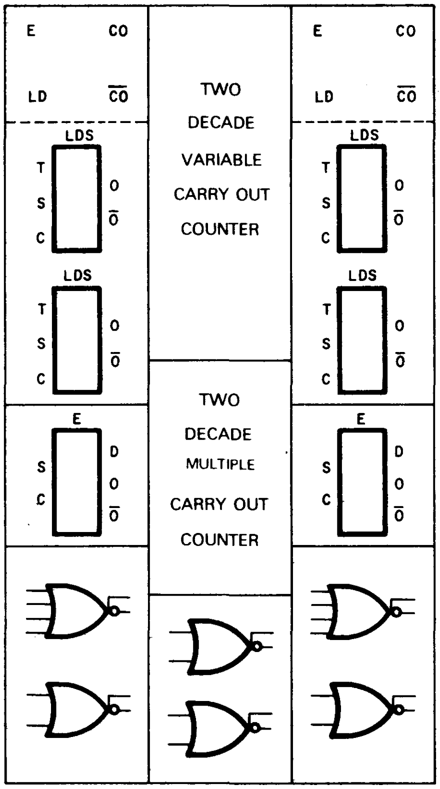
MODULE 2 C7.87



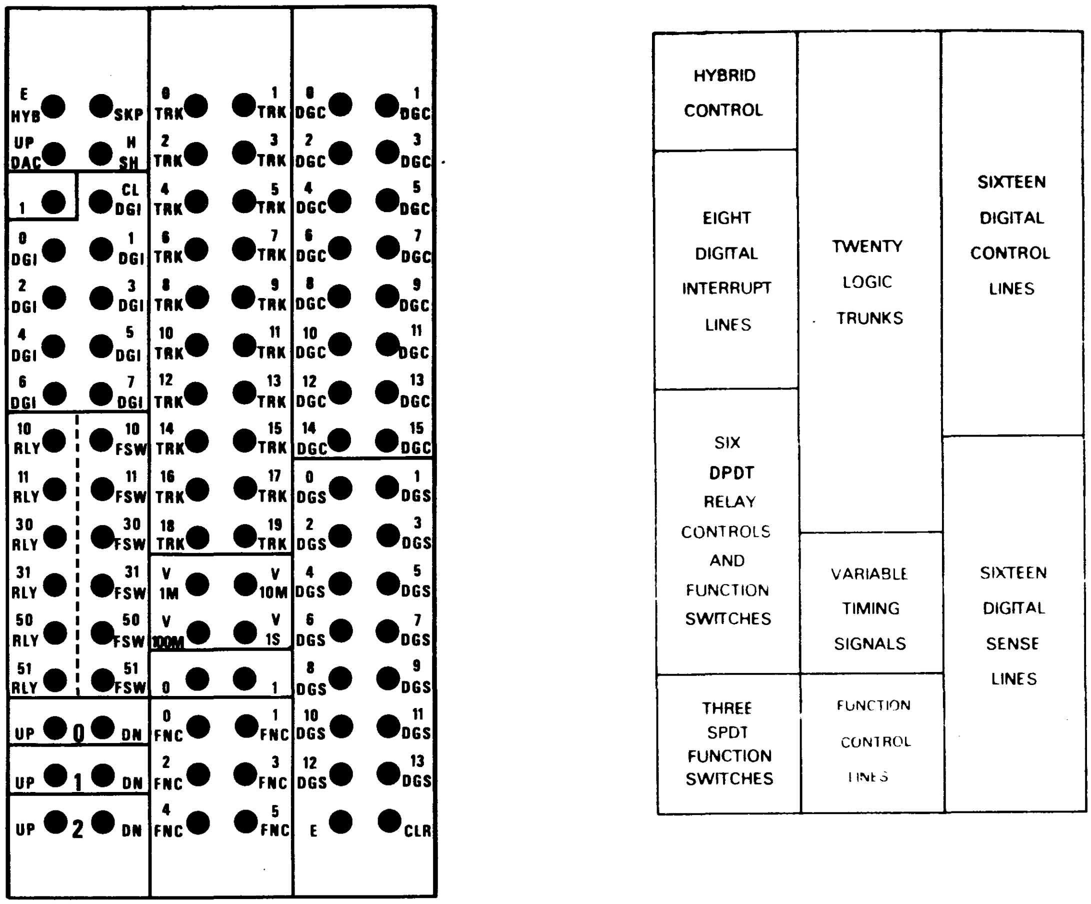


MODULE 3 C7.88

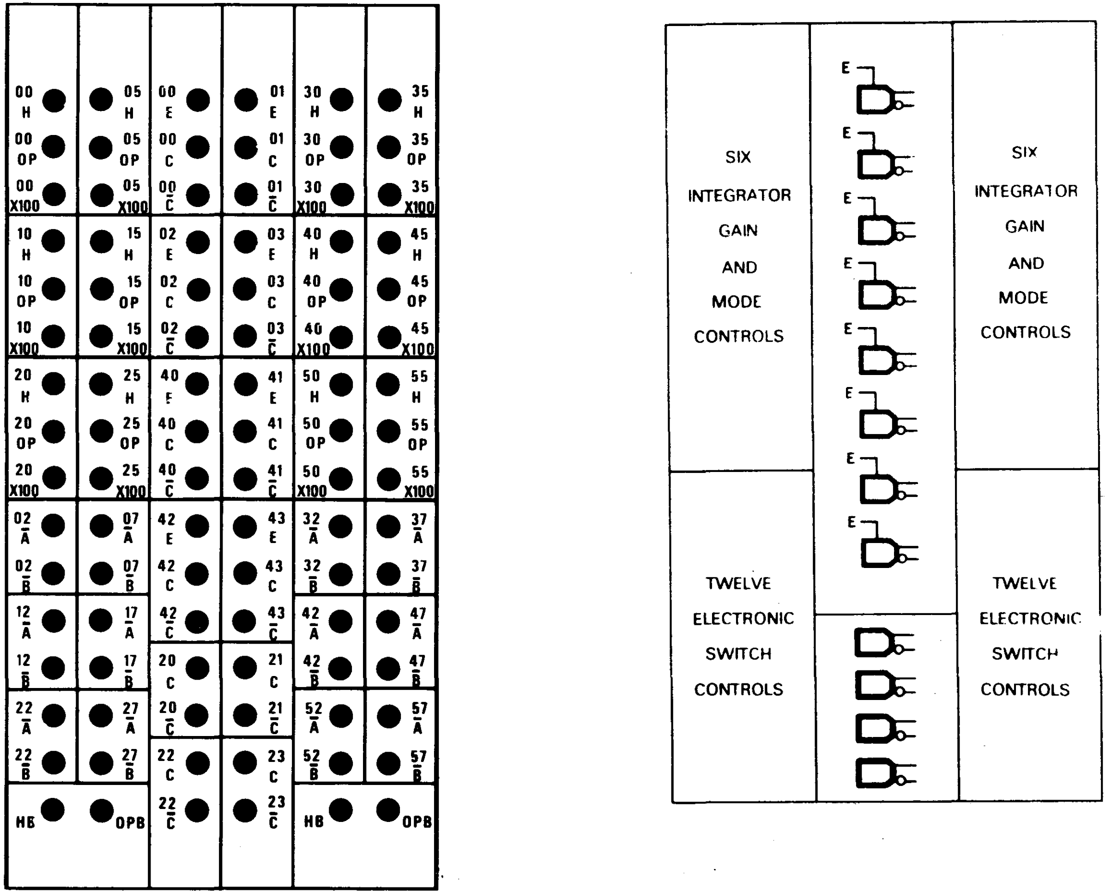




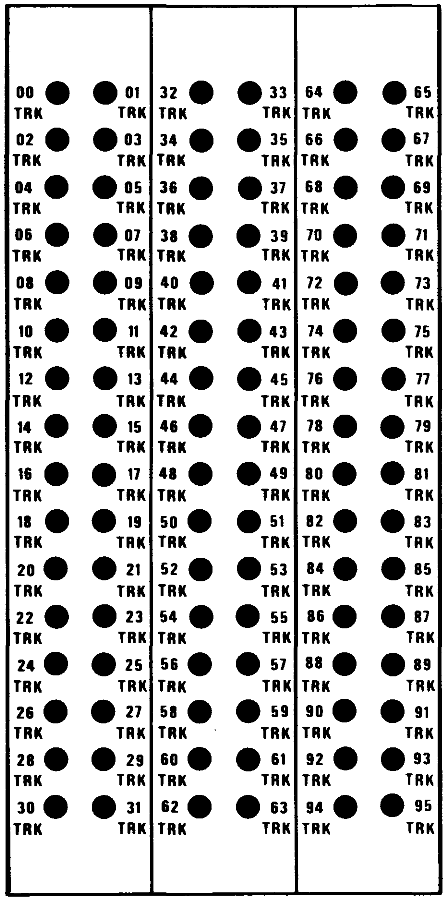
MODULE 4 C7.89



MODULE 5 C7.90

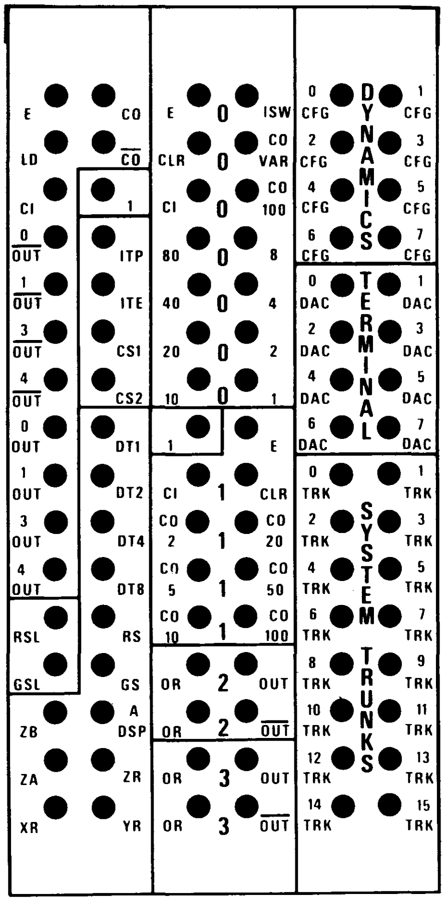


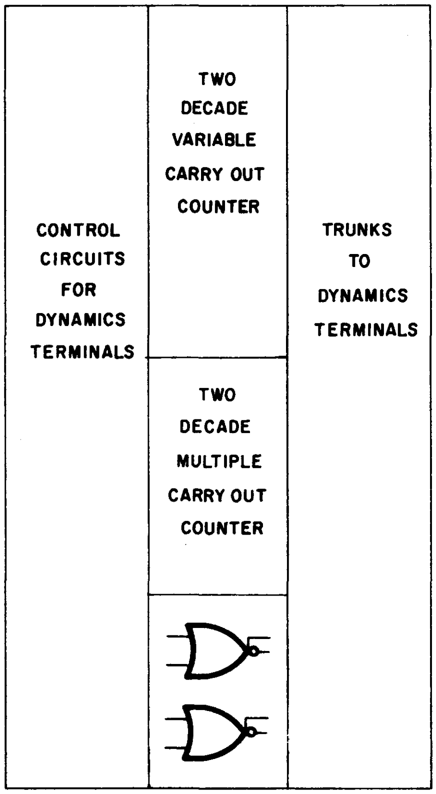
MODULE 6 C7.91

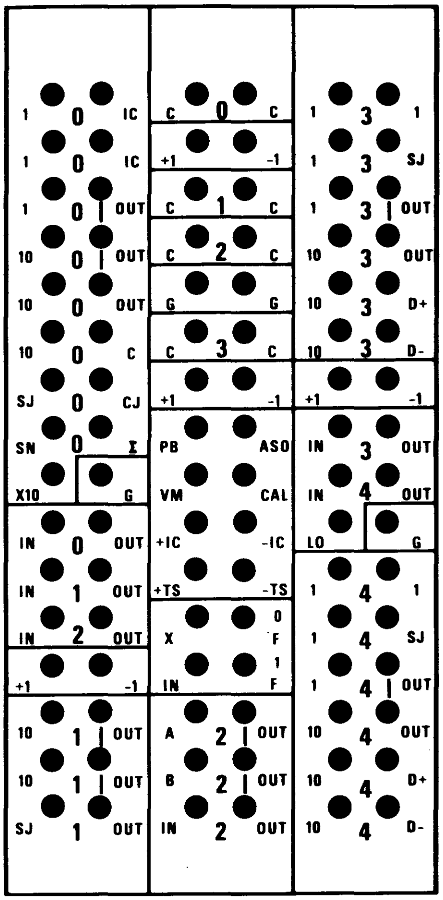


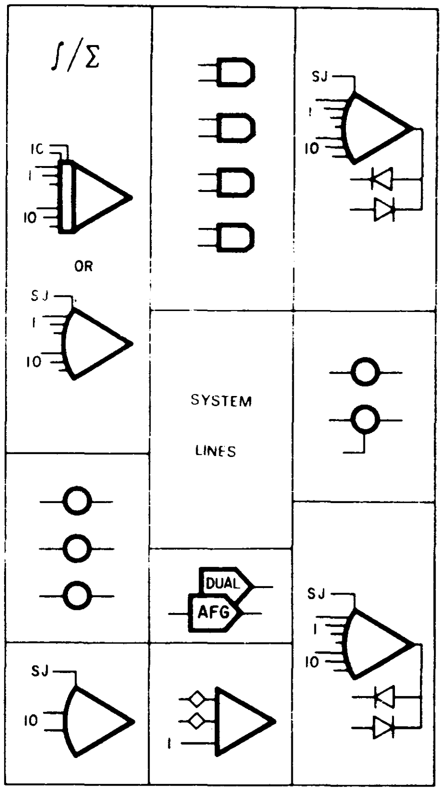
NINETY
SIX
GENERAL
PURPOSE
TRUNKS

MODULE 7 C7.92

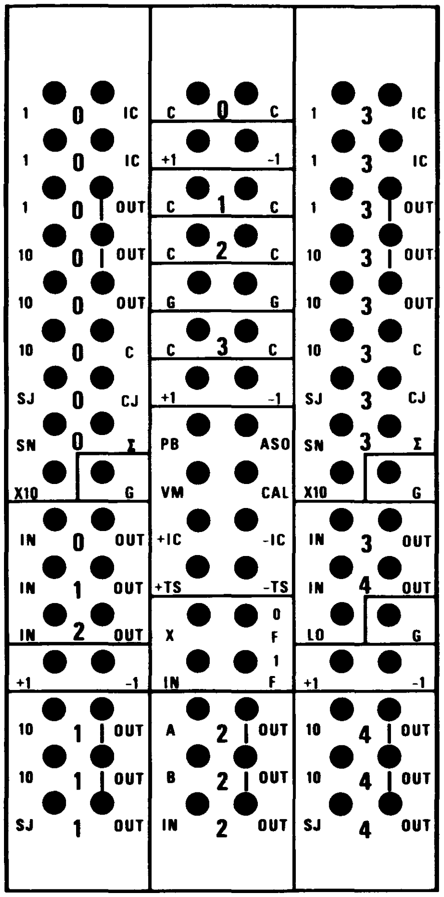


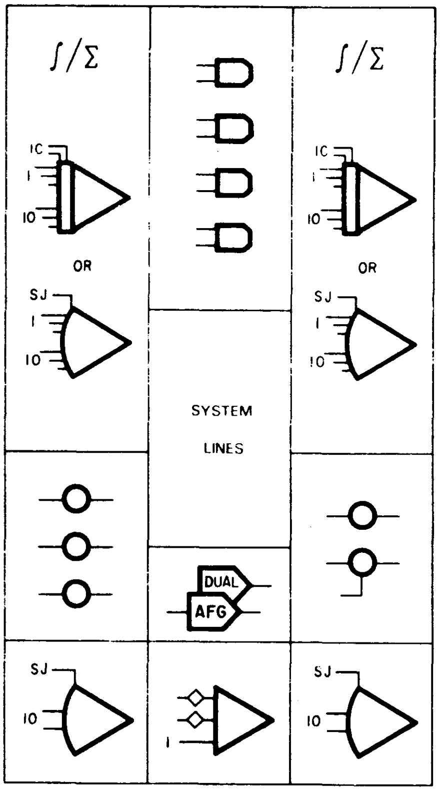

CONTROL CIRCUITS FOR DYNAMICS TERMINALS
TWO DECADE VARIABLE CARRY OUT COUNTER
TWO DECADE MULTIPLE CARRY OUT COUNTER
TRUNKS TO DYNAMICS TERMINALS



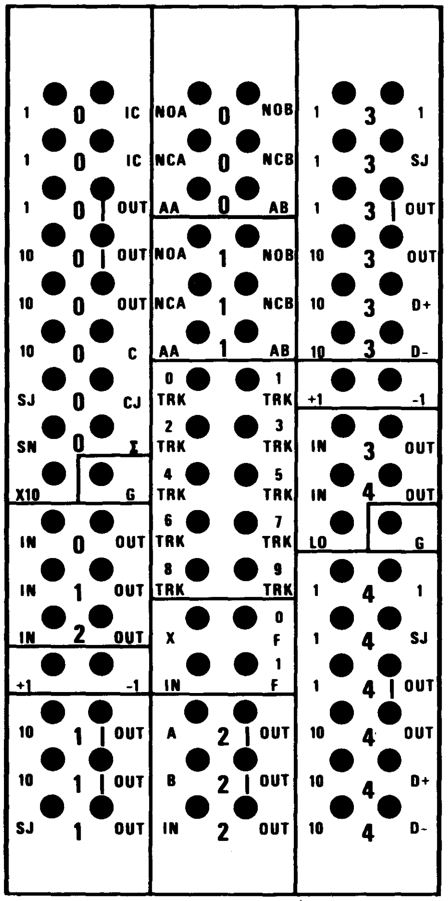


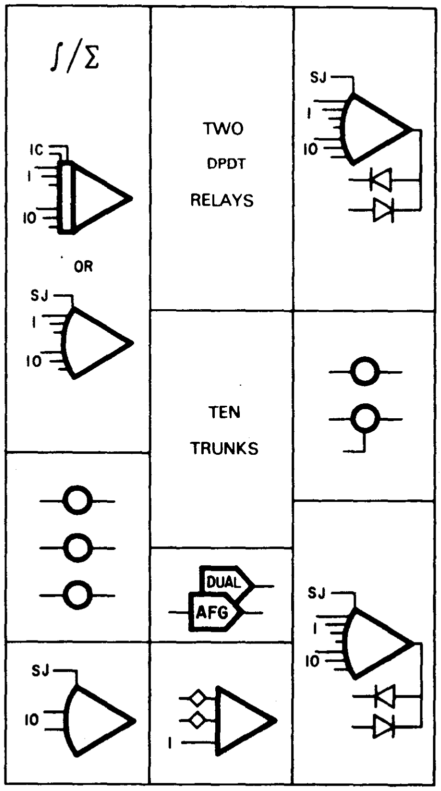
MODULE 10 C7.95



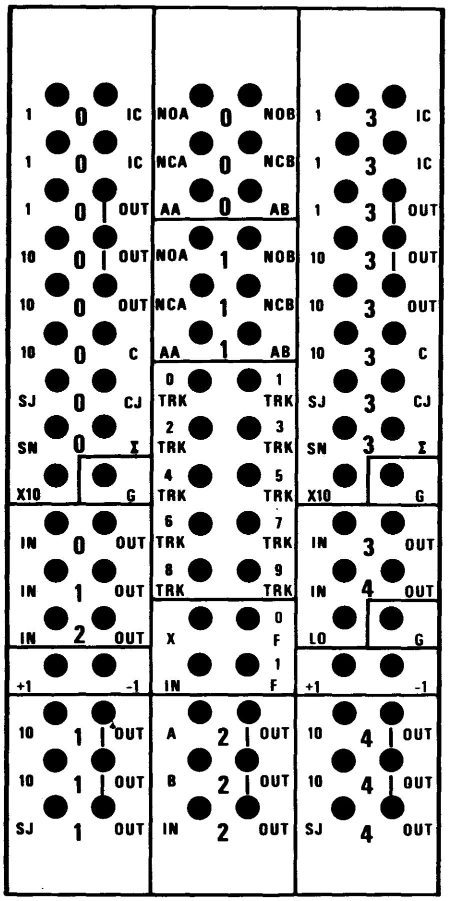


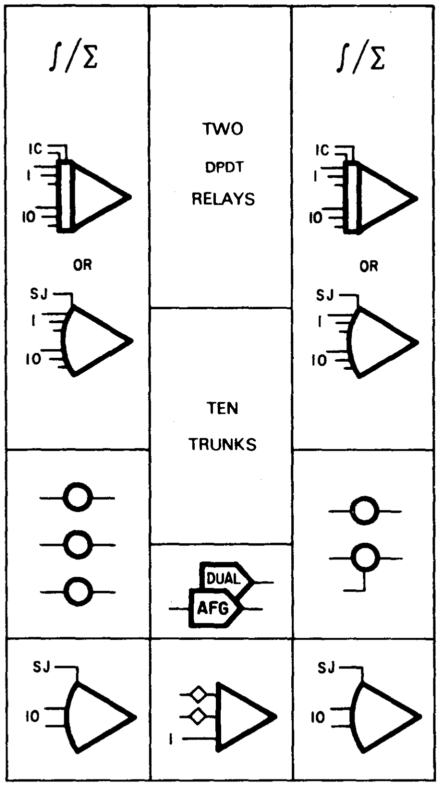
MODULE 11 C7.96



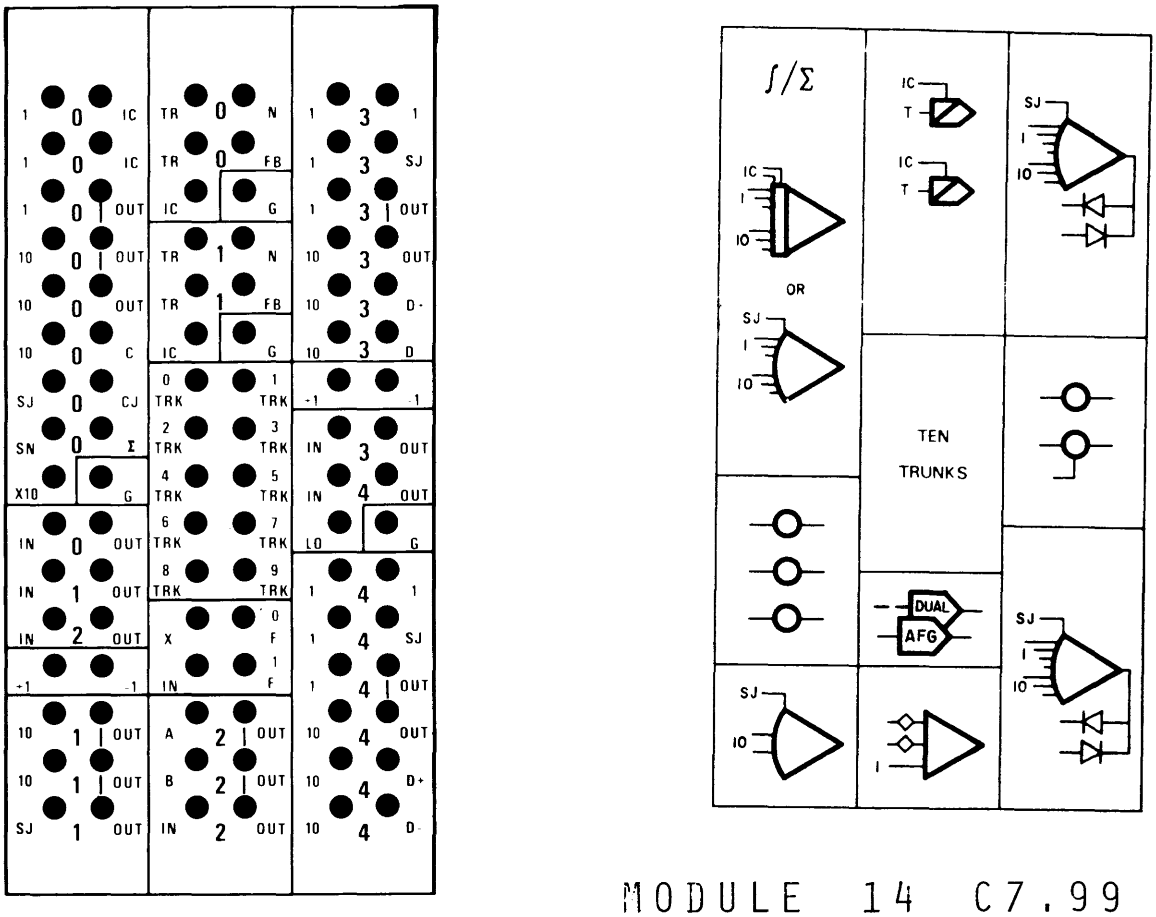


MODULE 12 C7.97

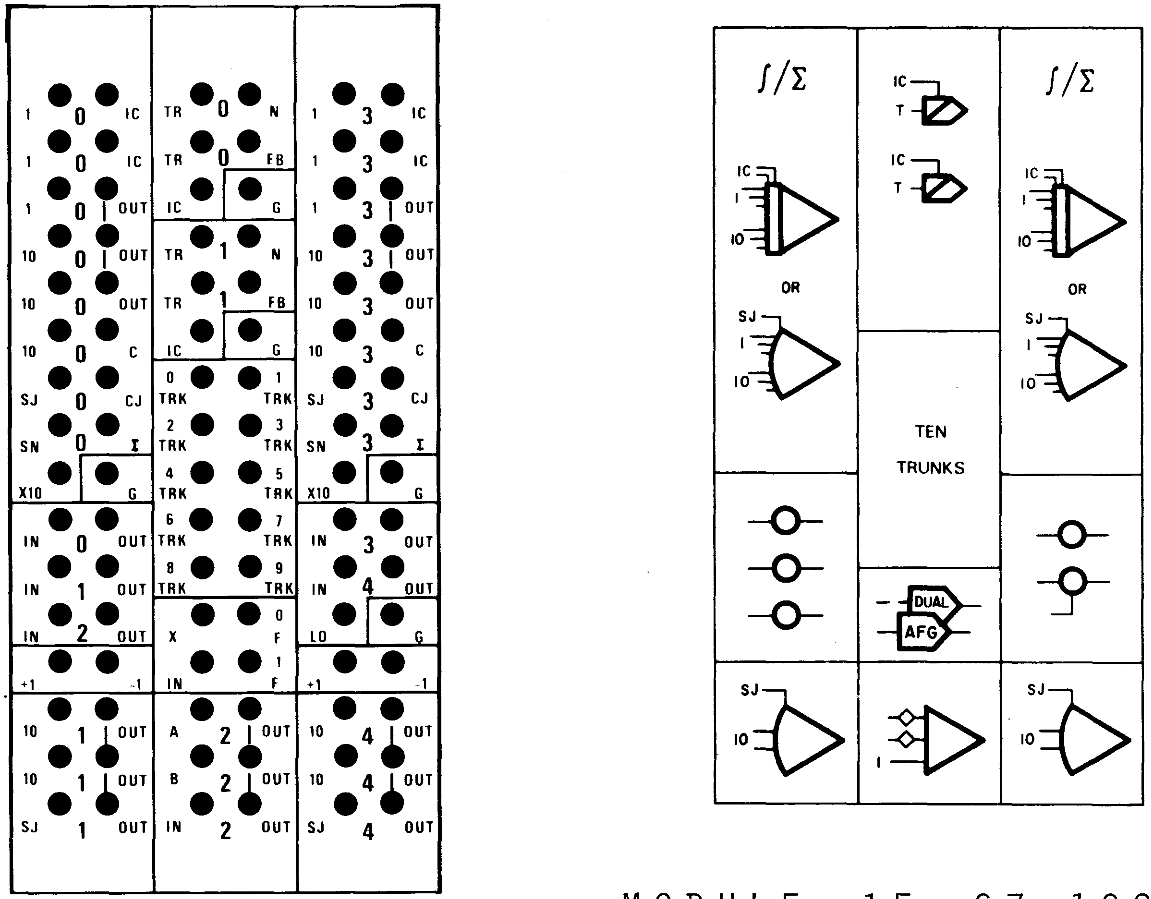




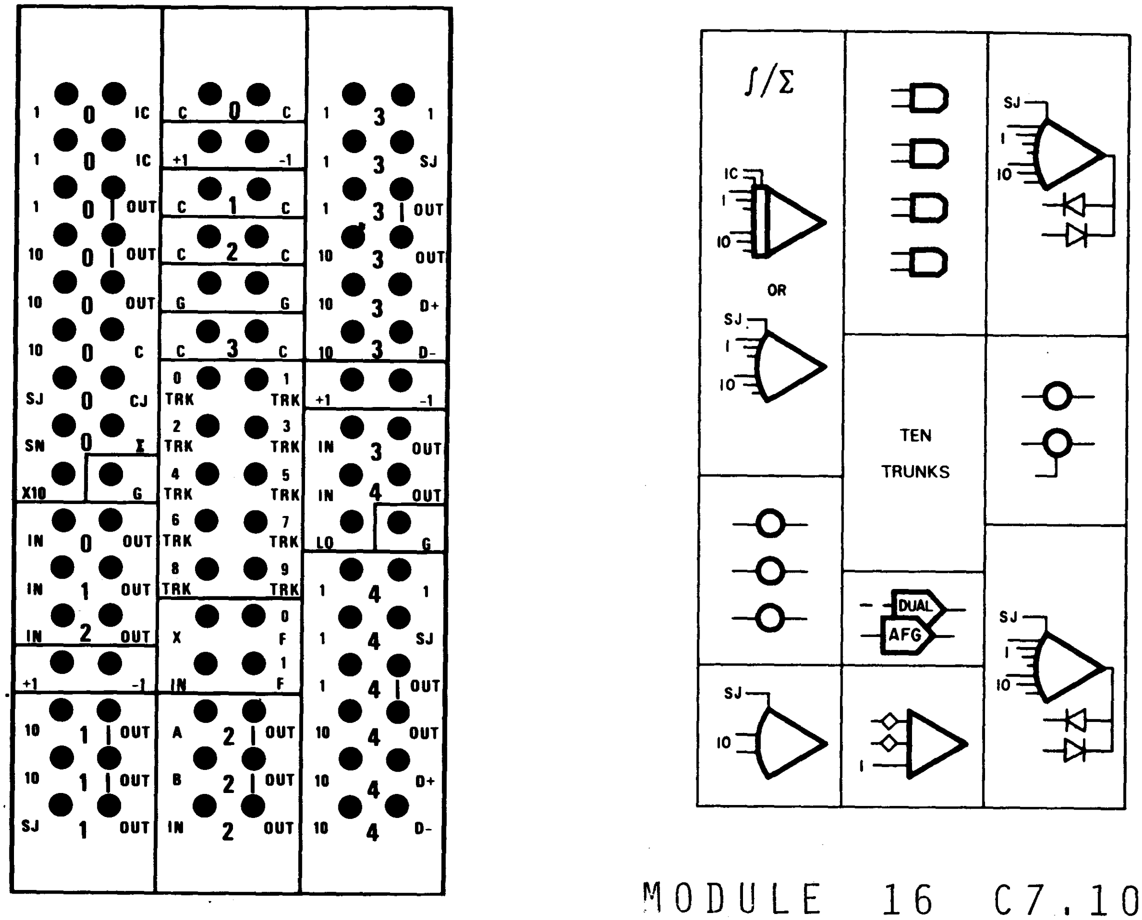
MODULE 13 C7.98



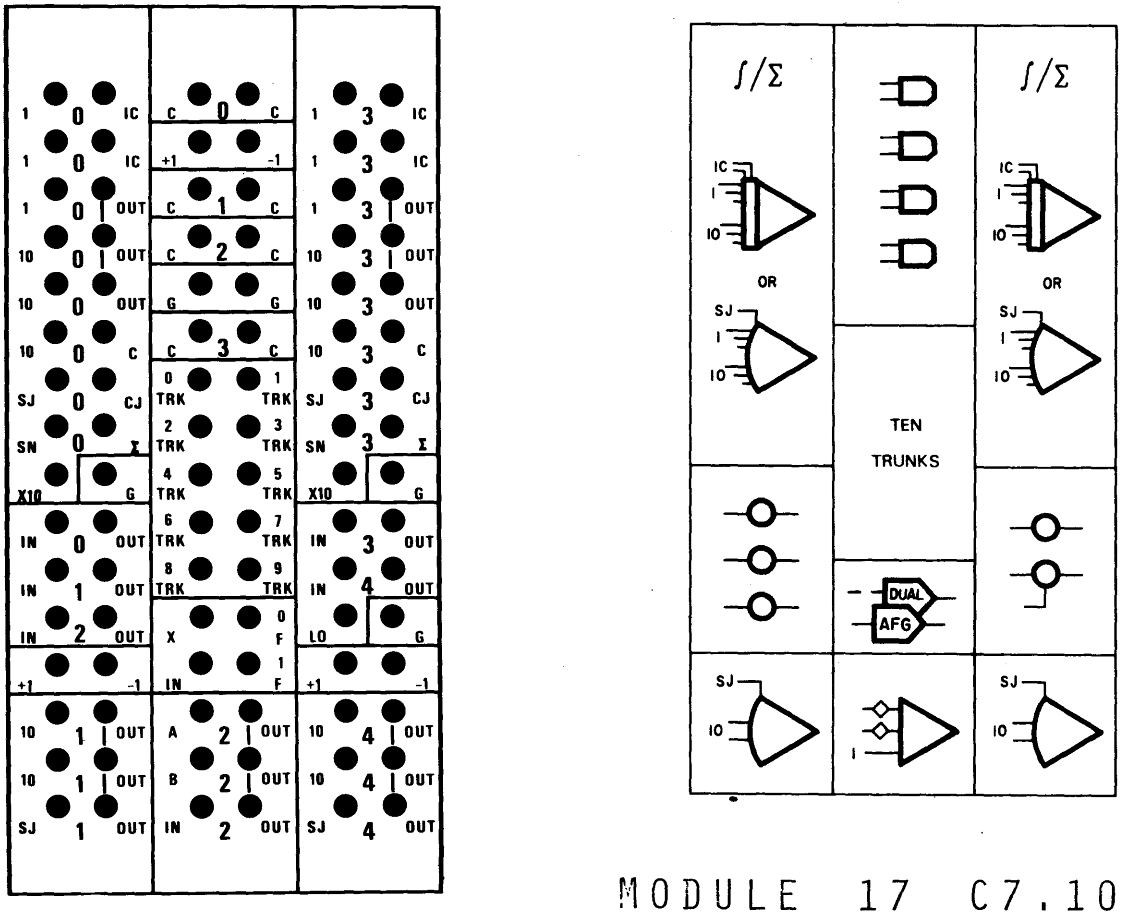
MODULE 14 C7.99



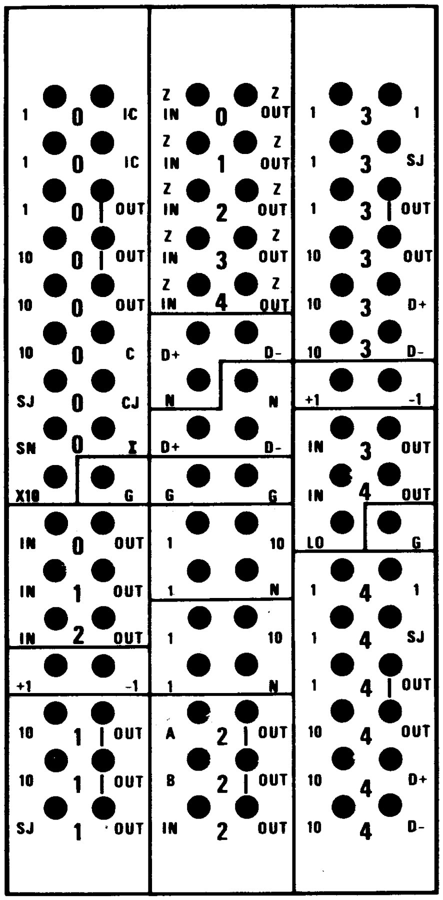
MODULE 15 C7.100

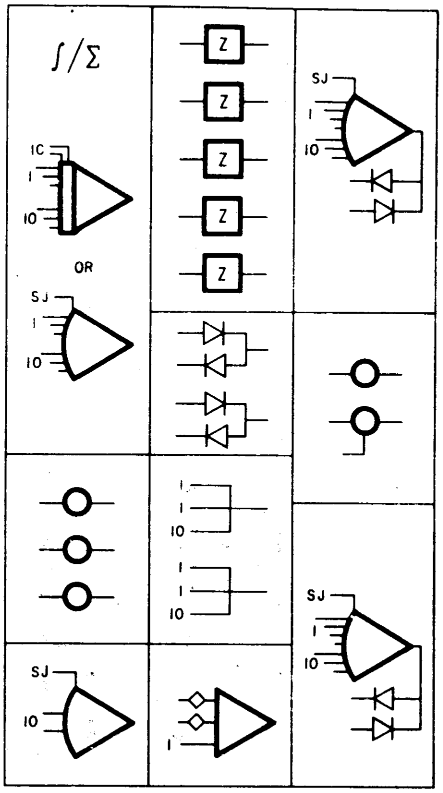


MODULE 16 C7.101

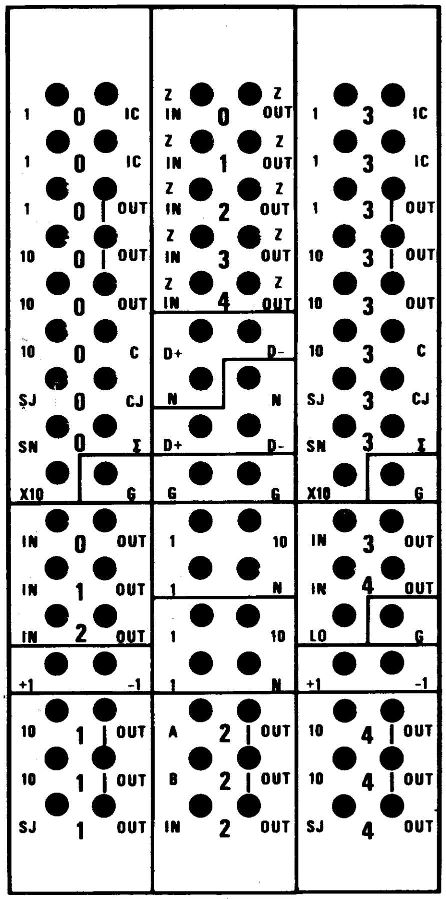


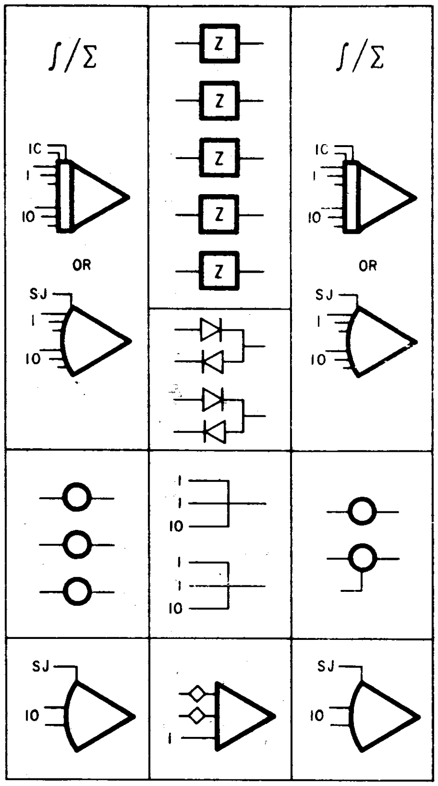
MODULE 17 C7.102



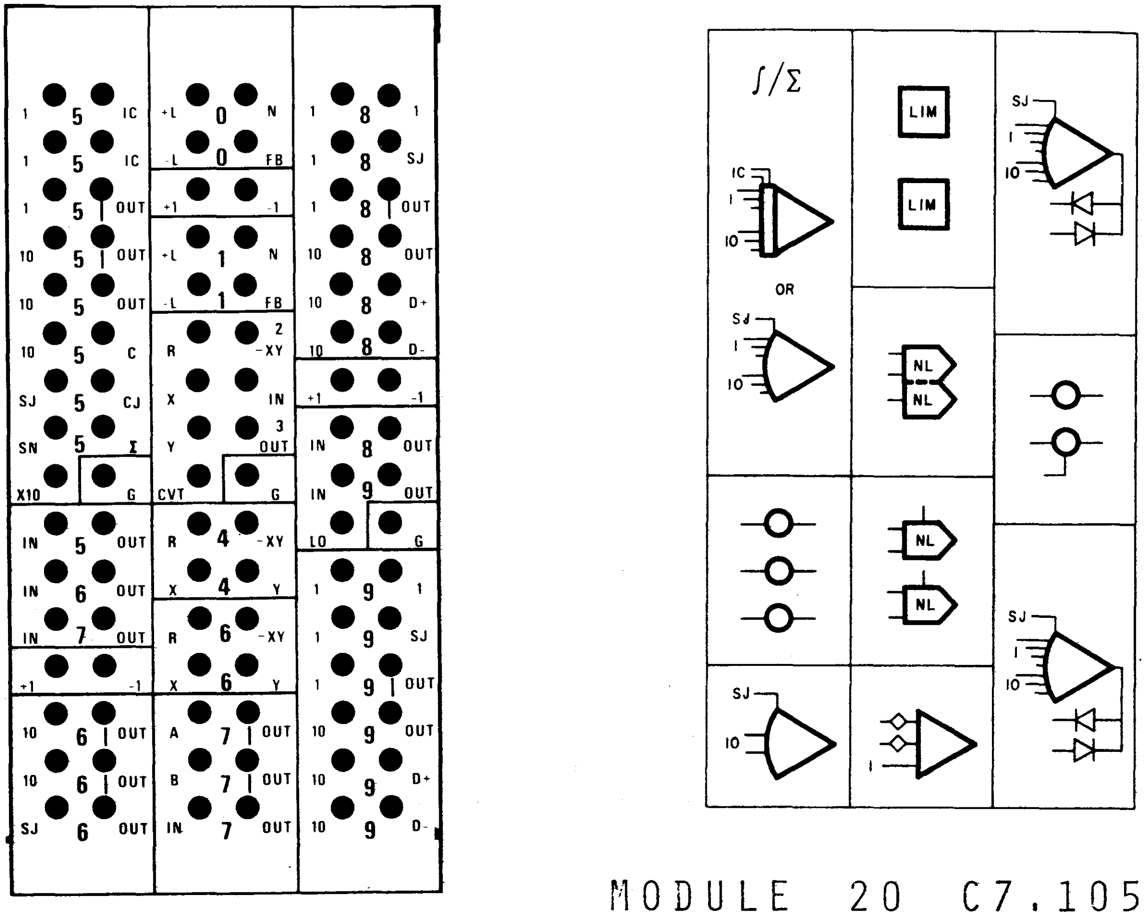


MODULE 18 C7,103

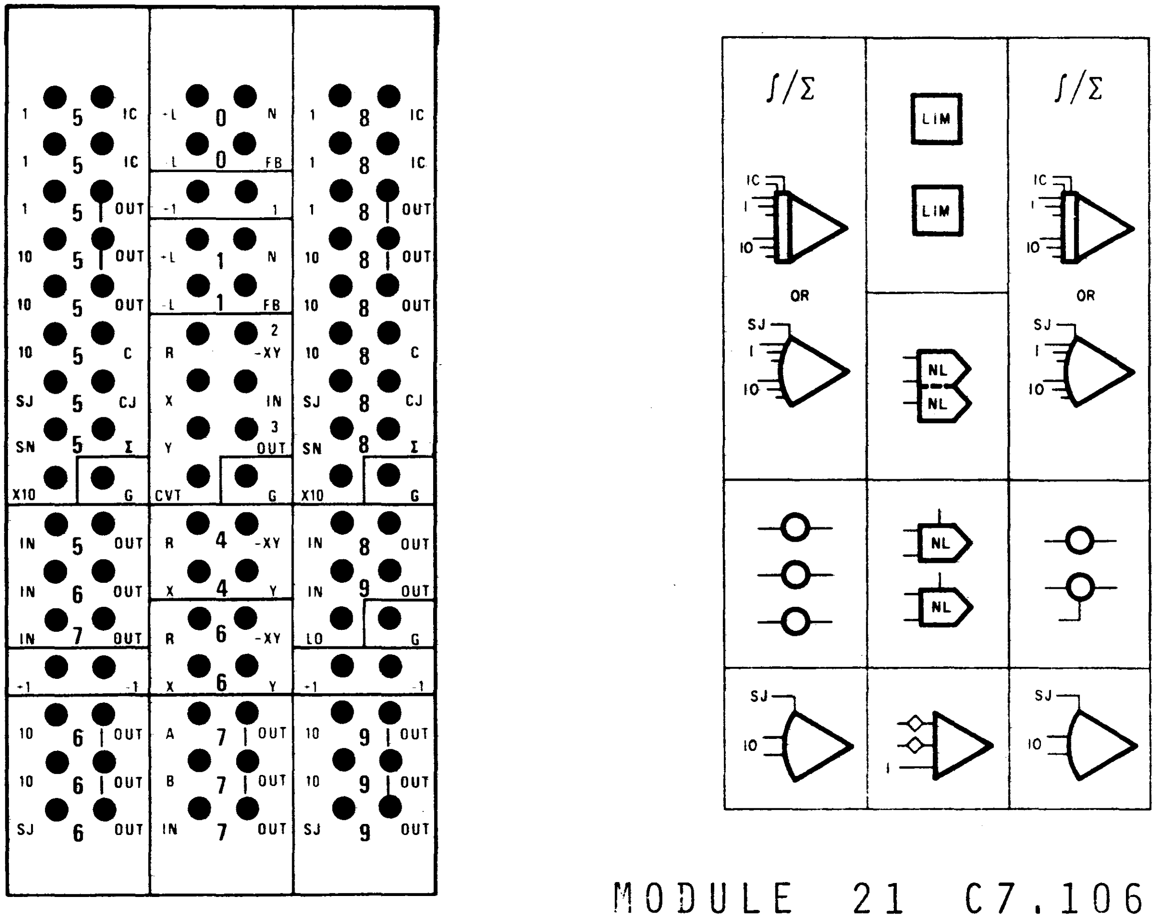




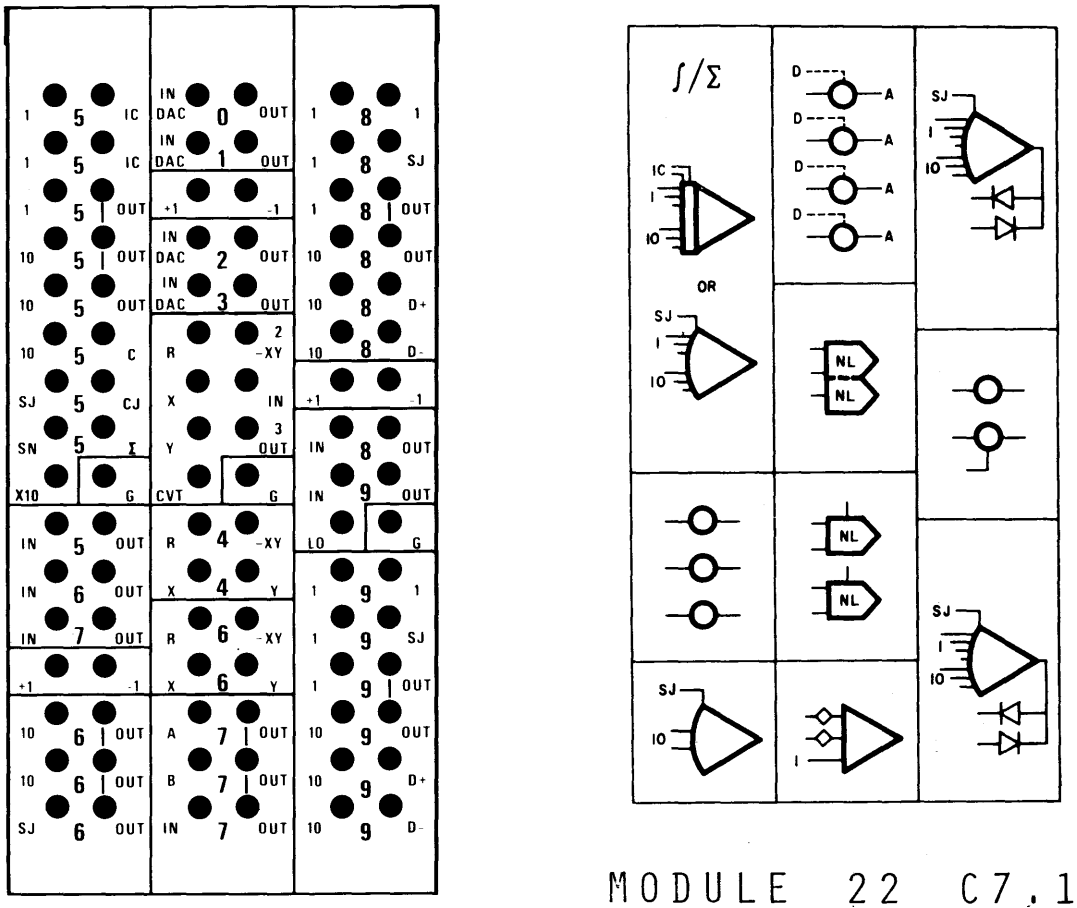
MODULE 19 C7.104



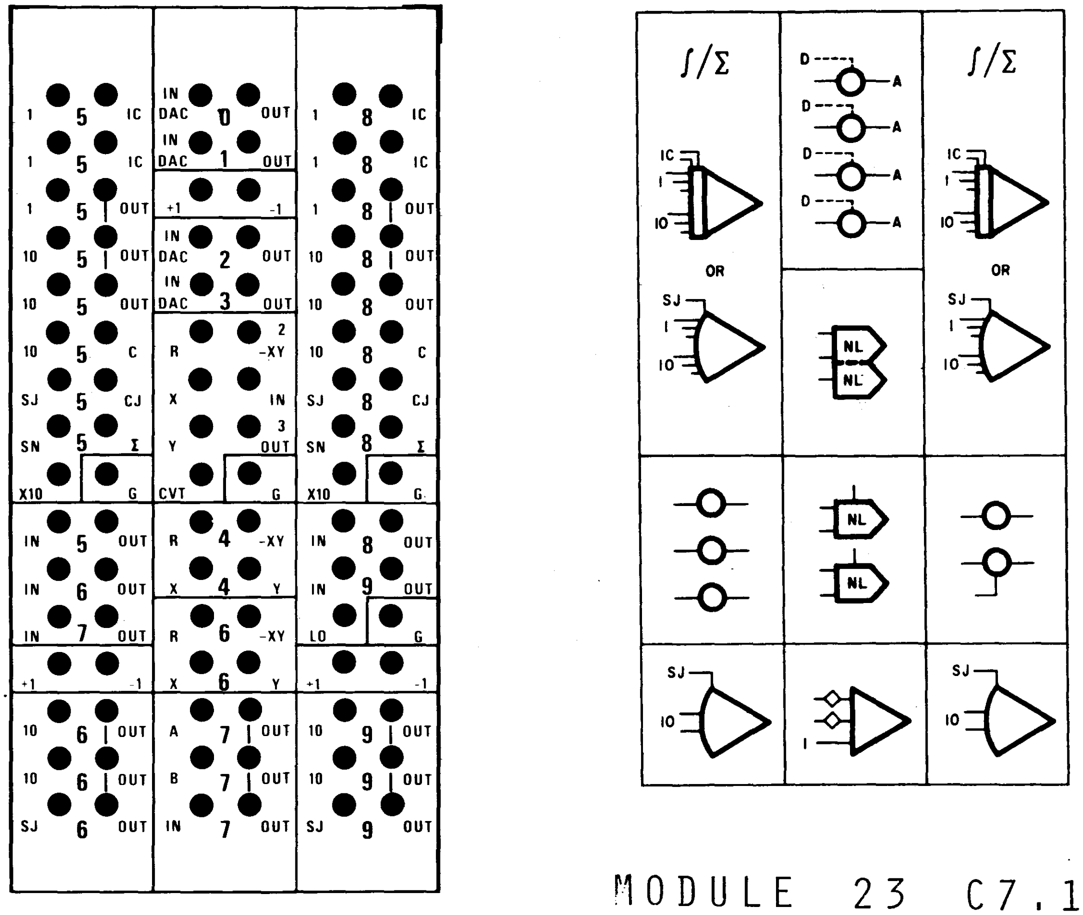
MODULE 20 C7.105



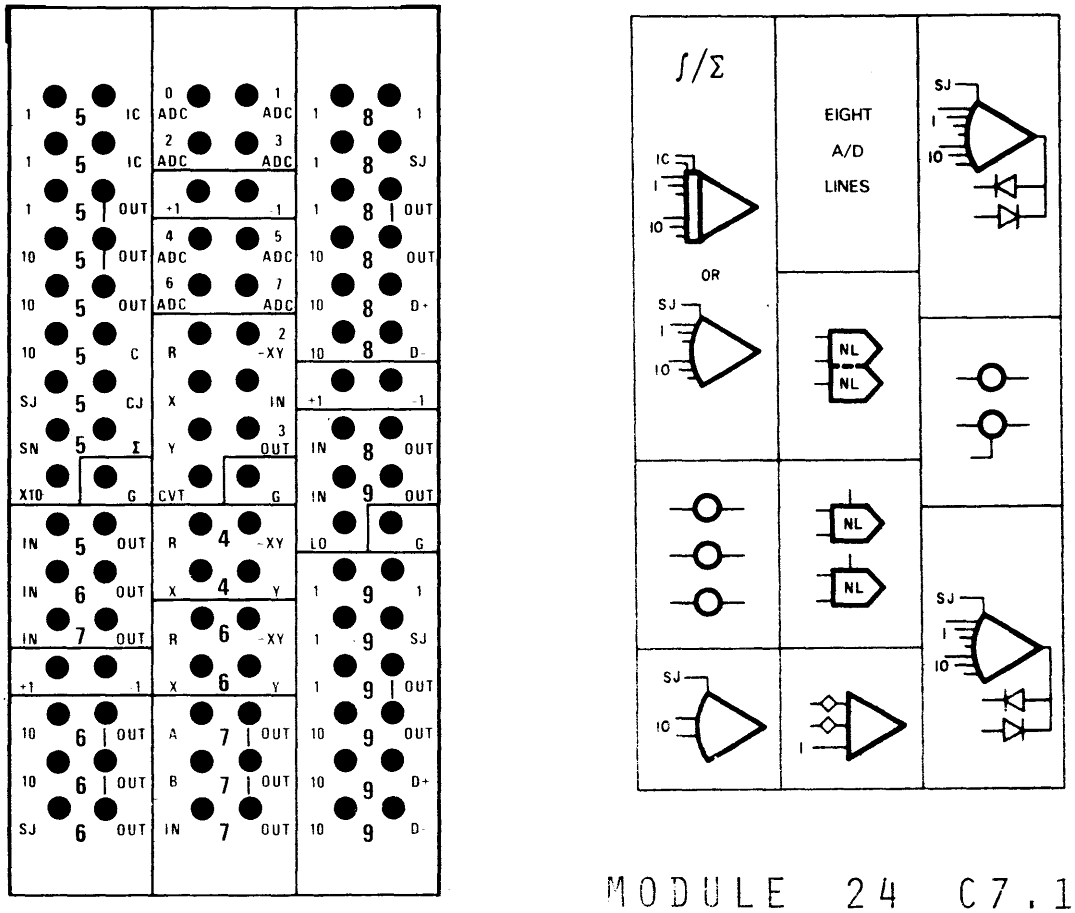
MODULE 21 C7.106



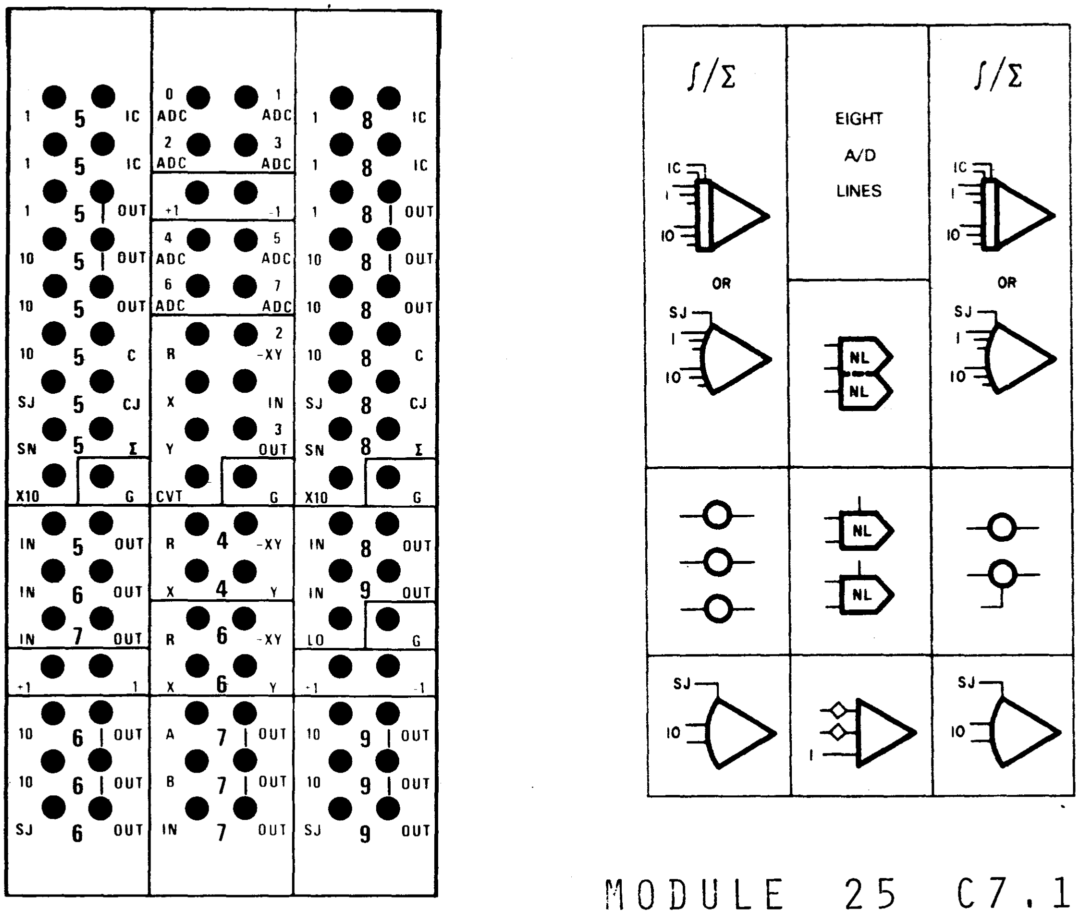
MODULE 22 C7.107



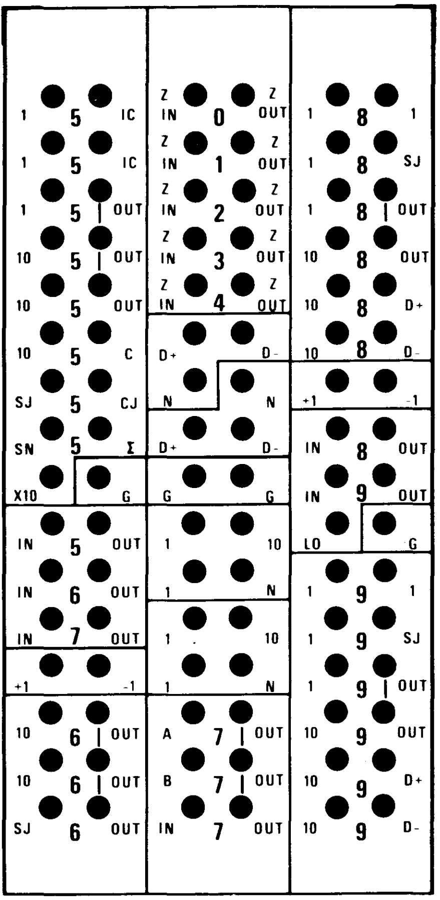
MODULE 23 C7.108

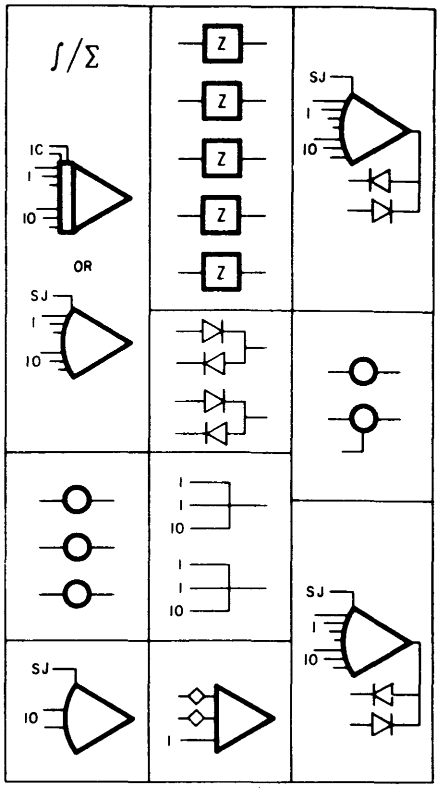


MODULE 24 C7.109

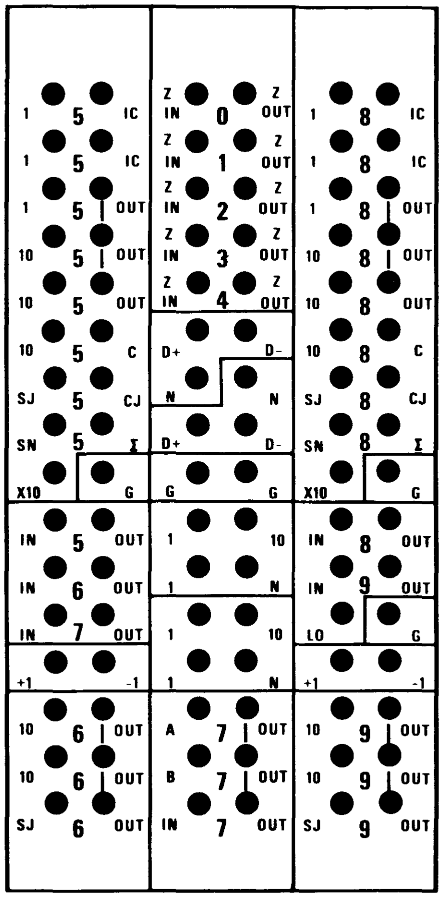


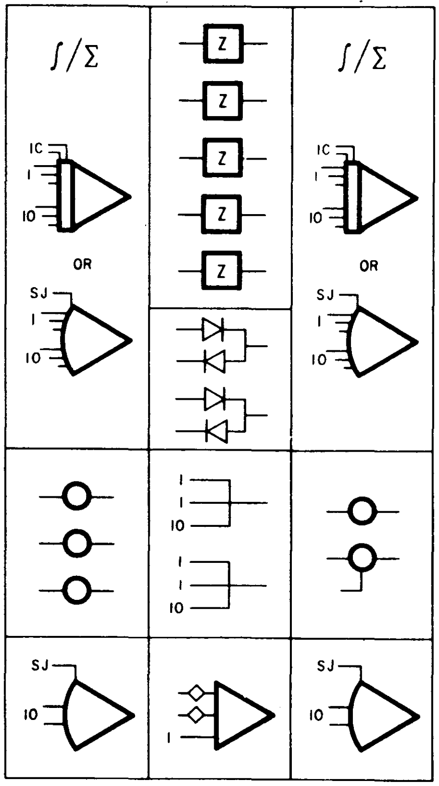
MODULE 25 C7.110



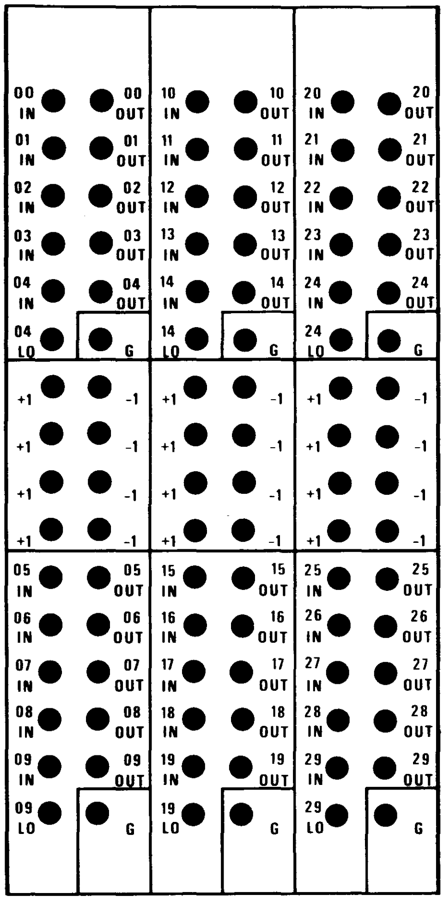


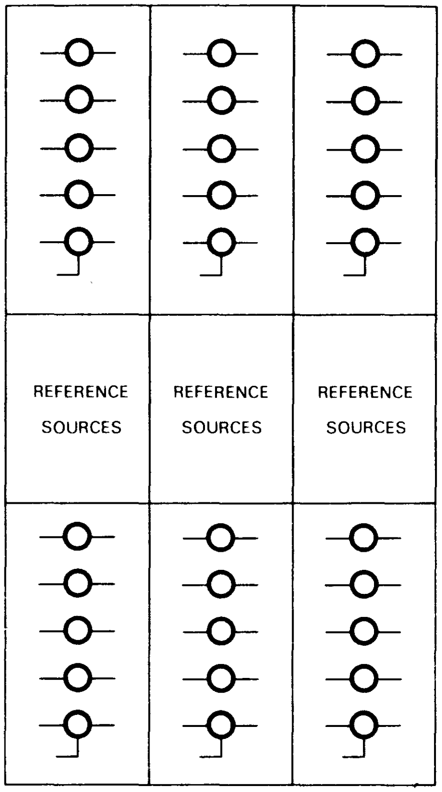
MODULE 26 C7.111





MODULE 27 C7.112





MODULE 28 C7.113

CHAPTER 5

APPLICATIONS

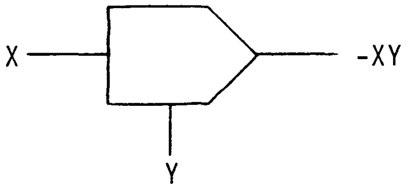
TABLE OF CONTENTS

ILLUSTRATIONS

5.0 INTRODUCTION

Chapter 5 presents a collection of useful circuits which can be used with the computing components which have been described in Chapter 4. The actual patching details which are described in Chapter 4 will not be repeated here; only circuit diagrams will be given in Chapter 5, along with brief descriptions of the purpose of the circuit or technique illustrated. Unity reference scaling is used throughout this chapter (i.e., +1 and -1 are equivalent to +reference and -reference voltages respectively). If no enable signal is indicated for a logic device, then that device is assumed to be enabled by a logic one.

Labelling conventions are described below. If the inputs to a multiplier are X and Y and the output is -XY, then the circuit will be the following:



NOTE: —⊙— signifies a patchboard hole throughout this chapter, and is labelled in the same way as the patchboard.

Amplifier values in the AD/FIVE are read by a digital ratiometer. For the circuit above, if the source of the signal X is addressed and read as 0.6000, and if the source of Y is addressed and read as 0.7000, then the multiplier when addressed will yield a reading of -0.4200.

5.1 CIRCUITS FEATURING ANALOG COMPUTING COMPONENTS

This section describes useful circuits utilizing the analog computing elements in a primary role.

5.1.1 Integrator Circuits

This section describes circuits featuring the integrator.

5.1.1.1 Use of the Integrator as a Switch Amplifier

The integrator in the AD/FIVE may be used as a two-switch multiple input switch amplifier by omitting the patching between C and CJ (this patching is usually performed as shown in Section 4.2.1.1). To use the integrator as a switch amplifier, patch the output to the input as shown in Section 4.2.1.1; do not patch C to CJ. Upon exercising mode control, the output of the integrator will be:

HOLD	output frozen at last value
IC	sum of the IC inputs
OPERATE	sum of the integrand inputs

The integrator is not intended for this use, and the output will spike on switching, but the circuit is valuable if it is necessary to augment the electronic switch complement for such uses as display channel multiplexing, where the display is blanked during the switch spike.

5.1.1.2 Triangle Generator

The triangle waveform generator is a useful circuit for providing a forcing function for function generator setup and checkout. This simple circuit is superior in many applications to the sine wave generator for two important reasons: it is amplitude-stable (the output does not decay with time), and the slope of the output has a constant magnitude for all amplitudes of the function. When the triangle wave generator is used to assess the hysteresis error in an arbitrary function generator, all breakpoints will be swept at the same rate. When a sine wave input is used, however, breakpoints at large input values are swept very slowly, and this can cause erroneous conclusions about high speed performance.

Pot 1 adjusts slope. Pot 2 adjusts amplitude. Changing the amplitude of the output waveform also changes the frequency of the output. For a fixed amplitude (Pot 2 setting) Pot 1 adjusts the frequency.

Output 2 provides the triangle waveform. Output one provides another useful wave form, the square wave.

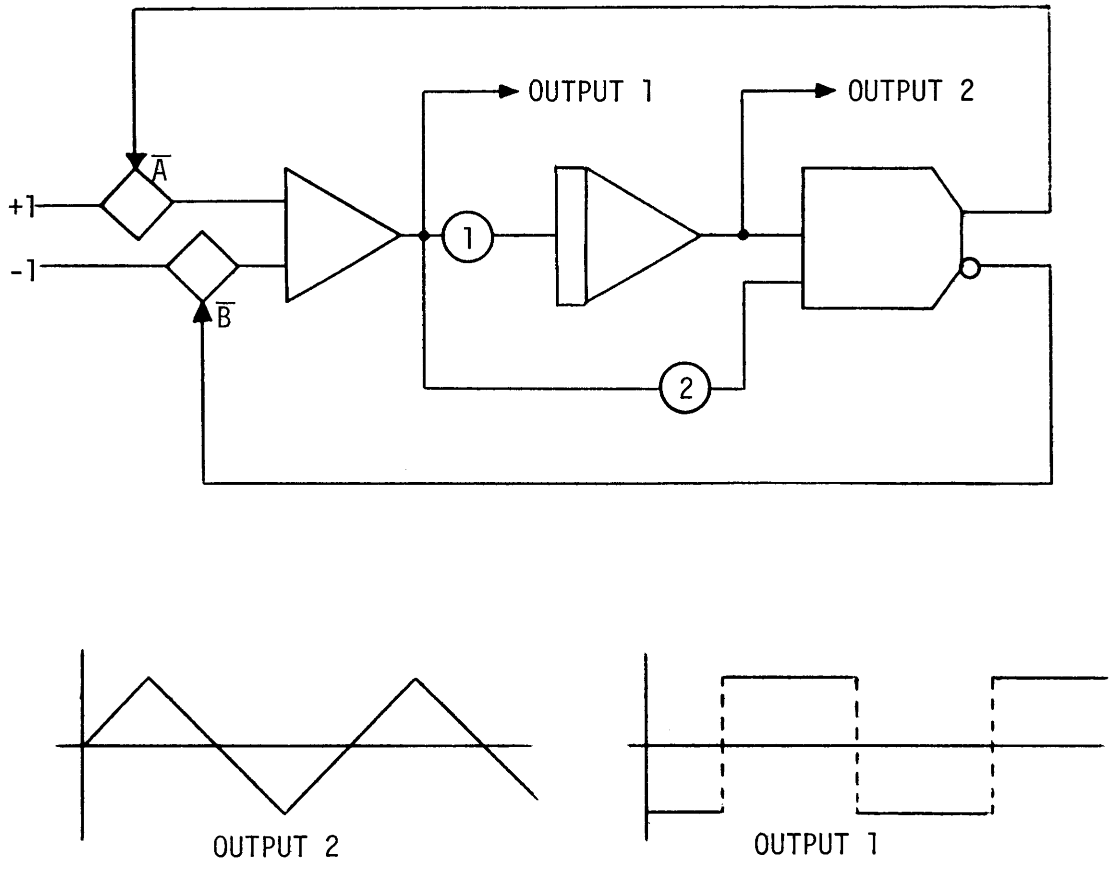


Figure 5-1 Triangle Wave Generator

5.1.1.3 Track Store

The integrator can be used to perform the role of a track-store device. The patching is shown in Figure 5-2. The track input is patched to the normal IC input of the integrator, and the desired IC is patched to a gain 10 input hole. In the IC mode the amplifier output will go exponentially to the IC value. 1 millisecond should be allowed for the output to reach within 0.1% of the desired IC when the patching is as shown in Figure 5-2.

MODE	H	OP
IC	0	1
TRACK	0	0
STORE	1	1 or 0

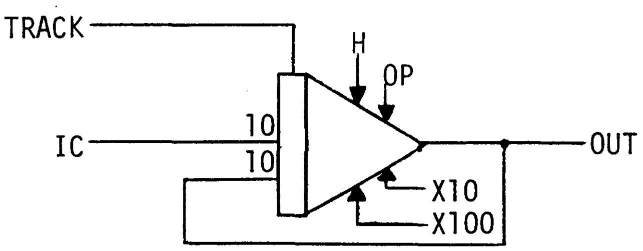
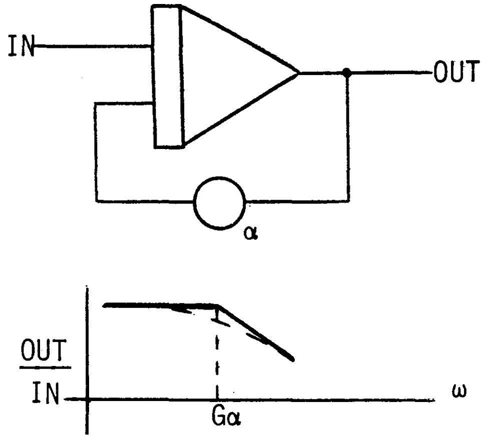


Figure 5-2 Using the Integrator as a Track Store

5.1.1.4 Low Pass Filter

A first order low pass filter can be programmed using an integrator with a pot in the feedback. For a gain 1 integrator the corner frequency is determined directly by the pot setting α, and for other gains the roll off point is determined by the product of the integrator gain G and the pot setting α (Gα).

TRANSFER FUNCTION: $\frac{-1}{S + G\alpha}$



The corner frequency (6db/octave roll off point is α radians/second.

Figure 5-3 The Low Pass Filter

5.1.1.5 Algebraic Equation Stabilizer

Since an analog computer summer is not perfect it can cause difficulty in algebraic problems. In particular, if at some frequency a closed algebraic loop has 360° of phase shift and a gain of unity, then the Barkhausen Criterion for Oscillation is satisfied, and the circuit will oscillate at that frequency.

There are two troublesome conditions to look for in analog circuits for algebraic problems. One is a closed loop consisting of an even number of inverting summer amplifiers.

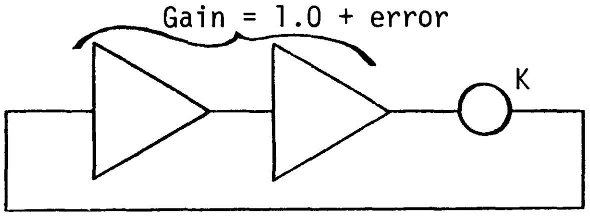


Figure 5-4 Algebraic Loop with Even Number of Summers

Clearly in Figure 5-4, K must be less than 1.0 to avoid high frequency oscillation, since an even number of inverting summers is equivalent to a phase shift of 360°. This is true if perfect unity gain summers are assumed. However, the summers can contribute enough extra gain error at some high frequency to give a net loop gain greater than 1.0, and the circuit will oscillate at that frequency.

The other condition occurs when a closed algebraic loop consists of an odd number of inverting summer amplifiers.

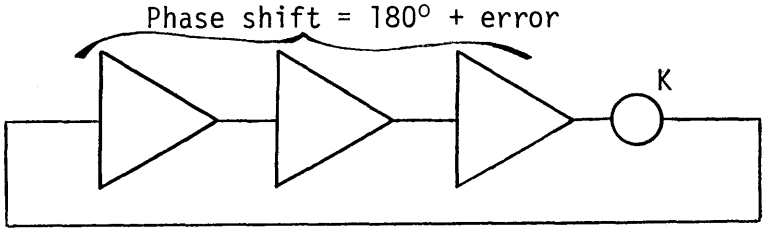
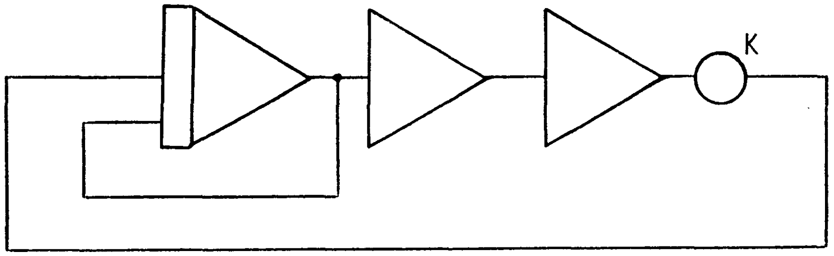


Figure 5-5 Algebraic Loop with Odd Number of Summers

In this case the loop gain is K and the phase shift is 180°, assuming ideal inverting summers. According to the oscillation criterion mentioned above, this loop should be stable regardless of the value of K. However, at very high frequencies a summer may introduce additional phase shift besides its usual 180° sign inversion. Even though the phase shift error for an individual summer is small, if the loop has a large number of amplifiers in series and the sum of the phase errors of the amplifiers at some high frequency produces a net 360° phase shift around the loop, then the circuit will oscillate at that frequency.

Both oscillation problems can be eliminated by substituting a low pass filter for one of the summers in the loop. Such a filter can be made by using an integrator which has the same DC gain feedback resistor as the previous summer.

This is illustrated in Figure 5-6. The integrator time constant should be chosen so that the oscillation frequency is suppressed and desired problem frequencies are not affected

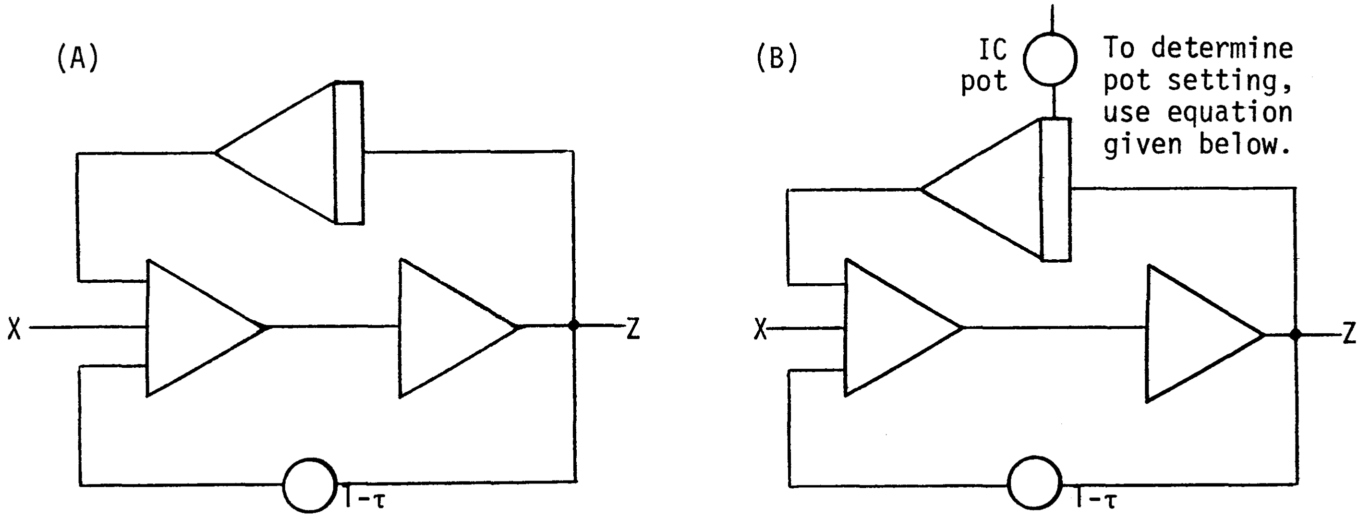


The feedback resistor for the integrator must be the same resistor as in the summer which it replaces.

Figure 5-6 Substituting a Low Pass Filter for a Summer

5.1.1.6 Differentiator

Differentiation is not recommended unless it is unavoidable. Avoidance techniques include implicit function generation, dummy variables, and careful selection of necessary variables to be explicitly formed. The circuit given in this section is easily patched and is the recommended approach to differentiation when this is necessary. Set the pot to the largest value which will not cause output oscillations.



To determine pot setting, use equation given below.

$$Z = \frac{dx}{dt} - \tau\frac{dZ}{dt}$$

It can be seen that setting the feedback pot as high as possible without causing oscillation will result in making τ small enough that the expression $\tau\frac{dZ}{dt}$ will become neglible, resulting in a very close approximation of dx/dt.

Figure 5-7 Differentiator Circuits

The circuit shown in Figure 5-7(A) is one that is usually recommended by most authors in discussions of differentiator circuits. Figure 5-7(B) shows an improved circuit in which the initial value of the derivative is established. This produces a much better result. In this example X = sin t. Then dx/dt = cos t. At t = 0, dx/dt = 1. The IC pot setting is determined by solving the equation:

$$\frac{dx}{dt} = (1-\tau)\frac{dx}{dt} + x + IC$$

or: $$IC = \frac{dx}{dt} - (1-\tau)\frac{dx}{dt} - x$$

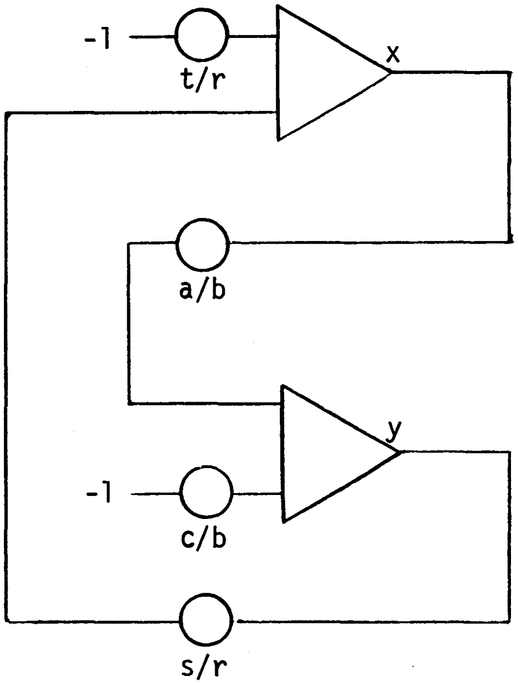
which reduces to: $IC = -\tau\frac{dx}{dt} -x \quad (\text{at } t = 0)$

5.1.2 Summer Circuits

The following sections feature the summing amplifier. When maximum high frequency performance is desired from the AD/FIVE, use gain 10 feedback (10K) and alter input gains accordingly. For example, if a gain 1 summer is programmed using gain 10 inputs and gain 10 feedback, this will give a considerable improvement in usable summer bandwidth, as opposed to the use of gain 1 inputs and gain 1 feedback.

5.1.2.1 Solution of Algebraic Equations.

Agebraic equations can be solved on an analog computer with somewhat more effort than differential equations. Although the examples which follow are trivial, they illustrate the method used when algebraic sub-problems are implemented on the analog computer. This occurs, for example, when differential equations are coupled in the highest derivative, such as mutual mass or mutual inductance. Figure 5-8 illustrates the basic method.



PROBLEM: $ax + by = c$

$rx + sy = t$

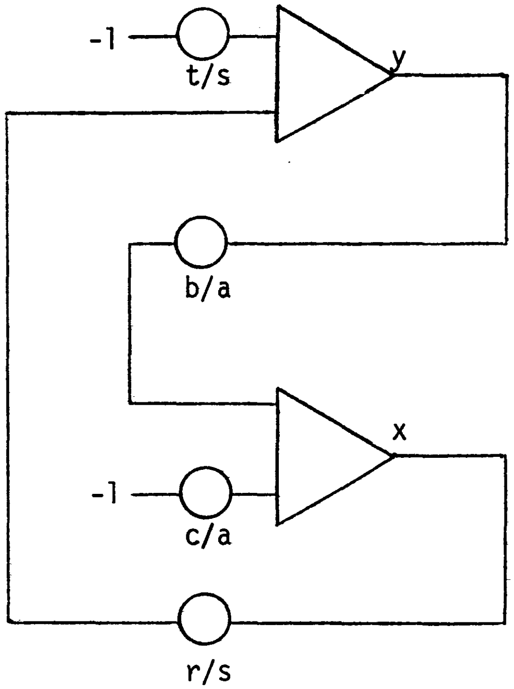
ISOLATE: $-x = \frac{a}{b} y - \frac{c}{b}$

$-y = \frac{s}{r} x - \frac{t}{r}$

Scaling is almost always required. See Figure 5-10 for a specific example.

Figure 5-8 Algebraic Equation Circuit

Note that the loop gain is (a/b) · (s/r). This loop gain must be less than one for stability in the case where the algebraic signs are such that an even number of amplifiers is in a loop (i.e., where the net positive loop gain is equivalent to 360° phase shift). If (a/b) · (s/r) is greater than one, the circuit will oscillate at some high frequency. If this is so, then the equations have been solved for the wrong variables. To obtain a solution, solve the first equation for y and the second equation for x. This is shown in Figure 5-9. The loop gain is now (b/a) · (r/s), which is the reciprocal of the loop gain in the previous example, and thus less than one. Such intelligent manipulation of equations becomes essential in higher order cases.



ISOLATE:

$$-y = \frac{r}{s}x - \frac{t}{s}$$

$$-x = \frac{b}{a}y - \frac{c}{a}$$

Figure 5-9 Stable Algebraic Solution

If a problem has algebraic signs such that the loop has an odd number of amplifiers, then a loop gain greater than one is feasible. However, instabilities at high frequencies may result due to summer phase shift error. Refer to section 5.1.1.5 for a discussion of the use of the integrator as an algebraic equation stabilizer to solve the problem of instability in algebraic loops which cannot be rescaled.

Figure 5-10 gives a specific example of the solution of algebraic equations.

5.1.2.2 Arbitrary Transfer Characteristic

The use of the summing amplifier to obtain an arbitrary transfer characteristic is often overlooked. A few simple characteristics may be summed to produce a desired characteristic. The outputs of circuits which produce simple characteristics are patched to the inputs of a summer, which produces a desired arbitrary transfer characteristic. Each of the constituent circuits is driven with the same input as shown in Figure 5-11.

PROBLEM: $2x + 3y = 27.75$

$3x + y = 16$

ISOLATE: $-y = \frac{2}{3}x - \frac{27.75}{3}$

$-x = \frac{1}{3}y - \frac{16}{3}$

REDUCE: $-y = 0.6667x - 9.2500$

$-x = 0.3333y - 5.3333$

SCALE EQUATIONS: $\frac{-y}{10} = 0.6667\,\frac{x}{10} - 0.9250$

$\frac{-x}{10} = 0.3333\,\frac{y}{10} - 0.5333$

Scaling allows normalized x and y values in the range:

$-10 \leq x \leq +10$

$-10 \leq y \leq +10$

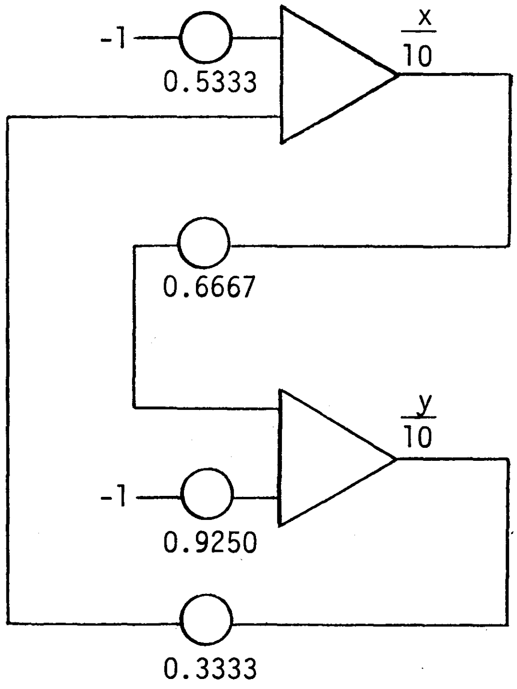


Figure 5-10 Example of Algebraic Solution

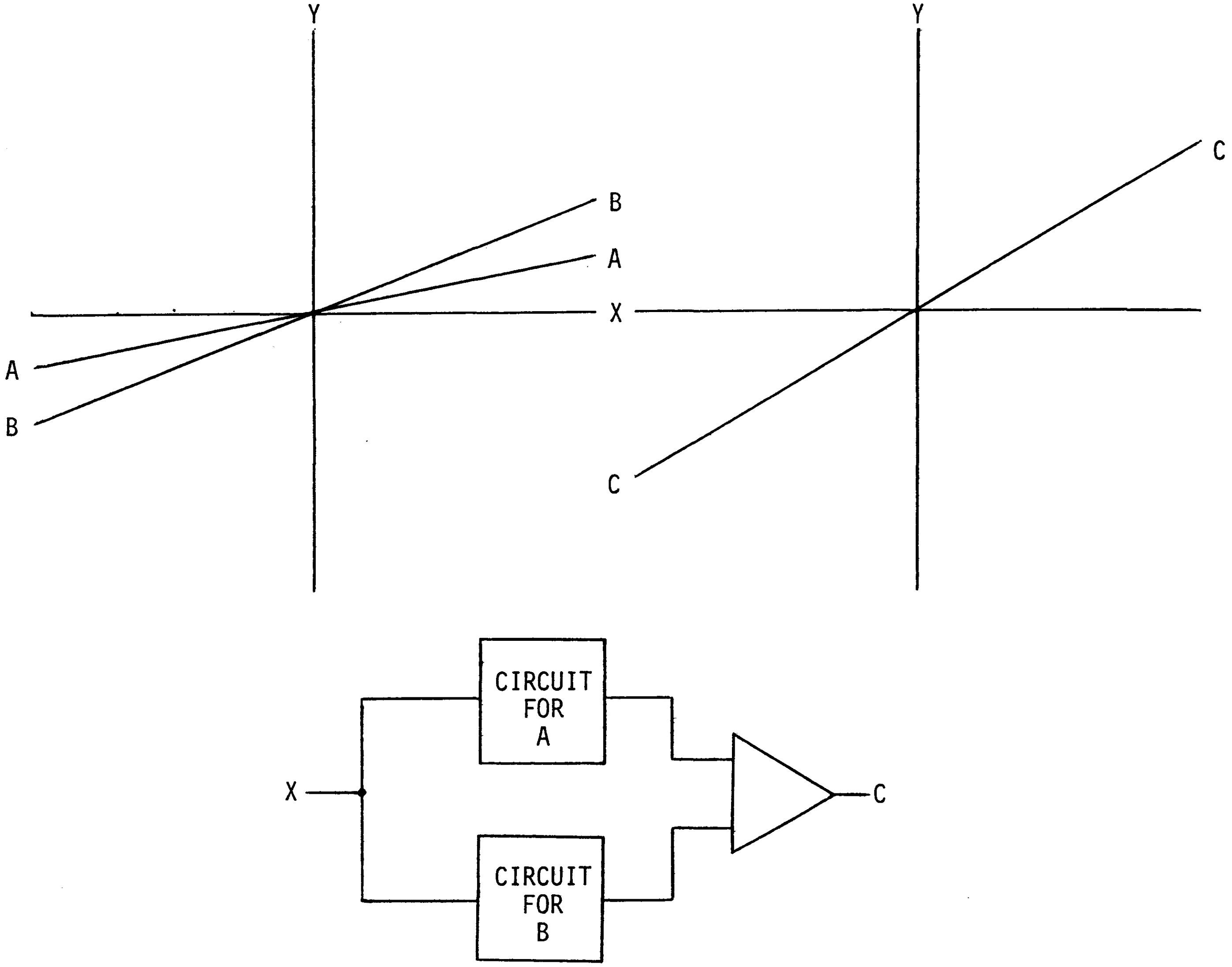


Figure 5-11 Sum of Two Transfer Characteristics

In Figure 5-11, the circuits for A and B are summed together, producing as a result the graph of C.

In Figure 5-12, the Dead Space Characteristic Z is desired. Graphical analysis of Z reveals that Z is the sum of characteristics A and B. B is the characteristic of a limited inverter. On the analog computer, Z can be obtained by adding with a summer the outputs of the circuits which produce characteristics A and B. The programmer will develop an intuition which will lead him to facility in using the summer to obtain desired arbitrary transfer characteristics. Practice in this technique yields a remarkable flexibility in programming.

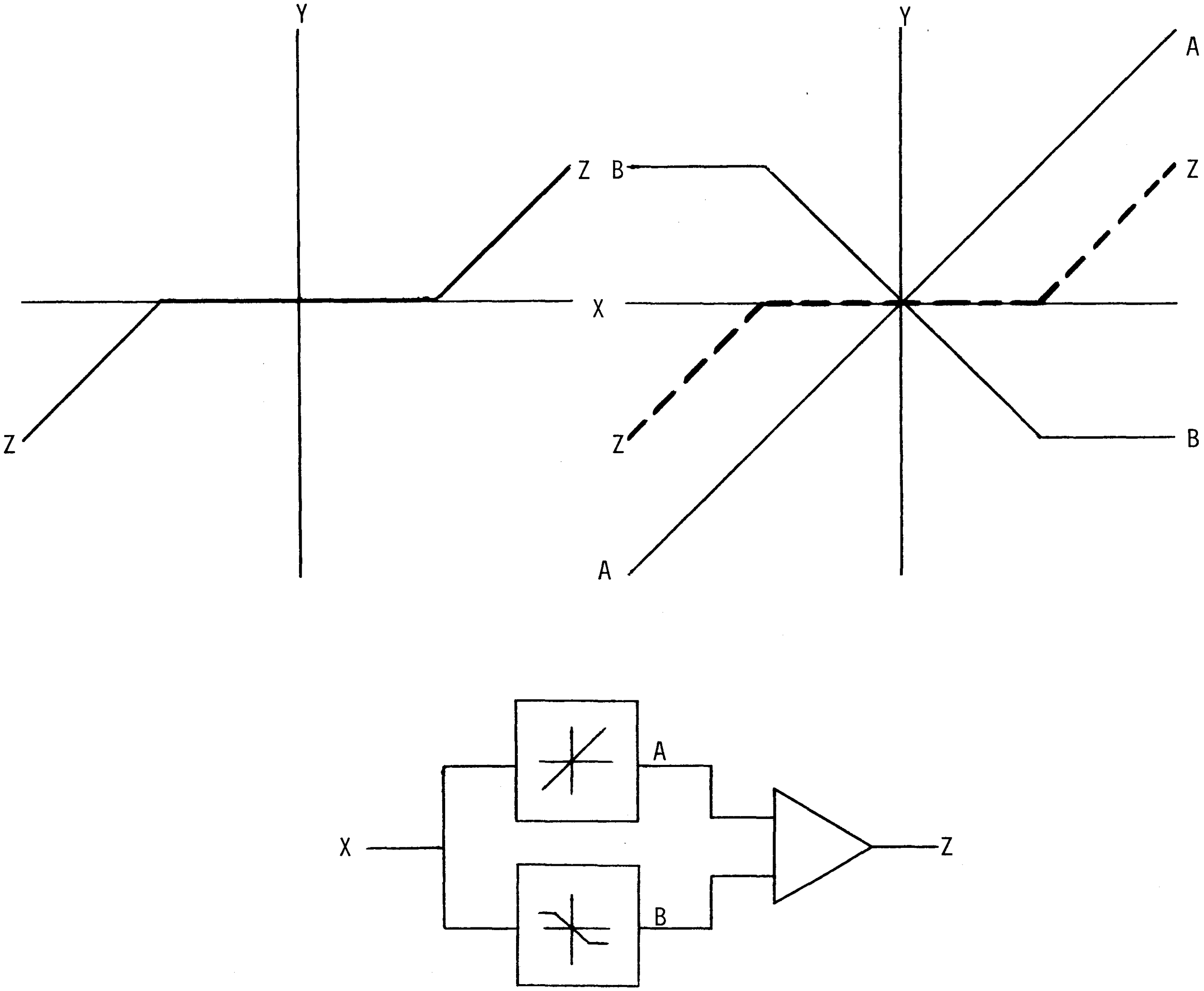


Figure 5-12 Summing for the Dead Space Characteristic

Figure 5-13 depicts two different methods of achieving a horizontal shift of the characteristic. In Figure 5-13(A) the desired characteristic is shown. This is the same as the dead space characteristic of Figure 5-12, but shifted to the left by 0.3, i.e., $Z_1 = Z(X + 0.3)$. Figure 5-13(B) shows how to do this by shifting the input to obtain X + 0.3 and then providing that value to the Z_1 circuit. The circuit in Figure 5-13(B) is the same as that in Figure 5-12; only the input is changed. However, if the input X is greater than ΔX (0.7 in this example), then the input summer will overload. As a general technique this method is not preferred because of the potential for overload difficulties. Instead, the programmer should seek a way of deriving the desired characteristic by using only characteristics of known circuits.

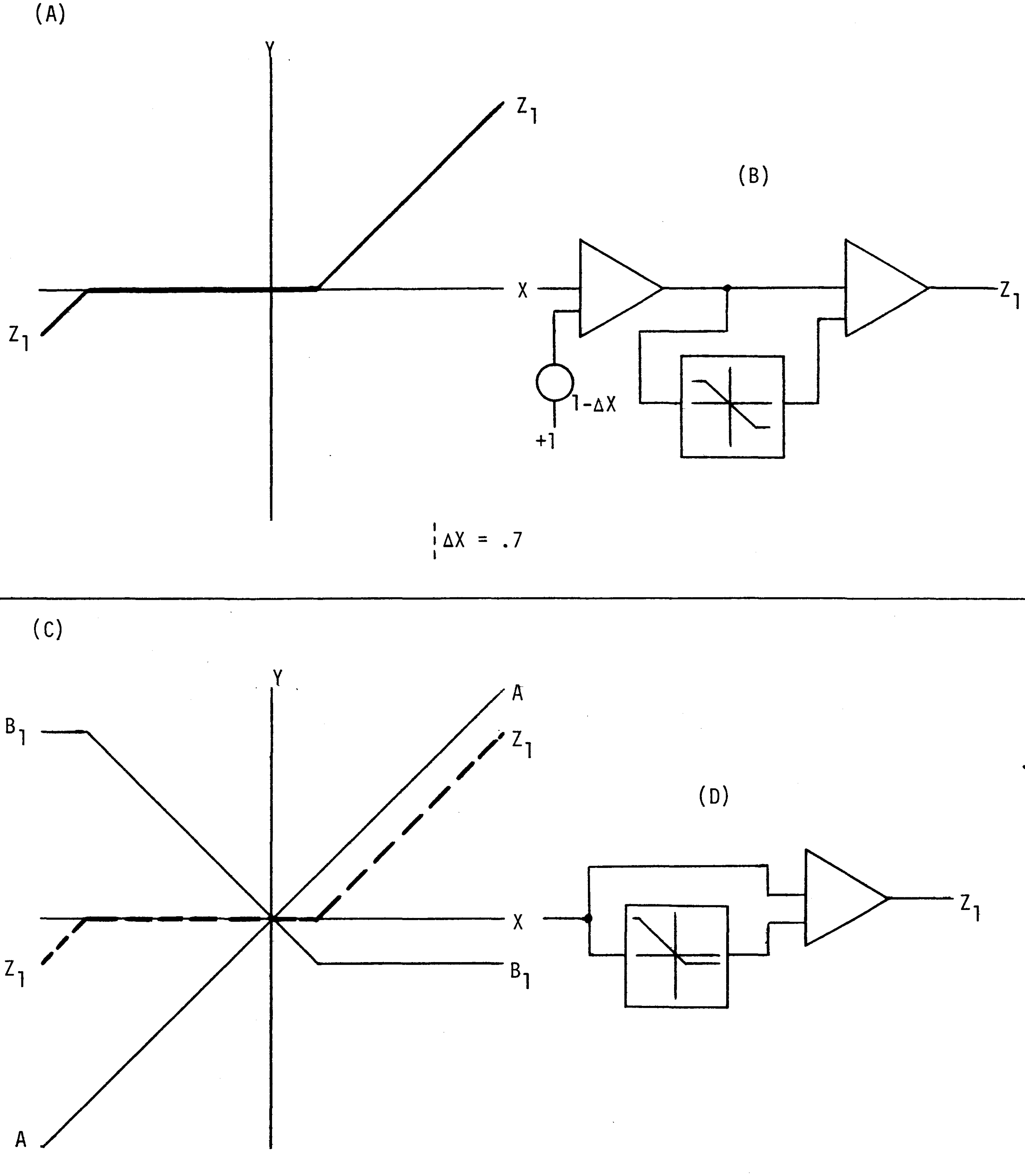


Figure 5-13 Horizontal Shift of a Characteristic

Figure 5-13(C) shows a graphical analysis of the characteristics which will sum to produce the desired characteristic. A comparison of Figure 5-13(C) with Figure 5-12 shows that the only difference is that B_1 is substituted for B. B_1 can be generated using a circuit with limit values different from those of the circuit for B. Once this is done properly, the circuit shown in Figure 5-13(D) will produce the desired characteristic Z_1.

The programmer should strive to master the technique of programming known circuits to obtain desired characteristics by summing. The advantages of employing symmetry, absolute value circuits, limiters, etc. will enable the programmer to obtain desired characteristics quickly and reliably.

Scaling requirements should be mentioned at this point. The input to any function generator is the scaled value available from the rest of the program, such as an integrator output. The scaling for this term is accomplished independently of the fact that the term is to be used to generate some function. This usually fixes the value of the breakpoints. The output magnitude of the function should be chosen for the best circuit performance. Then the gain of the following circuit is adjusted for the correct value for the simulation. See Introduction to Analog Computation, Joseph J. Blum, New York, 1969 for an unusually simple but lucid treatment of this important concept, which is dealt with in Chapter 8 and especially Example 8.4.

Figure 5-14 illustrates an additional scaling concept. The desired characteristic is A, shown in Figure 5-14(A). Graphical analysis of A shows that it is composed of the sum of the characteristics B and C, shown in Figure 5-14(B). If C were formed explicitly, it would overload the amplifier at whose output it existed. The trick therefore is not to form C explicitly, but rather to form .1C, and then to perform the final summation using a gain 10 input as shown in the circuit of Figure 5-14(C). Note also that the limit value for X = -0.5 is achieved by limiting the final summing amplifier. By using this circuit the minimum number of summing amplifiers is used, and the programmer achieves the desired characteristic C with the greatest economy of means.

5.1.2.3 Diode Outputs: Magic Circuit and Absolute Value Circuit

There is a diode circuit, often referred to as the "Magic Circuit", which simulates an ideal diode under load, i.e., a half-wave rectifier. This useful circuit is easily patched on the AD/FIVE using the convenient output diodes on the summers, as shown in Figure 5-15. If the D+ and D- terminals are interchanged, the complimentary characteristic is obtained. Figure 5-15 shows the circuit diagram for the magic circuit enclosed by a dashed line; the entire circuit produces an absolute value function. The output of the magic circuit is obtained at the D+ output; the output of the absolute value circuit is obtained at the output of the final summer. The OUT patchhole of the summer is not used in these two circuits. Figure 5-15(A) shows the circuit patching; 5-15(B) shows the transfer characteristic for the magic circuit; 5-15(C) shows the transfer characteristic for the absolute value circuit. The circles represent patchboard holes and are labelled in the same manner as the patchboard itself. NOTE: Do not attempt to monitor the D+ output with the PB input and the DRM. When the output is zero, the DRM would be "floating".

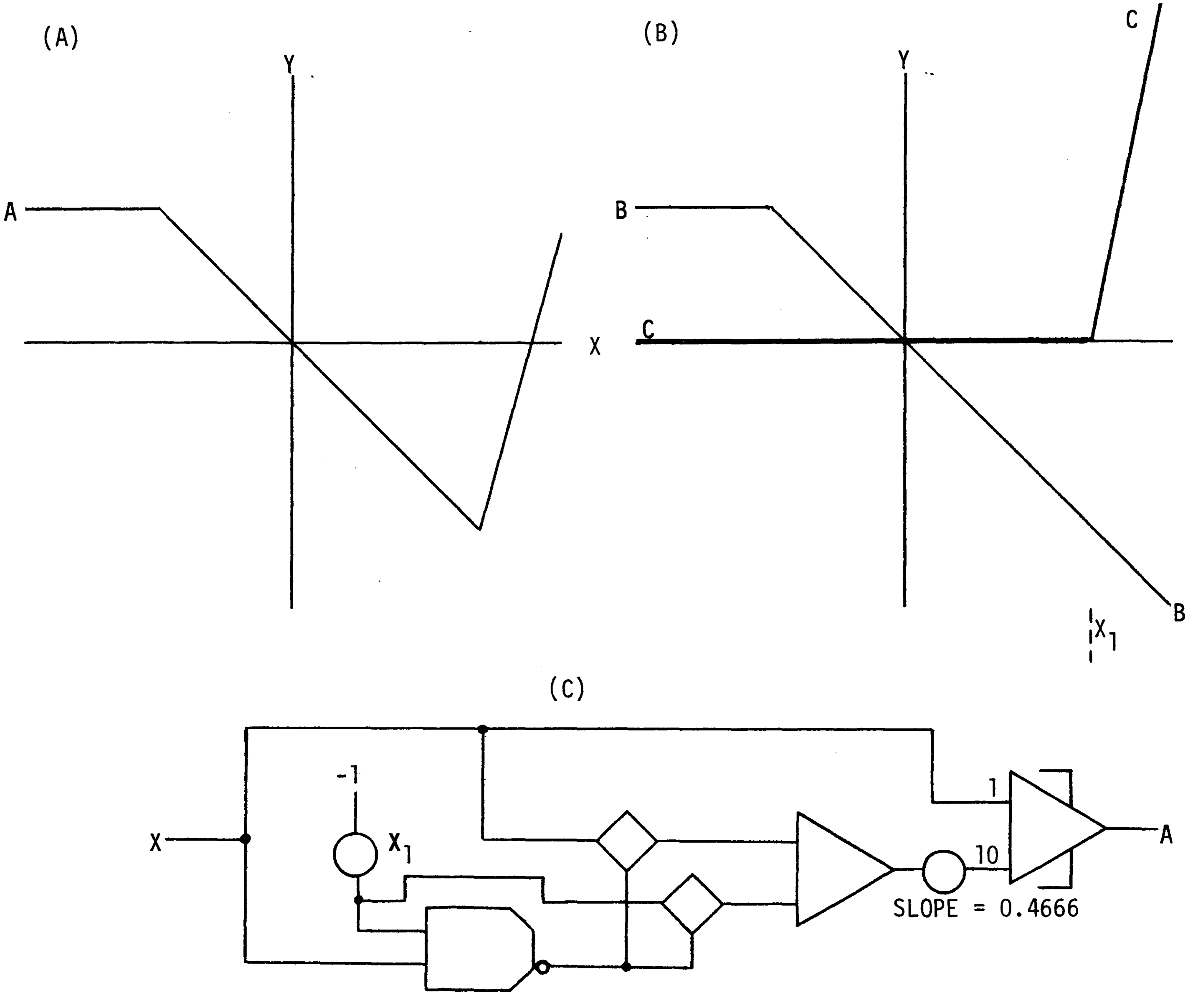


Figure 5-14 Example of an Arbitrary Characteristic

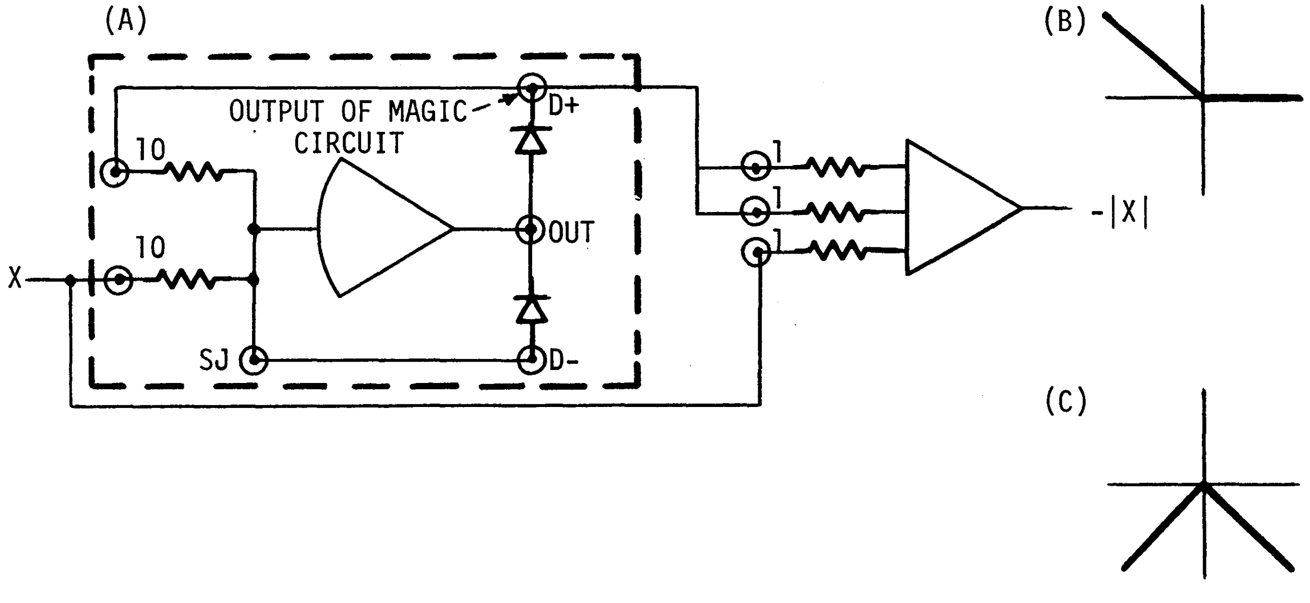
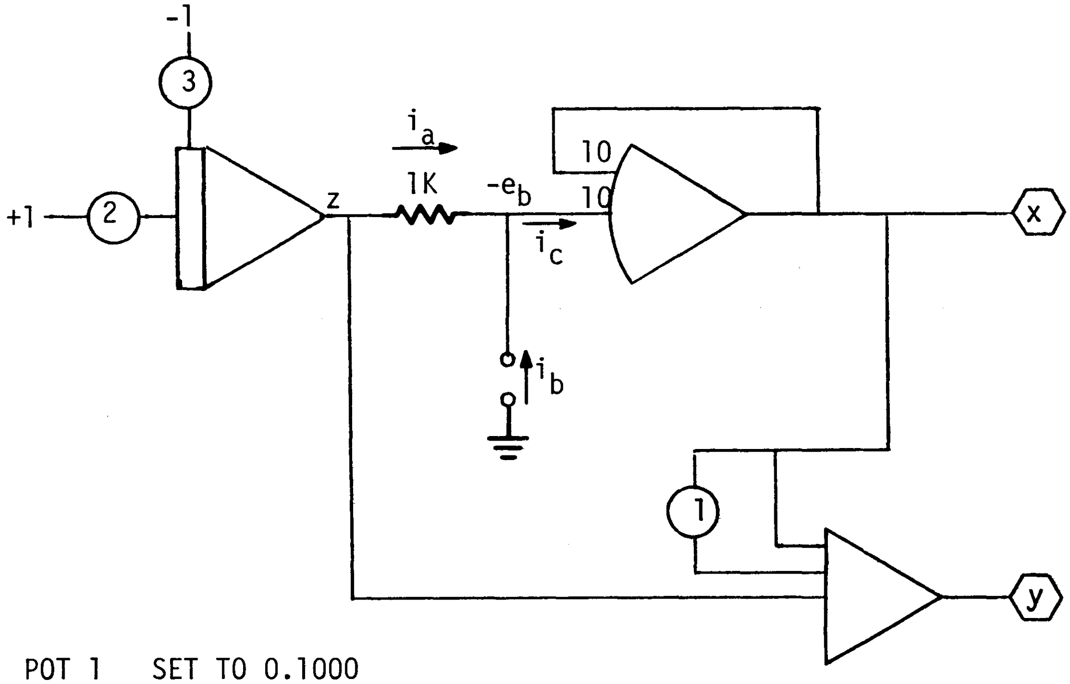


Figure 5-15 Magic Circuit and Absolute Value Circuit

5.1.2.4 Volt-Ampere Characteristic Plotter

In this section a very useful circuit is discussed. The circuit features summing amplifiers and allows the analog computer to plot the current through a device versus the voltage across it. Thus it is a self-impedance plotter and makes the computer a general test instrument which can replace certain very expensive, limited use test equipment such as transistor curve tracers. Figure 5-16 Part A illustrates the technique. The derivation is included in Figure 5-16 Part B. An intriguing feature of this circuit is the fact that it is self-limiting, both in voltage and in current applied to the device. Note that if the device is a short circuit (a diode) the maximum current would be 10 milliamperes, a safe value. If such a device is attached to the terminals, the voltage across the device will automatically adjust to the device. Characteristics of devices such as zener diodes are neatly plotted by this circuit.

A further refinement is easily implemented to allow the circuit to plot families of characteristic curves of three terminal devices such as transistors, FET's, SCR's, etc. Parts C and D of Figure 5-16 depict the method of producing a constant current source to provide base current to a transistor. The constant current circuit can be driven by a staircase generator, such as the one described in Section 5.1.3.1.



POT 1 SET TO 0.1000
POT 2 SETS SWEEP SPEED
POT 3 SETS EXCURSION LIMIT IN LEFT HALF PLANE ($e_b < 0$)

Figure 5-16 Part A Volt-Ampere Characteristic Plotter Circuit

Kirchoff Current Law at $-e_b$ node:

$$i_c = i_a + i_b$$

$$-\frac{e_b}{10^4} = \frac{z + e_b}{10^3} + i_b$$

or $i_b = -10^{-3}\,(z + 1.1e_b)$ ampere

Output to Trunk $x = e_b$ by definition

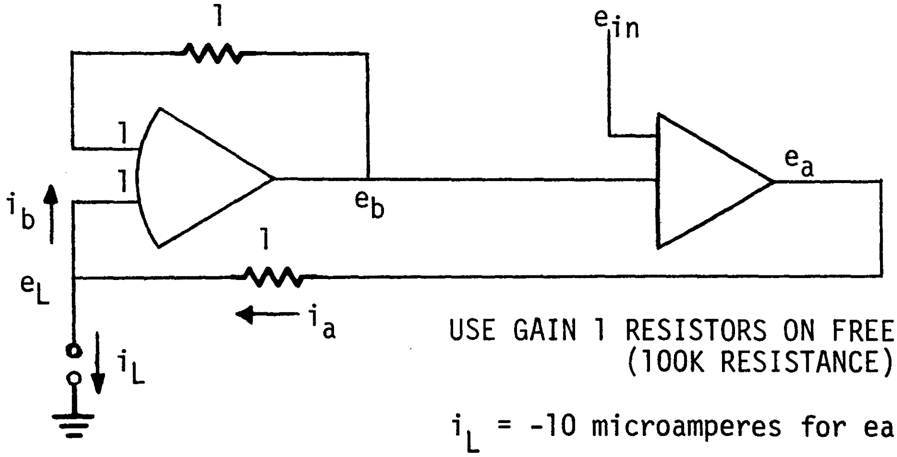
Output to Trunk $y = -(z + 1.1e_b)$ by definition

Therefore, on a 10 volt reference computer:

±1 at x represents ±10 volts at the terminals

±1 at y represents ±10 milliamperes at the terminals

Figure 5-16 Part B Volt-Ampere Characteristic Plotter Derivation



USE GAIN 1 RESISTORS ON FREE NETWORK CARD (100K RESISTANCE)

i_L = -10 microamperes for each volt of e_{in}

NOTE: THE TWO EXTERNAL RESISTORS MUST HAVE EQUAL RESISTANCE (R)

$$i_L = \frac{e_{in}}{R}$$

Figure 5-16 Part C Constant Current Source

$$i_a = i_L + i_b \qquad\qquad i_L = \frac{e_a}{10^5} - \frac{2e_L}{10^5}$$

$$\frac{e_a - e_L}{10^5} = i_L + \frac{e_L}{10^5} \qquad\qquad i_L = \frac{-e_b - e_{in}}{10^5} - \frac{2(-.5e_b)}{10^5}$$

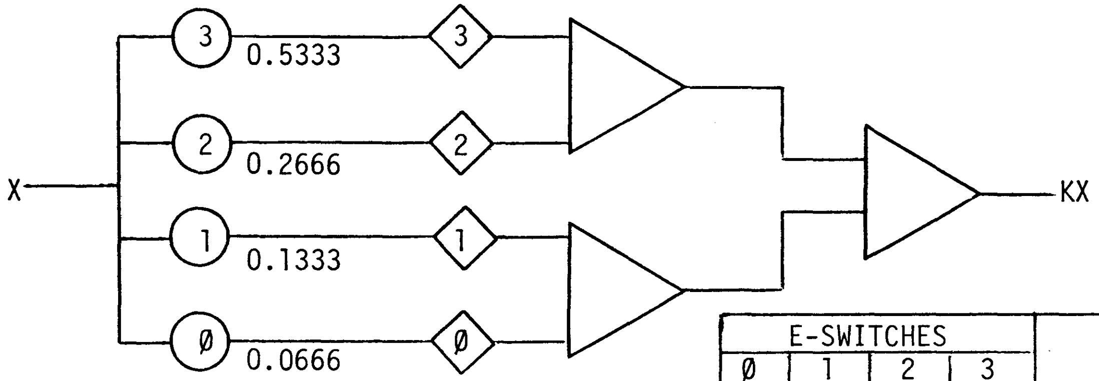
$$e_b = -2e_L \qquad\qquad i_L = -\frac{e_{in}}{10^5}$$

$$e_a = -e_b - e_{in}$$

Figure 5-16 Part D Derivation of Constant Current Source

5.1.3 Switch Summer Circuit: Parameter Sweeper

This parameter sweeper provides a staircase generator circuit when controlled by a binary counter. The pot values shown in Figure 5-17 divide the computer reference into sixteen equal parts. The same circuit is included in the Grid Reference Pattern Generator of Section 5.1.6, in which different binary weighted values are used in order to divide the computer reference into ten equal parts.



E-SWITCHES				K
Ø	1	2	3	
1	1	1	1	0.0000
0	1	1	1	0.0666
1	0	1	1	0.1333
0	0	1	1	0.1999
1	1	0	1	0.2666
0	1	0	1	0.3332
1	0	0	1	0.3999
0	0	0	1	0.4665
1	1	1	0	0.5333
0	1	1	0	0.5999
1	0	1	0	0.6666
0	0	1	0	0.7332
1	1	0	0	0.7999
0	1	0	0	0.8665
1	0	0	0	0.9332
0	0	0	0	0.9998

E-SWITCHES: 0 = NORMALLY CLOSED
1 = OPEN

In the AD/FIVE the electronic switch is always closed if its control input is unpatched (logic zero). A logic one on the control input will open the switch.

A sequential binary switch pattern produces 16 equally spaced values of K as shown in the truth table at the right. Other combinations of binary weighted pot values can be used. The E-switches are controlled by the outputs of a four-bit binary counter (Section 5.2.5.2).

Figure 5-17 Parameter Sweeper

5.1.4 Inverter Circuits

The AD/FIVE Inverter is configured with two gain 10 input resistors. The most common configuration of this device is obtained by patching one input to the output. The other input is then a unity-gain inverter input. Since this amplifier has the fewest inputs, it is the logical one to use to obtain a high-gain amplifier.

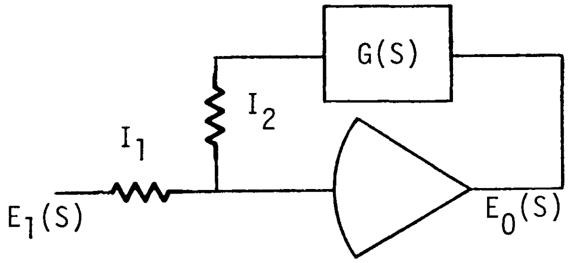
5.1.4.1 High Gain

Elementary analog computation texts usually present the proof that any mathematical operation which is performed in the feedback path of a high-gain operational amplifier results in the inverse mathematical operation in the forward path. Several examples of this are listed in Figure 5-18.

$$I_1(S) = -I_2(S)$$

$$E_1(S) = -G(S)E_0(S)$$

$$\frac{E_0(S)}{E_1(S)} = \frac{1}{G(S)}$$



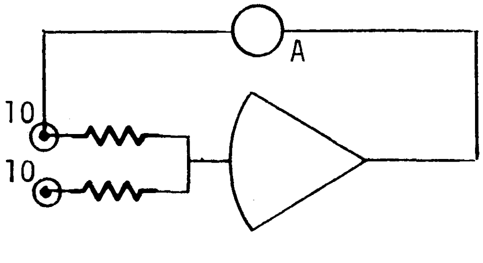
EXAMPLES

$G(S)$ = GAIN K........................$1/G(S)$ = GAIN 1/K
$G(S)$ = MULTIPLICATION BY E_2..........$1/G(S)$ = DIVISION BY E_2
$G(S)$ = FUNCTION: SQUARE..............$1/G(S)$ = FUNCTION: SUARE ROOT
$G(S)$ = 1/S (INTEGRATION).............$1/G(S)$ = S (DIFFERENTIATION)

Figure 5-18 High Gain Circuits

5.1.4.2 Variable Gain

If the feedback element around a high-gain amplifier is a potentiometer, then the gain of each of the remaining inputs is the labelled gain multiplied by the reciprocal of the potentiometer setting. Any such gain will be greater than one. Gains on summers which are greatly in excess of one usually indicate poor scaling. The exception is when small differences of large quantities are taken. See Figure 5-19.



If the output of a high gain amplifier is patched to the input of a pot, and the output of the pot is patched to an input labelled 10 on the patchboard, then the gain of the remaining 10 input will be:

$$\text{GAIN} = \frac{1}{A}\text{, WHERE A IS THE POT SETTING}$$

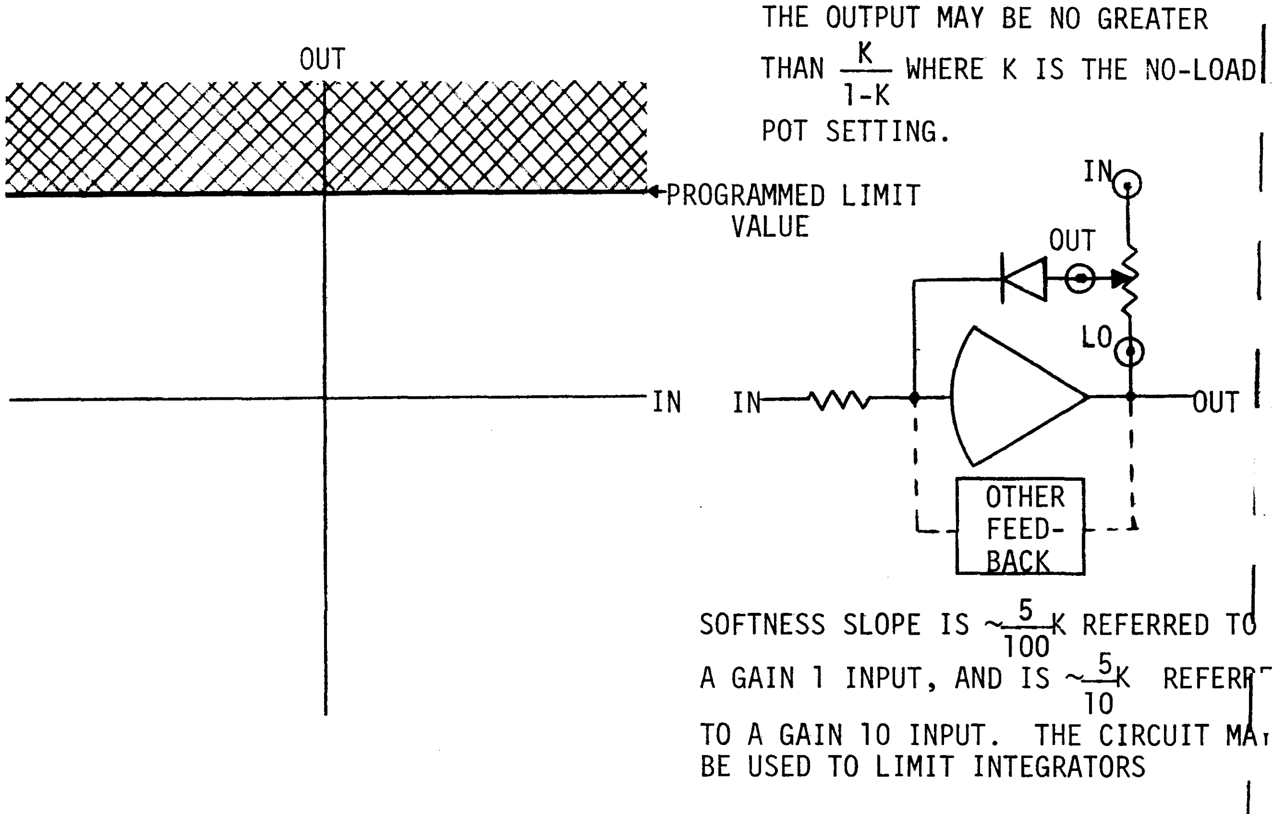
Figure 5-19 Variable Gain Circuit

5.1.5 Potentiometer Circuits

In reading this section, keep in mind that a potentiometer should not be patched to any non-linear device input. The general rule is to patch pots only into amplifier inputs. Trunks—plotters should be avoided because of non-linear loading. The diode circuits shown in this section are permissible because pot loading effects are not a factor in their operation.

5.1.5.1 Soft Limiter

A limiter circuit is a circuit which restricts the output amplitude excursions of an amplifier. The circuit of Figure 5-20 is the classic circuit which restricts the amplifier output from being more positive than some arbitrary value. Negative excursions may be similarly restricted by reversing the diode connections and changing the pot input from -1 to +1. Both circuits may be used concurrently. Action of these circuits is completely independent of any additional feedback elements. For example, a positive and negative limiter with no additional feedback results in a Bang-Bang circuit. The crosshatch area in the characteristic graph of Figure 5-20 indicates the forbidden region. To plot the characteristic of circuits using these limiters, plot the characteristic ignoring the limiter, then replace all line segments in the forbidden region with segments at the margin of the forbidden region. Note that this circuit uses the ungrounded pot. On the AD/FIVE these pots are those with addresses ending in a 4 or a 9. This circuit forms a soft limiter and is described in Figure 5-20. (A hard limiter is available with the AD/FIVE. See Chapter 4, Section 4.2.1.12.)



THE OUTPUT MAY BE NO GREATER THAN $\frac{K}{1-K}$ WHERE K IS THE NO-LOAD POT SETTING.

SOFTNESS SLOPE IS $\sim\frac{5}{100}K$ REFERRED TO A GAIN 1 INPUT, AND IS $\sim\frac{5}{10}K$ REFERRED TO A GAIN 10 INPUT. THE CIRCUIT MAY BE USED TO LIMIT INTEGRATORS

Figure 5-20 Soft Limiter

5.1.5.2 Dead Space Circuit

The ungrounded potentiometers can be used to obtain a dead-space characteristic as shown in Figure 5-21. This circuit has poor performance because the full amount of the silicon junction potential of the diodes is present as an error. It is presented here because other texts sometimes include the circuit with no cautions to the user. Figure 5-12 shows the recommended dead space circuit. Another restriction on the use of this circuit is that the slope of the output characteristic is highly dependent upon the position of the breakpoint, which is set, in turn, by each input potentiometer. The closer the breakpoint is to zero, the greater the slope. If the characteristic has horizontal symmetry, then the gain control afforded by the use of a feedback pot can be used to adjust the required slope, after first setting the desired breakpoint, and the circuit will probably be adequate for use in the hysteresis circuit shown in Section 5.1.5.3. The method described in Section 5.1.6 is the recommended method for adjusting this circuit to achieve the desired characteristic. A final comment on the behavior of this and similar circuits. The circuit will exhibit hysteresis error. This error, easily observed by using the method of Section 5.1.6, is primarily a function of two factors. The first is the input sweep, which is related proportionally to the problem frequency content. The second is the gain of the output amplifier, as adjusted by the feedback pot. The lower the pot setting, the greater the gain, the greater the slope of the characteristic, and the greater the hysteresis error due to amplifier bandwidth degradation. This trade-off between high-speed performance and large slopes is well established, and is the reason why a gain control is usually furnished with an arbitrary function generator such as the AD/FIVE VDFG described in the Appendix to this manual.

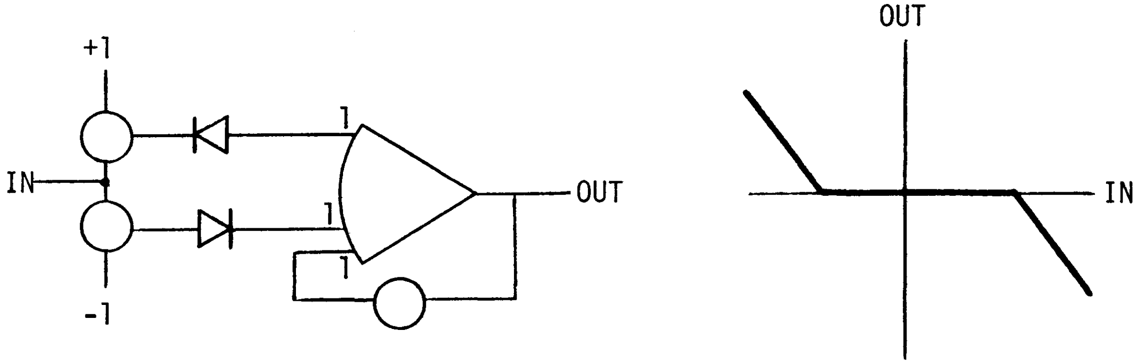


Figure 5-21 Dead Space Circuit

5.1.5.3 Hysteresis

Figure 5-22 shows how to achieve a hysteresis gap in a transfer characteristic. This is useful for the simulation of gear backlash and similar phenomena. The hysteresis value (A) in the curve is equal to the dead space (A). An additional complication arises at high speed because the hysteresis circuit is itself subject to hysteresis error. Indeed, it is considerably more sensitive to high speed than its self-contained dead space circuit. Thus the circuit shown suggests the use of a pot as shown to optimize the performance. Of course, the only way to know what the circuit will do at problem solution speeds is to use the method of Section 5.1.6 in setting up the circuit.

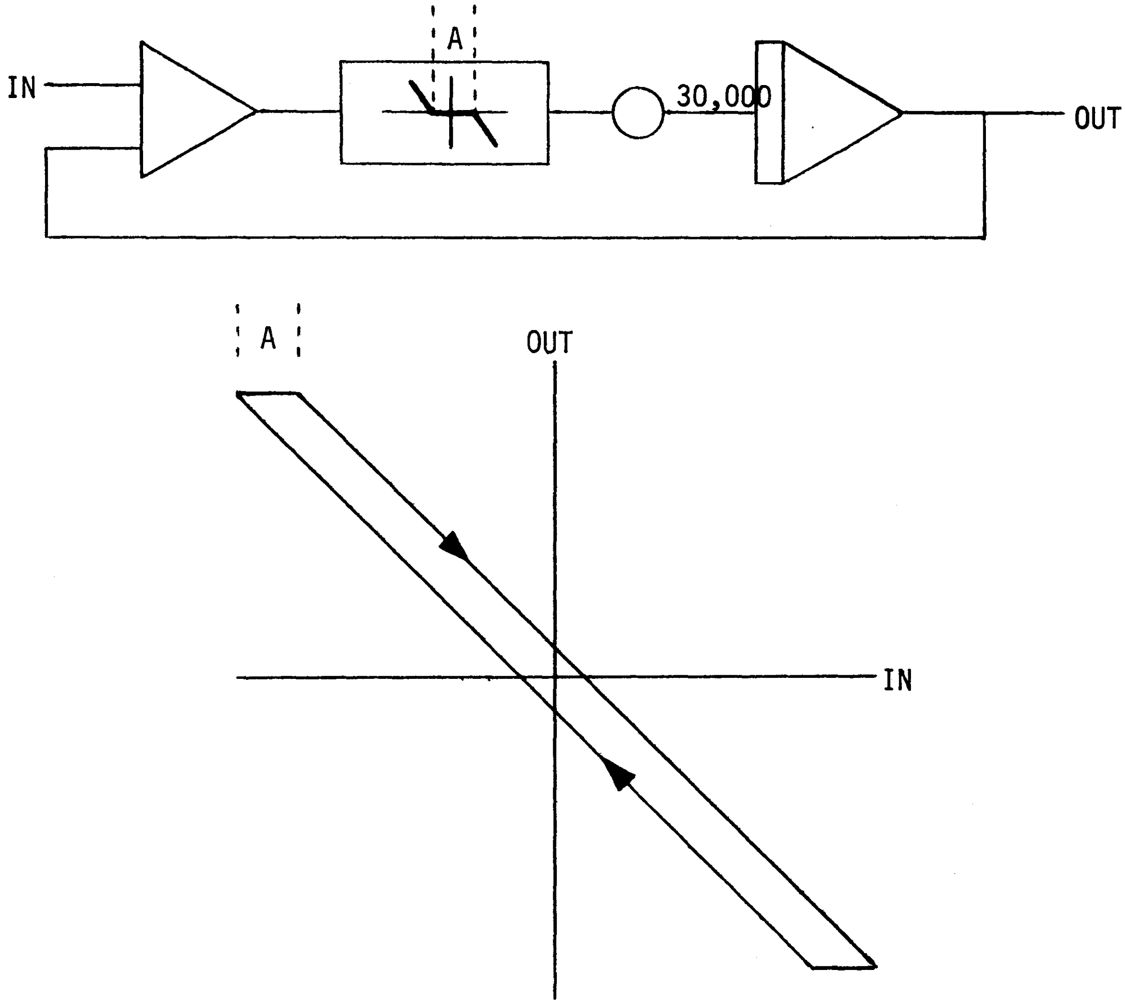
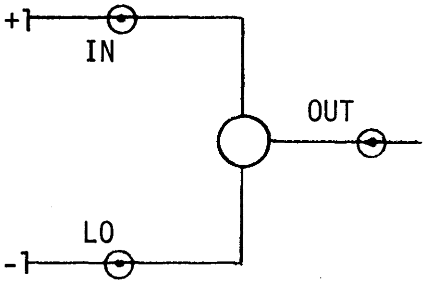


Figure 5-22 Hysteresis Circuit

5.1.5.4 Reference Potentiometer

This is an almost trivial circuit, yet it is quite useful and allows the operator to sweep an input across the full computer range without repatching or closing switches.



Output can be swept from -1 to +1 continuously.

Figure 5-23 Reference Potentiometer

5.1.6 Arbitrary Function Generator: Set Up Grid Reference Pattern

A dramatic reduction in the time required to set up arbitrary function generators is achieved by using optical techniques and the dynamic set up procedure described in the appendix to this manual. A grid pattern, illustrated in Figure 5-24, is generated, and the output (SOL.Y) of the function generator versus the input (SOL.X) is plotted on the grid pattern using the circuit of Figure 5-25. The X and Y outputs of the circuit go to the X and Y inputs of the oscilloscope, and a logic signal is generated to provide a blanking signal which is fed to the Z input of the oscilloscope. A reference amplitude triangle generator (Section 5.1.1.2) provides the function generator input. The oscilloscope is driven alternately from the grid pattern and the function generator output. The output of the function generator is directly displayed against a calibrated grid pattern. Since the output is being viewed by the operator, he may instantly observe the effect of each breakpoint setting as he makes it. This results in the fastest and easiest method of function generator set up; the subject is fully discussed in the appendix. This high-speed technique may require operation of the function generator at speeds higher than intended during the problem run. If so, care must be taken to observe hysteresis effects. This is, however, a distinct advantage over slower set up methods if the problem solution is intended to be run at high speeds, as the operator can observe the output under actual operating conditions. The circuit can be used to plot any transfer characteristic on a grid pattern.

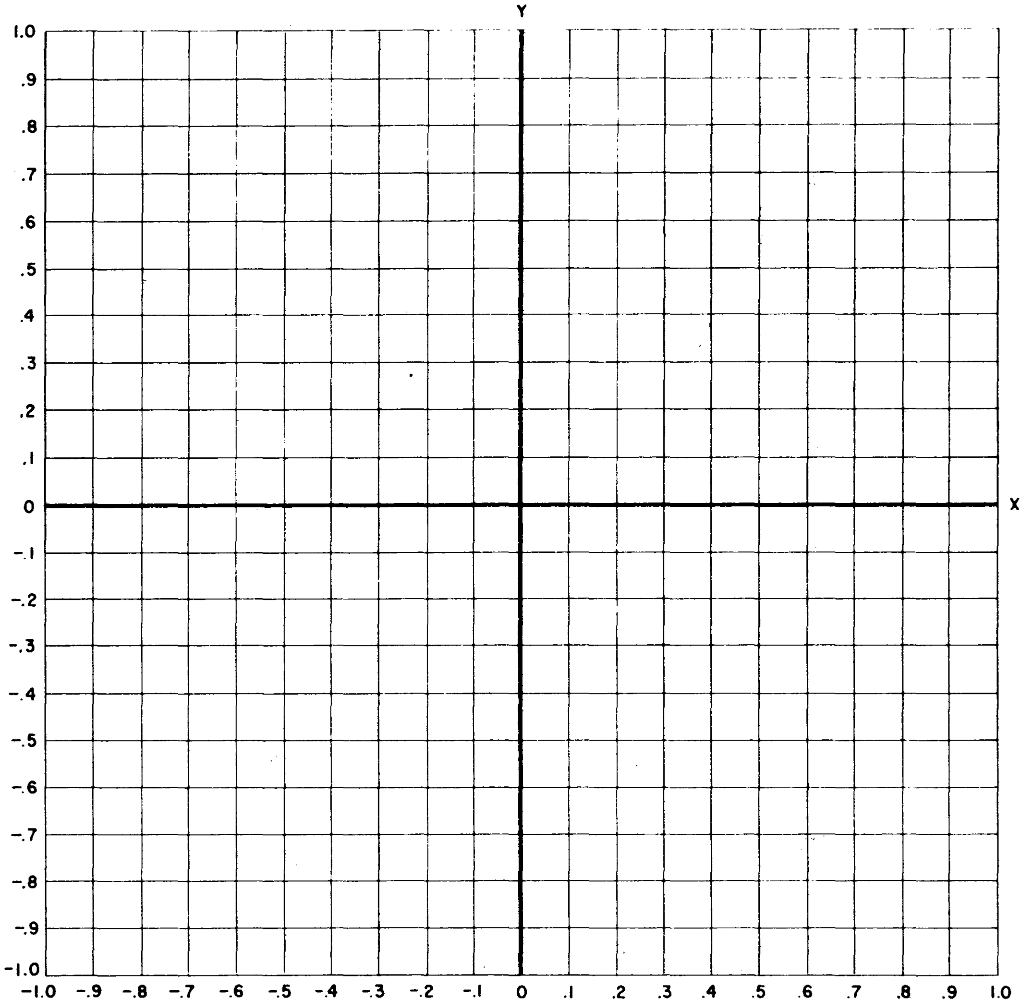


Figure 5-24 Grid Reference Pattern

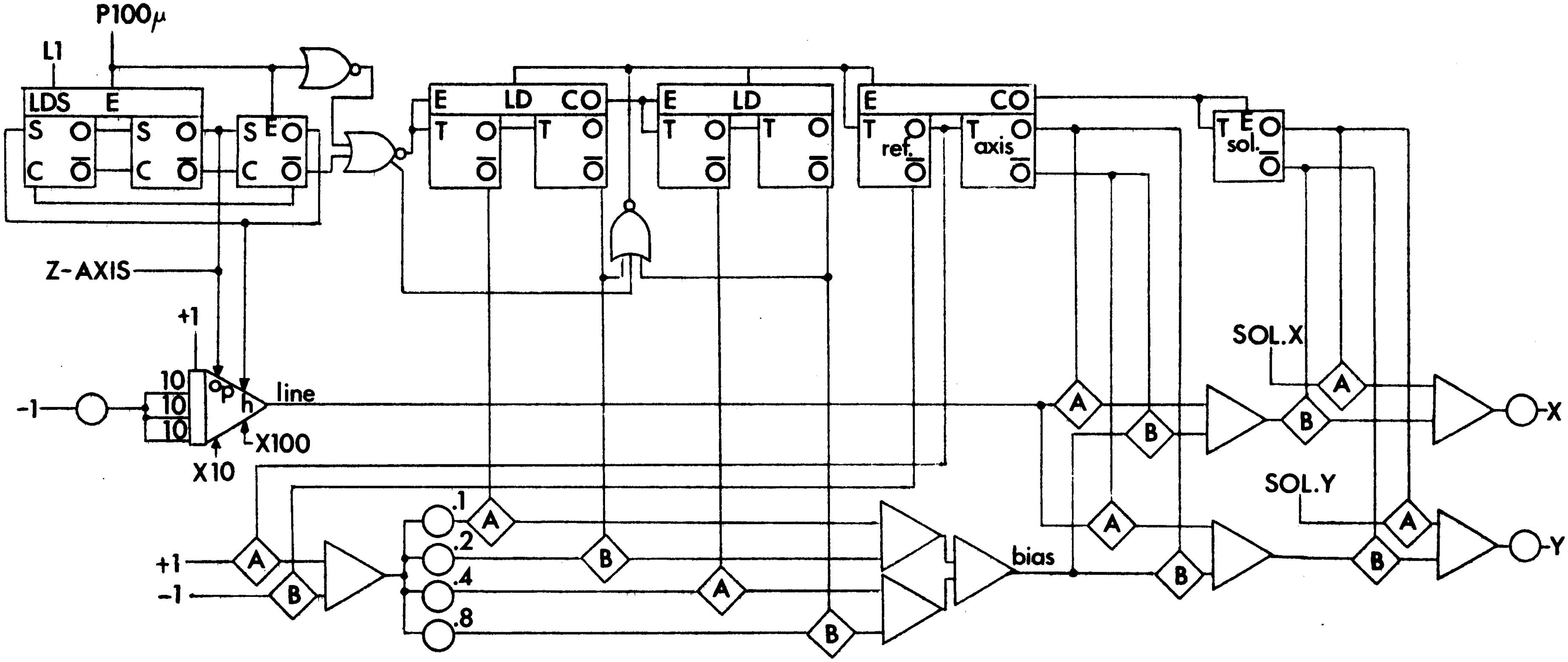


Figure 5-25 Circuit for Grid Reference Pattern Generator for Arbitrary Function Generator Set Up

5.2 CIRCUITS FEATURING LOGIC COMPUTING ELEMENTS

Counters and Flip-Flops have enable inputs which must be patched in order to use the element. If such patching is not shown in the following sections, then it is assumed that the element is enabled with a logic one. If it is desired to have the entire logic program respond to the computer time scale, the V signals are used for enabling. Certain circuits are presented in this section which use the enable input of an element to obtain the basic task performance of the circuit. These circuits are considered to be enabled by a logic one, and are shown in this manner on the Flow Chart Symbol. Figure 5-29 is an example of this practice. The circuits in this section provide at least one example of how to use each of the logic elements. Collectively they constitute a nucleus for the user's application notebook.

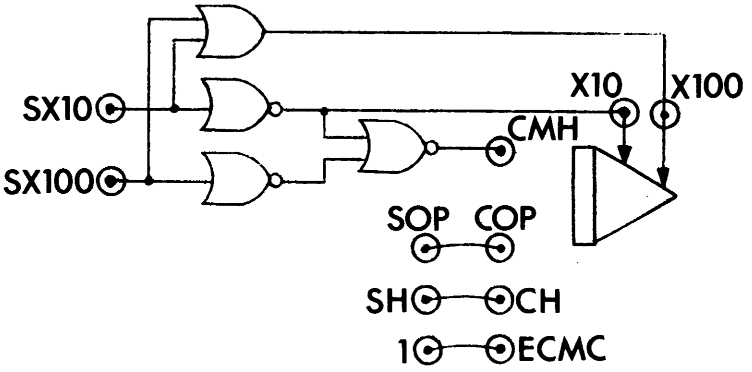
5.2.1 Circuits Using Gates

Gates are a very useful and fundamental logic computing component. The AD/FIVE features OR/NOR gates which can be patched for AND/NAND functions. Section 4.2.2.1 shows the patching configurations. Gates are usually supporting elements in circuits which feature some other functional element. The following sections, however, present circuits which feature the gate as the main component.

Other information which is necessary to know when using gates is given here. Gate outputs can be fanned out to ten other inputs. Gates are not clocked; they are considered to be instantaneous. In the TTL logic used in the AD/FIVE the actual gate propagation time is about 20 nanoseconds, which means that a signal could propagate through the longest possible cascade of gates and be at the destination well prior to the next clock pulse.

5.2.1.1 X10 Circuit

If the X10 gain control hole of an integrator is used in order to increase its inputs by a factor of ten to implement gains required by the analog program then the use of the X10 pushbutton for operator time-scaling is contraindicated, because the patched integrator will not respond to pushbutton control since it is already at X10. (See Chapter 4, Figure 4-1.) This restriction can be overcome with a simple gate circuit as shown in Figure 5-26.



The patching to CMH, COP, and CH provides a Master Hold in case X1000 is inadvertently requested by the operator. This is optional, but highly recommended.

Figure 5-26 X10 Circuit

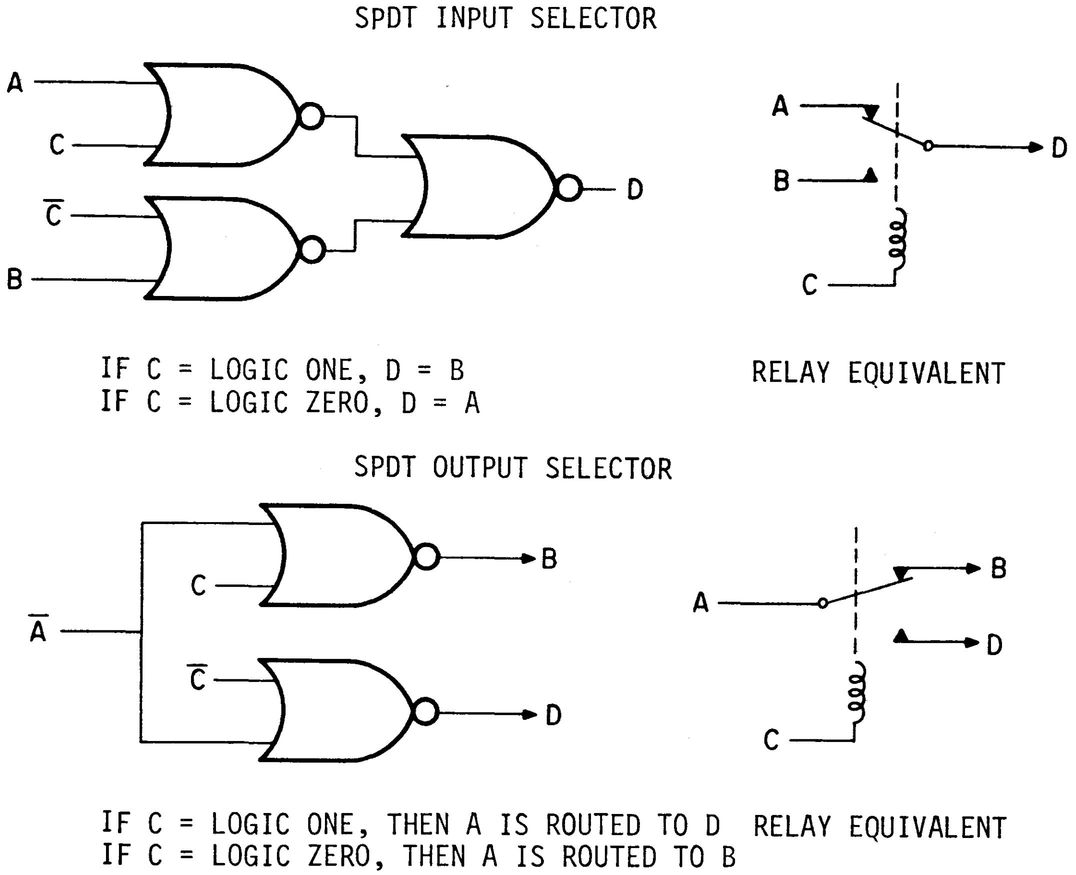
The gain of the integrator which is patched with the X10 circuit will relate to the Console Time Scale as shown below:

CONSOLE TIME SCALE	PUSHBUTTONS	INTEGRATOR GAIN*
X1	NONE	10
X10	X10	100
X100	X100	1000

X1000 IS FORBIDDEN, AND WILL PUT THE CONSOLE IN MASTER HOLD IF THE PATCHING IS AS SHOWN IN FIGURE 5-26.

5.2.1.2 Gate Switches

Gates can be used as automatic high speed switches. Figure 5-27 shows two gate-switch circuits for logic signal routing, and their relay circuit equivalents.



SPDT INPUT SELECTOR

IF C = LOGIC ONE, D = B
IF C = LOGIC ZERO, D = A

RELAY EQUIVALENT

SPDT OUTPUT SELECTOR

IF C = LOGIC ONE, THEN A IS ROUTED TO D
IF C = LOGIC ZERO, THEN A IS ROUTED TO B

RELAY EQUIVALENT

Figure 5-27 Gate Switches

5.2.1.3 Exclusive OR

Refer to Figure 5-28 for the Exclusive OR Circuit, which is patched from three OR/NOR gates. Note that symbols can be labelled with either a term (A·B), or an equation (A = B). This is common convention in logic circuitry. However, the equation symbol does not mean the same in logic labelling as it does in mathematics: it really means a proposition. Thus, if the proposition is true, the output is logic one in Figure 5-28.

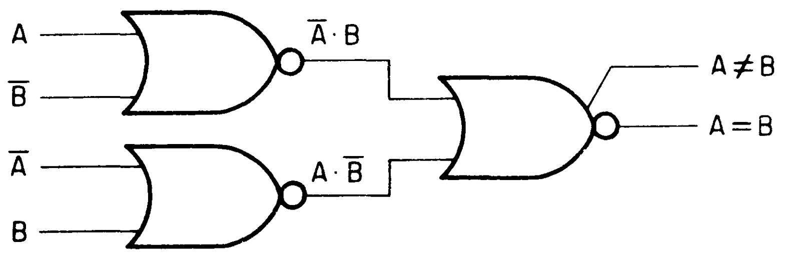


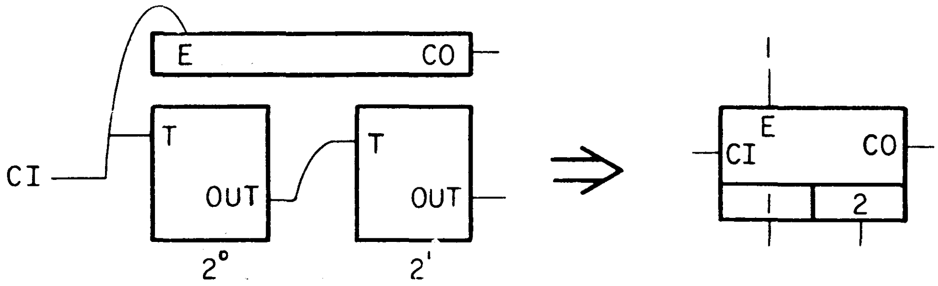
Figure 5-28 Exclusive OR Circuit

5.2.2 Dual Flip-Flop Circuits

The Dual Flip-Flop has internal gates which facilitate its patching as a counter. It is dual because one input enables two flip-flops. It can be initialized by either a panel switch or a patchboard input. The load input is synchronous, so that when the flip-flop is used as a counter the count in the counter can be used to clear (load a zero) the counter before the count is destroyed. This loading is synchronous with the next clock pulse, i.e., it is not gated with the enable input.

5.2.2.1 Two-Bit Binary Counter

The Two-Bit Binary Counter is patched as shown in Figure 5-29. It counts as follows: 0, 1, 2, 3, 0, 1, 2, 3, 0, 1, 2, 3, etc. In other words, it is a Modulo 4 counter. Note the difference between the actual patching of the enable input and the logic one shown on the Flow Chart Symbol. This is not an error. The enable input must be patched from the CI signal. Otherwise a logic one in the 2^0 bit would trigger the 2^1 bit, even if the CI was logic zero. A counter cannot be allowed to change count when the CI is logic zero. Thus the patching is to be done exactly as shown and the counter thus produced is one which is enabled with a logic one, i.e., it counts higher on every clock pulse when CI is logic one. The prudent programmer will be well advised to always consider the rigorous definition of CI and E of a counter. This definition is found in Section 5.2.2.4.



Patching Diagram

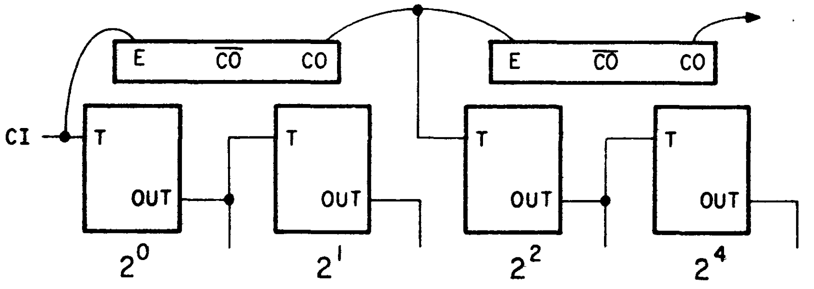
Flow Chart Symbol

Use the CO terminal for output to another counter. The two flip-flops have a common enable. The count is incremented eachtime CI is logic one and there is a clock pulse. CI must be patched to the low numbered (0 or 3) flip-flop of the pair.

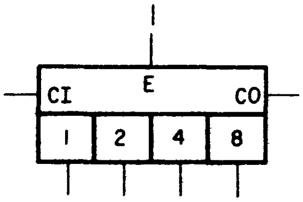
Figure 5-29 Two-Bit Binary Counter

5.2.2.2 Four-Bit Binary Counter

The Four-Bit Binary Counter is made up of two Two-Bit Binary Counters in cascade. It is really a Modulo 16 counter which is always enabled, in the same manner as the Two-Bit Binary Counter of Section 5.2.2.1.



Patching Diagram



Flow Chart Symbol

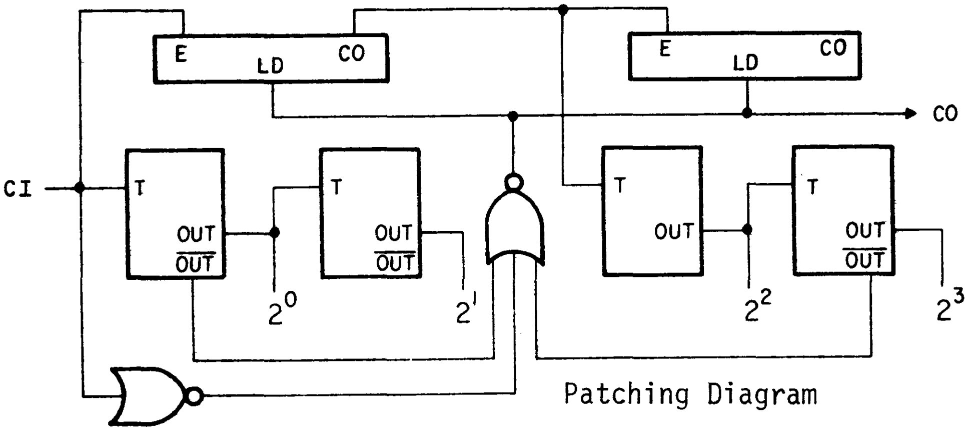
CI must be patched to the low numbered (0 or 3) flip-flop of each pair.

Figure 5-30 Four Bit Binary Counter

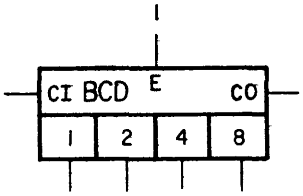
5.2.2.3 BCD Counter With Outputs

Although the AD/FIVE has some very useful BCD Counters, intermediate counts in these counters can be decoded only with certain obvious limitations since the only outputs are CO (Variable, Multiple, or 100) type outputs. A more generally useful BCD Counter can be patched using two Dual Flip-Flops as shown in Figure 5-31.

Switches and LDS inputs should be zero if count is to initially start at zero. If switches or LDS inputs are not zero, then initial count will not be zero, and counter will count to 9 and then cycle to the non-zero value. Switch settings or LDS inputs greater than nine are invalid.



Patching Diagram

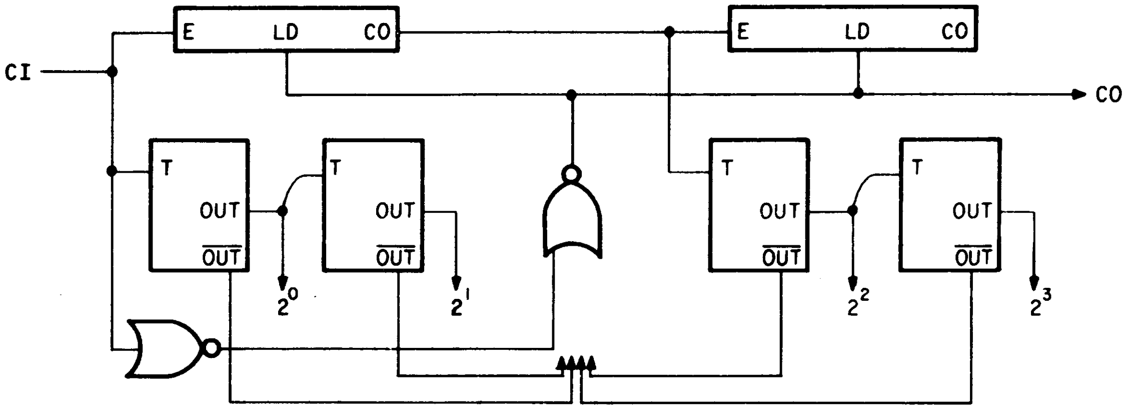


Flow Chart Symbol

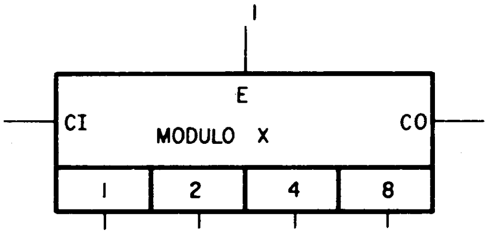
Figure 5-31 BCD Counter With Outputs

5.2.2.4 General Purpose Binary Coded Modulo-X Counter

A General Purpose Binary Coded Modulo-X Counter with count outputs can be built using the flip-flop as the basic element. Binary coding is used. Modulo Two through Modulo Sixteen Counters are discussed in this section; higher values can be easily obtained using the basic principle illustrated in this section. Figure 5-32 shows the general purpose counter.



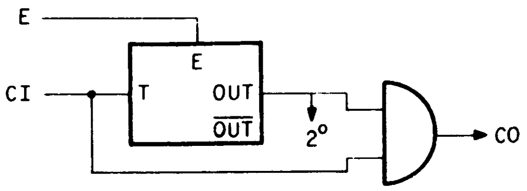
Patching Diagram



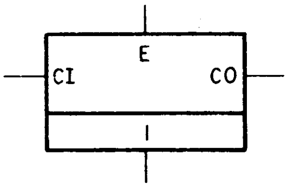
Flow Chart Symbol

Figure 5-32 General Purpose Binary Coded
Modulo-X Counter

Modulo Two: A flip-flop is a Modulo Two Counter. Gating is necessary to obtain a carry-out. A carry-out signal, to be useful, must have the following characteristic: the carry-out is a logic one if the maximum count is present and the carry-in is a logic one. A carry-out can be patched to a carry-in and thus two or more counters can be cascaded to achieve higher count values. The Modulo Two counter is illustrated in Figure 5-33.



Patching Diagram



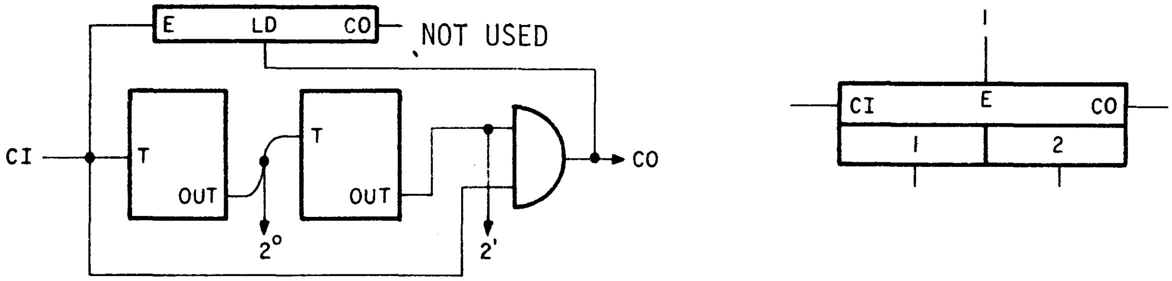
Flow Chart Symbol

Figure 5-33 Modulo Two Counter

The flip-flop output is used to provide the output which indicates that the value of the count in the counter is 2^0.

Care must be taken to observe an additional property of the carry-out signal. It is potentially of indefinite duration,i.e., it will last until the next enable signal if CI is logic one. Thus, in general, a counter has two inputs: CI and ENABLE. Both inputs must be a logic one for the count in the counter to increment. The difference between the two inputs is that the CI contributes to the CO, the enable does not. In cascading counters, care should be taken to enable each unit with the same signal as a general rule.

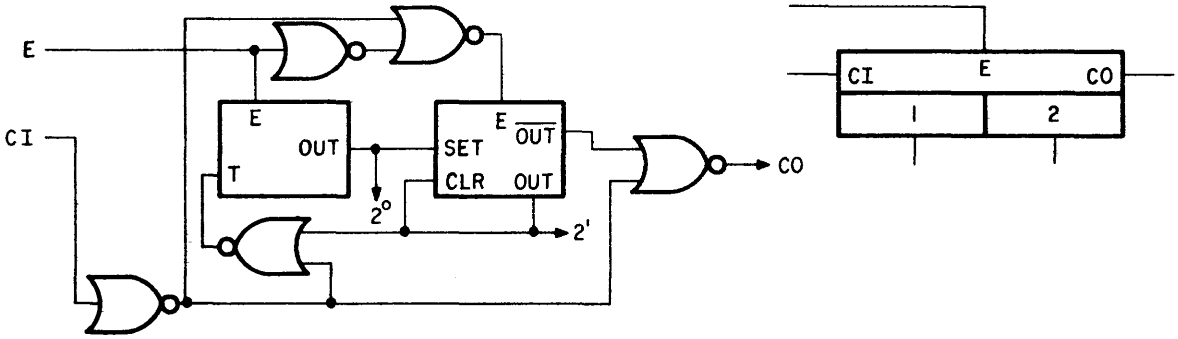
Modulo Three: Decoding is necessary to implement a Modulo Three counter. This is illustrated in Figure 5-34. If a separate enable input is needed, use the circuit of Figure 5-35.



Patching Diagram

Flow Chart Symbol

Figure 5-34 Modulo Three Counter Using Dual Flip-Flop

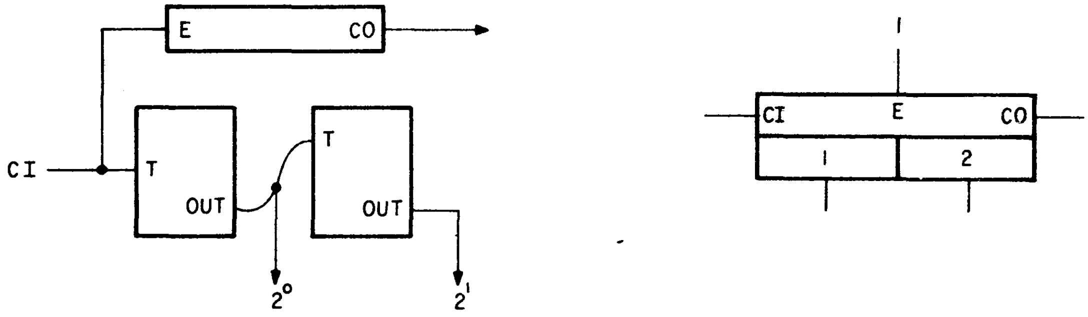


Patching Diagram

Flow Chart Symbol

Figure 5-35 Modulo Three Counter Using Independent Flip-Flops

Modulo Four: This counter is easily patched using the dual flip-flop. This is the same circuit as the Two-Bit counter of Figure 5-29.



Patching Diagram

Flow Chart Symbol

Figure 5-36 Modulo Four Counter

Modulo Five through Modulo Nine: Use the General Purpose Modulo-X Counter of Figure 5-32. Select the appropriate inputs to the gate in order to decode for the proper maximum count.

Modulo Ten: This is really a BCD counter. The BCD counter of Figure 5-31 is a Modulo Ten counter.

Modulo Eleven through Modulo Fifteen: Same as Modulo Five through Modulo Nine.

Modulo Sixteen: Use the Four-Bit counter of Figure 5-30.

It is important to thoroughly understand each of these circuits in order to readily exploit the power of a synchronous logic system. For instance, one very powerful aid to the operator is the Grid Reference Pattern Generator of Figure 5-25, which requires Modulo Four and Modulo Eleven counters.

5.2.3 Differentiator Circuits

The Differentiator performs logic differentiation. Exactly what is meant by that term is a very arbitrarily defined concept. Unfortunately, rigorously defined mathematical concepts such as differentiation are sometimes used to describe processes which only faintly resemble the original concept. Thus the Differentiator in the AD/FIVE is clearly defined as to its actual operation in Chapter 4, Section 4.2.2.3. A brief study of the truth table in Figure 4-22 reveals that the DIF output is a signal of programmable duration which <u>predicts</u> the logic zero to logic one transition of the associated flip-flop. The Differentiator is useful for counting and synchronizing applications.

5.2.3.1 Trailing Edge Differentiator

Complementation of the input to a differentiator will result in an output signal on the trailing edge (1 to 0 transition) of the input signal. Time duration of the DIF output is a function of the ENABLE input and is one clock period if E = logic one. The circuit is shown in Figure 5-37.

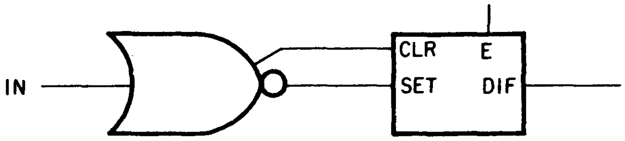


Figure 5-37 Trailing Edge Differentiator

5.2.3.2 Cycle Counter

Frequently it is desirable to count the number of times that an analog comparison is made. This is done by differentiating the output of the comparator so that the counter increments once each time the comparator transitions to a logic one, regardless of the duration of the comparison. The circuit is illustrated in Figure 5-38.

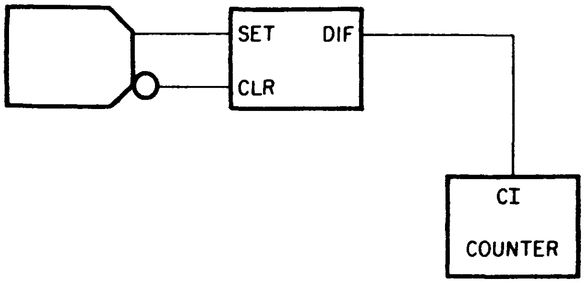


Figure 5-38 Cycle Counter

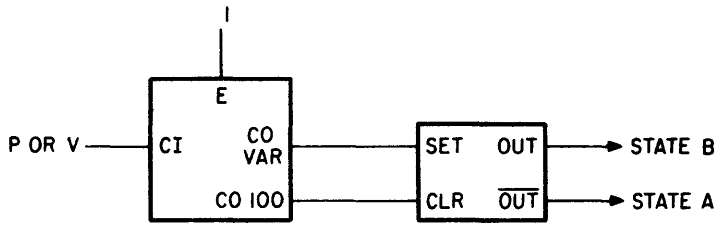
5.2.4 Variable Carry-Out Counter Circuits

The Variable Carry-Out Counter is a useful device. The circuits in this section describe interval timers which can be formed with the VCO counter, and which are only one of many useful applications of the device.

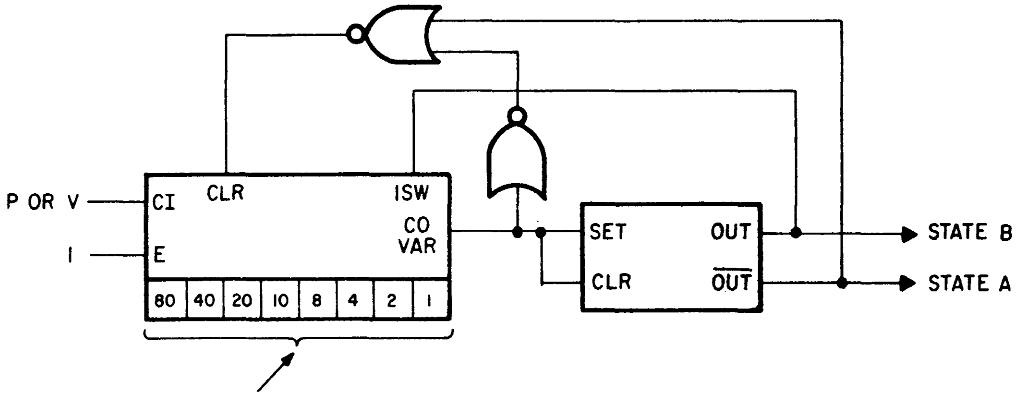
5.2.4.1 Two State Interval Timer

This circuit is presented in two forms in Figure 5-39, and is used to augment the system Interval Timer. Type A is very easy to patch and will serve most applications.

Type A: "A" period and "B" period together total 100 time units. The A period is set on the VCO counter thumbwheel switch (See Chapter 4, Section 4.2.2.4).



Type B: "A" period and "B" period may total any arbitrary number of time units.



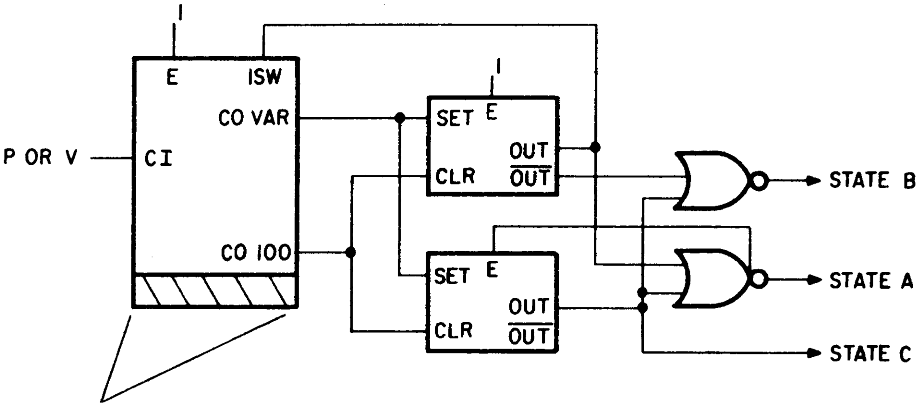
Patch the total cycle time ("A" period plus "B" period). Set the A period on the VCO counter thumbwheel switch.

Figure 5-39 Two State Interval Timer

5.2.4.2 Three State Interval Timer

Figure 5-40 illustrates the use of the Variable Carry-Out counter to form a Three State Interval Timer. This timer will have a total period of 100 time units. Therefore it can be nested within, or operated synchronously with, the system Interval Timer. Operation is as follows: in the load state, both flip-flops are cleared and the count in the counter is zero. The state is "A". After going to RUN, the counter counts the P or V signal which has been selected by the operator. The first flip-flop sets when the time reaches the value set on the VCO counter thumbwheel. This shifts the state to "B" and activates the INHIBIT SWITCH feature (See Chapter 4, Section 4.2.2.4). The count continues to increase until it reaches the patched value ("A" period plus "B" period) whereupon there is again a CO VAR output. The upper flip-flop is already set. Now the second flip-flop sets because it is enabled. The timer shifts to the "C" state. After 100 time units the flip-flops both clear and the cycle repeats.

"A" period is set on the VCO counter thumbwheel switch. "A" period plus "B" period plus "C" period equals 100 time units.



Patch sum of "A" and "B" periods.

Figure 5-40 Three State Interval Timer

5.2.5 Multiple Carry-Out Counter Circuits

The Multiple Carry-Out counter is also a very useful device for which three applications are given in this section

5.2.5.1 Quinary Counter

A Quinary Counter can be patched using the circuit of Figure 5-41. It makes a difference which signal is patched to the CI and which is patched to the ENABLE. Note that the counter will recycle (clear to a count of zero) one microsecond after the CO is a logic one. The counter clears on the next clock pulse (<u>not</u> the next <u>enabled</u> clock pulse) after the CLR input is logic one.

If there is any doubt about what is required, the operator should make a timing diagram to resolve the doubt. This is easily done and should not be neglected.

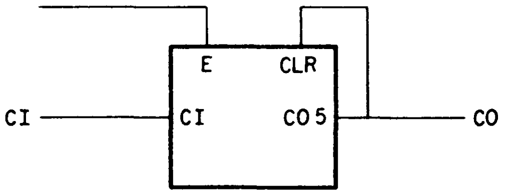
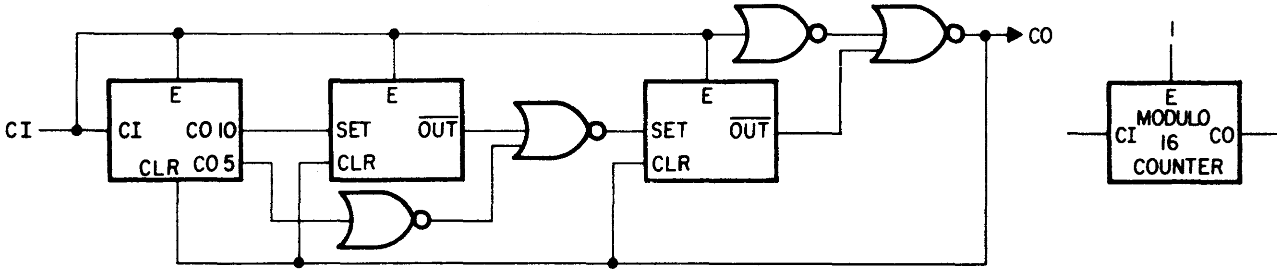


Figure 5-41 Quinary Counter

5.2.5.2 Hexadecimal Counter

The Hexadecimal Counter is a Modulo Sixteen Counter, and is easily implemented by means of the Dual Flip-Flop as shown in Figure 5-42. This circuit is included to illustrate another example of intermediate count-decoding for the Multiple Carry-Out Counter. Thus a counter of any size from Modulo Two to Modulo One Hundred is obtainable. Note the careful attention to the definition of Carry Out in this circuit.



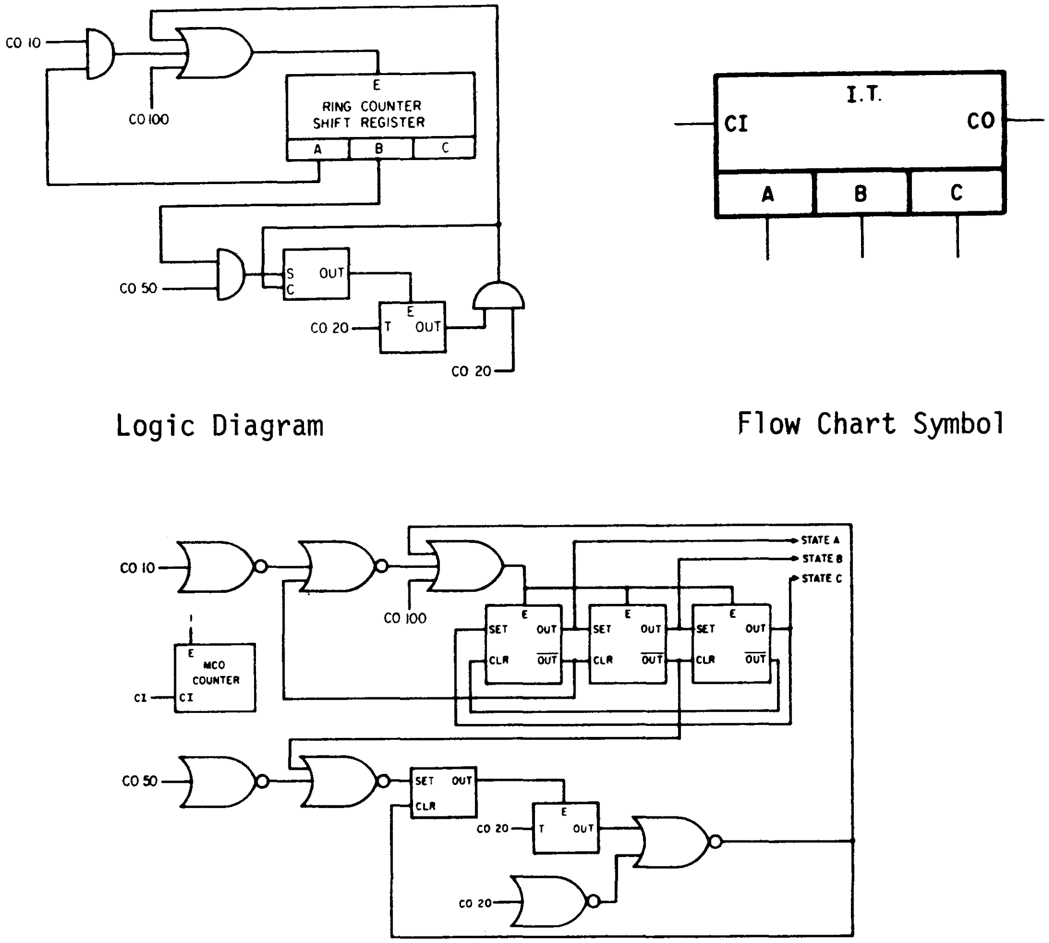
Patching Diagram

Flow Chart Symbol

Figure 5-42 Hexadecimal Counter

5.2.5.3 Three State Interval Timer Using Multiple Carry-Out Counter

Often an interval timer in addition to the system Interval Timer can be useful, as in a nested repetitive operation in which a portion of the simulation is solved repetitively during a single run of the rest of the problem, after which the entire problem is put into repetitive operation for display purpose. The interval timer shown in Figure 5-43 (or the interval timer shown in Figure 5-40), together with the system Interval Timer, will meet such a requirement. The circuit of Figure 5-43 operates as follows: A Multiple Carry-Out Counter counts the P or V signal desired; CO 10 shifts a three bit ring counter to the "B" state. CO 50 then sets a flip-flop enabling a second flip-flop to trigger on the next CO 20 (which is actually a CO 60). This flip-flop then gates the next CO 20 (i.e., CO 80) to shift the register to the "C" state. CO 100 shifts the register to the "A" state and the process repeats. This circuit may not be the best method to perform a desired operation in every case, but the circuit illustrates the use of the ring counter and the technique of intermediate storage of CO signals and the subsequent gating of other CO signals.



Logic Diagram

Flow Chart Symbol

Patching Diagram

Figure 5-43 Three State Interval Timer Using Multiple-Carry-Out Counter

5.2.6 System Interval Timer Circuits and Indefinite-Operate Controller Circuit

The Interval Timer is most often used as a repetitive operation controller. It is, however, a Three-Bit Ring Counter with count values settable from thumbweels or digitally-set registers, and as such can also be used as a versatile logic computing component. It is a timer only if the signal being counted is periodic. The Interval Timer can also be used as a free, patchable logic device. Figure 5-44 shows one useful application of the Interval Timer as a computational element.

The Indefinite-Operate Controller circuit solves the following control problem:

> The "A" period is to be set by the thumbwheels or by digitally-set registers, and normal operation in the "A" period is desired.
>
> The "B" period is to be of indefinite duration, lasting until some analog event causes a comparator to have a logic one present at its output. The "B" period is set to 01. The "B" period will then last one micro-second after the comparison occurs.
>
> The "C" period is to be set from the thumbwheel or from digitally-set registers and normal operation in the "C" period is desired.

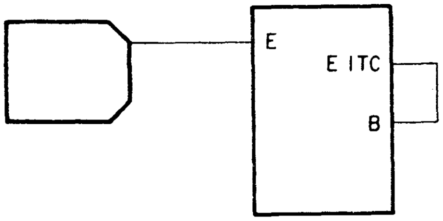


Figure 5-44 Indefinite-Operate Controller

APPENDIX

VARIABLE DIODE

FUNCTION GENERATOR

TABLE OF CONTENTS

ILLUSTRATIONS

1.0 INTRODUCTION

The E3.17 Variable Diode Function Generator (DFG) can be operated as two independent ten segment fixed breakpoint Diode Function Generators, or as one ten segment variable breakpoint DFG. In the fixed breakpoint mode the breakpoints of the DFG function are .1 machine unit (1 volt) apart; in the variable breakpoint mode breakpoints can be spaced as far apart or as close together as desired. The means of achieving either mode of operation are discussed in following sections of this manual.

If the E3.17 is not used for function generation, two output amplifiers are available for use as inverters.

Figure 1.0 shows the front panel of the E3.17 Variable Diode Function Generator.

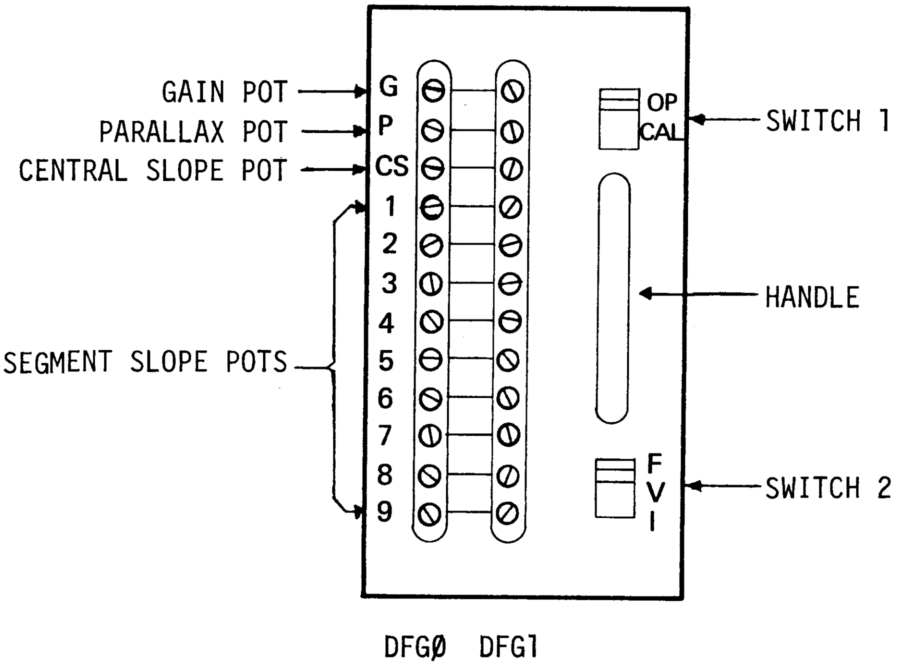


Figure 1.0 E3.17 Variable Diode Function Generator Front Panel

The gain, parallax, and slope adjustment pots can be turned with a small tip screwdriver or a tuning wand.

Each DFG is addressed as an element in the Nonlinear Class (Class N). For example, DFGØ terminated in Field Ø, Area 1, is addressed as NØ1Ø.

1.1 Fixed Breakpoint Operation

Switches 1 and 2 on the front panel control the modes of operation of the DFG. With switch 1 in the OP (Operate) position and switch 2 in the F (Fixed) position, the E3.17 is configured as two independent ten segment fixed breakpoint DFGs. Figure 1.1 shows the relationship between these DFGs and the patchboard terminations.

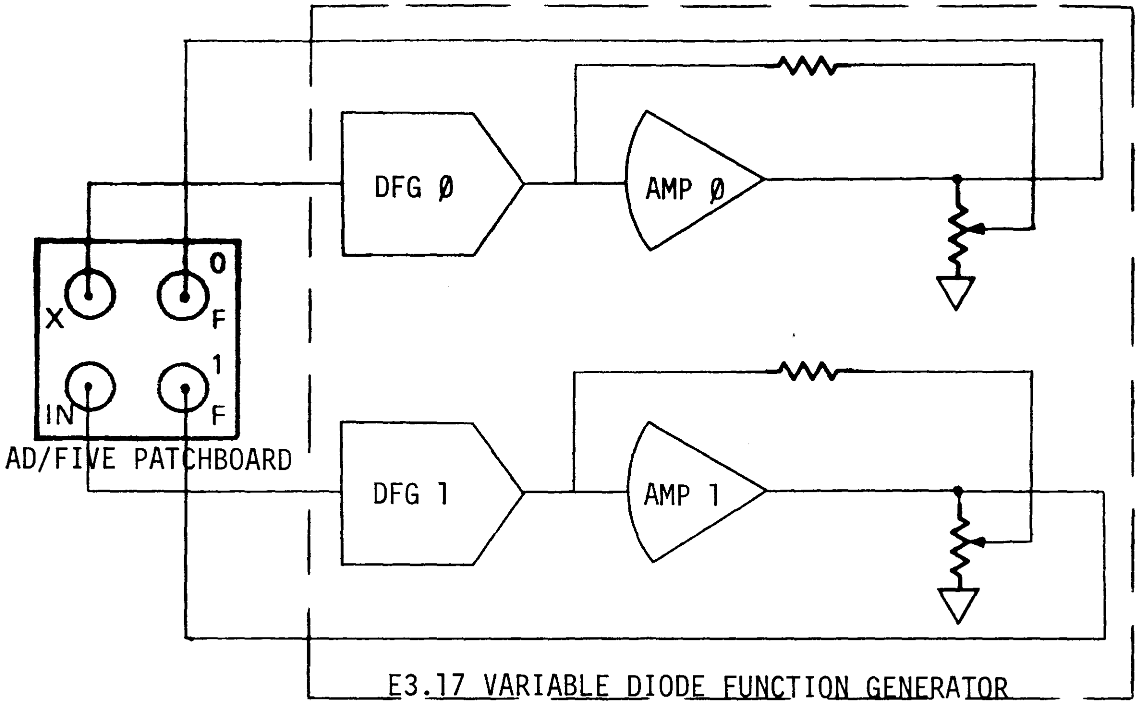


Figure 1.1 DFG Fixed Breakpoint Configuration

The method of programming the DFGs for the fixed breakpoint mode is described in Sections 3.1 and 3.5 of this manual.

1.2 Variable Breakpoint Operation

With switch 1 in the OP position and switch 2 in the V (Variable) position the E3.17 is configured as a single ten segment DFG with variable breakpoints. Figure 1.2 shows the relationship between this DFG and the patchboard terminations.

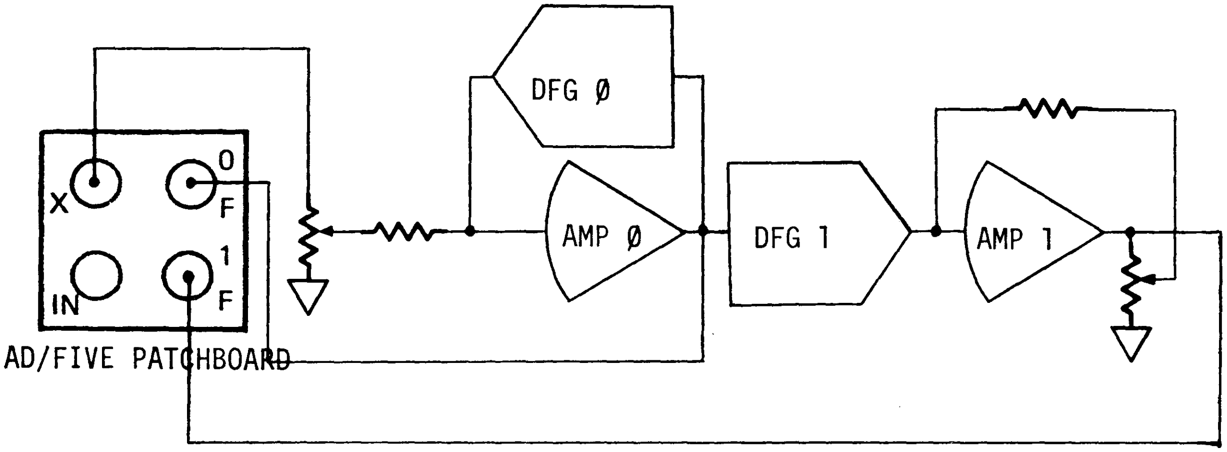


Figure 1.2 DFG Variable Breakpoint Configuration

Putting switch 2 in the V position puts the DFG network of the DFGØ channel into the feedback of the DFGØ channel output amplifier. In this configuration, DFGØ is programmed to provide the desired breakpoint locations, and DFG 1 is programmed to provide the desired segment slopes for the final desired function approximation. Sections 3.2 and 3.6 describe the method of programming the DFGs for the variable breakpoint mode.

1.3 Inverter Operation

With switch 1 in the OP position and switch 2 in the I (Inverter) position the DFG output amplifiers are terminated on the patchboard as two unity gain inverters as shown in Figure 1.3.

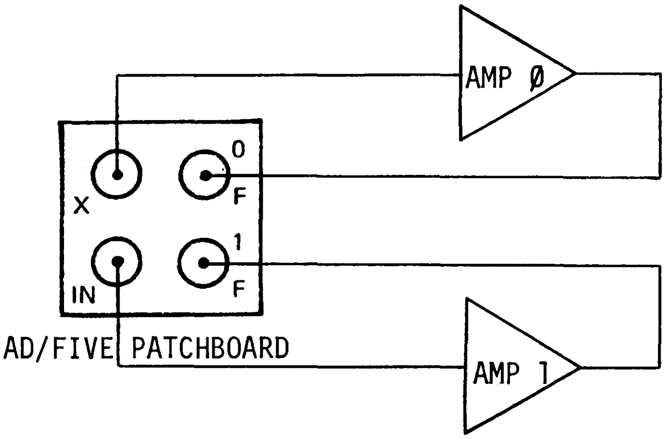


Figure 1.3 DFG Inverter Configuration

1.4 DFG Function Calibration Mode

The E3.17 has a special static set up or function calibration feature to aid the programmer. When switch 1 is put in the CAL (Calibration) position the action of switch 2 is overridden. Regardless of the position of switch 2, the E3.17 is configured as shown in Figure 1.4 when switch 1 is in the CAL position.

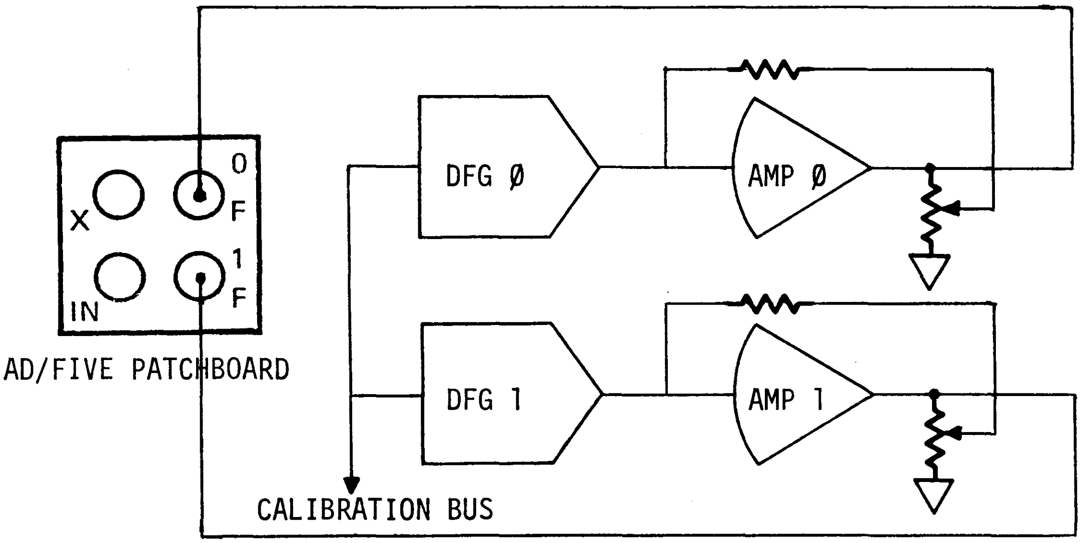
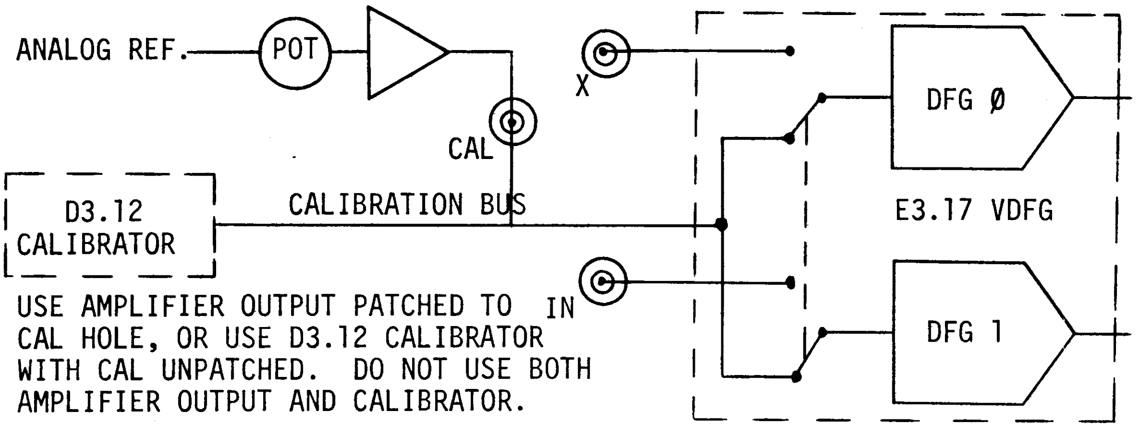


Figure 1.4 DFG Function Calibration Configuration

The Calibration Bus of Figure 1.4 is wired to each variable diode function generator slot in the AD/FIVE and terminates on the patchboard in a hole labelled CAL, located in Field 0 Area 0. The calibration feature is used in static set up of desired functions in the DFGs. An amplifier output can be patched in the CAL patchhole and used as a voltage source for DFG set up. The D3.12 VDFG Calibrator can also be used for this purpose.

In static set up of a DFG, the input voltage is fed to the Calibration Bus either from an amplifier output or from the D3.12 VDFG Calibrator. The breakpoint voltage is fed into the DFG, and the gain, parallax, and slope pots are used to achieve the desired output at each input breakpoint voltage. After set up is completed, switch 1 is put in the OP position, and the accuracy of the function can be ascertained and corrected if necessary.

The D3.12 VDFG Calibrator is a programmable voltage source designed to be plugged into any one of the VDFG slots in the area behind the hinged panel above the patchboard. The desired output voltage from the D3.12 is obtained via thumbwheel switches on the front panel of the Calibrator. This output is directly connected to the Calibration Bus. The use of this unit greatly simplifies the set up of the E3.17 VDFG.



USE AMPLIFIER OUTPUT PATCHED TO CAL HOLE, OR USE D3.12 CALIBRATOR WITH CAL UNPATCHED. DO NOT USE BOTH AMPLIFIER OUTPUT AND CALIBRATOR.

Figure 1.5 Calibration Mode Inputs

2.0 THEORY OF OPERATION

The E3.17 VDFG produces a straight line segment approximation to an arbitrary function of X by generating functions as shown in Figure 2.0 and summing these by connecting them in parallel.

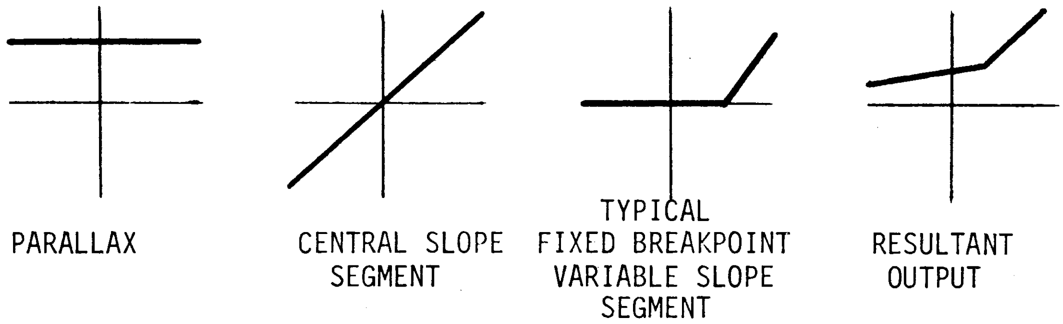


Figure 2.0 DFG Operation

A simplified schematic of the circuitry used to generate a fixed breakpoint variable slope segment is shown in Figure 2.1. (VB is the breakpoint voltage.)

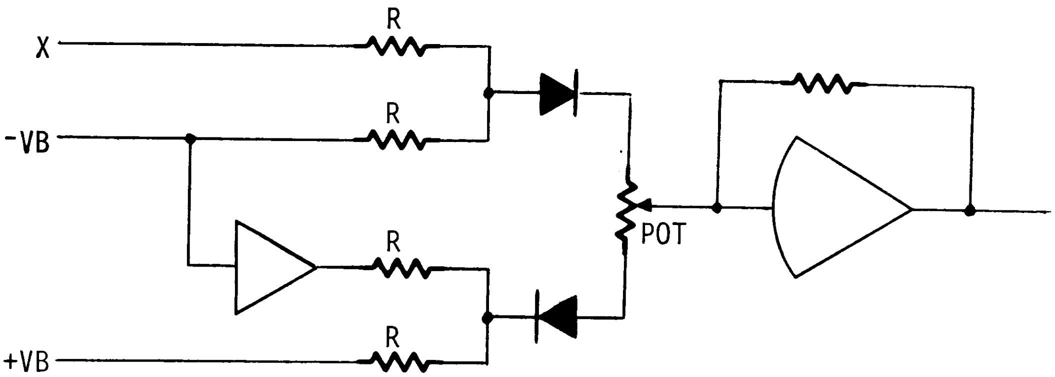
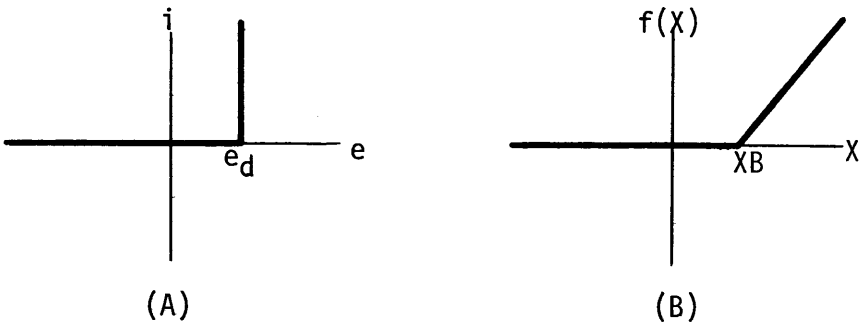


Figure 2.1 Simplified Slope Segment Circuit

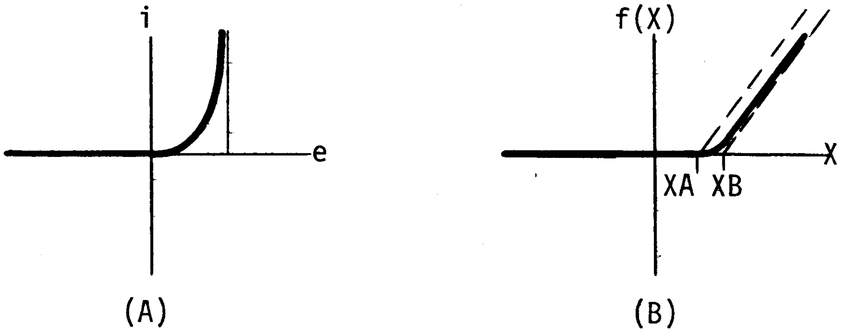
The slope is determined by the setting of the potentiometer. For (X-VB) < $2e_d$, where e_d is the diode potential, the diodes do not conduct and the amplifier output is not affected by X. For (X-VB) > $2e_d$ the diodes conduct, and the segment then contributes to the amplifier output. If the diodes had the ideal switch characteristic shown in Figure 2.2(A), then the circuit shown in Figure 2.1 would produce an output as shown in Figure 2.2(B).



(A) (B)

Figure 2.2 Ideal Diode Switch Characteristics

The diodes do not, however, exhibit the ideal i-e switch characteristics shown in Figure 2.2(A). The i-e characteristic is instead an exponential function as shown in Figure 2.3(A) below. As a result of the actual as opposed to ideal switch characteristics of the diodes, the circuit shown in Figure 2.1 actually produces an output as shown in Figure 2.3(B).



(A) (B)

Figure 2.3 Actual Diode Switch Characteristics

This rounding of the output f(X) in the vicinity of the breakpoint is significant in a computer with a 10 volt reference. The rounding can generally be used by the programmer to obtain a better approximation to a desired function than would be possible with the idealized characteristic of Figure 2.2(B).

For the function shown in Figure 2.2(B) there is no question that XB constitutes the breakpoint voltage. This is not the case, however, for the function shown in Figure 2.3(B); the definition of the breakpoint voltage in this case is somewhat arbitrary. The nominal values for the breakpoint voltages of the E3.17 VDFG are (in machine units) X = 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, and 0.9. These values correspond approximately to the point labelled XA in Figure 2.3(B).

3.0 VDFG SET UP PROCEDURES

In preparing to set up the E3.17 VDFG it is necessary to have a clear understanding of the function of which a straight line segment approximation is desired. A graph of the function, or a function value table, should be prepared. Studying the function will enable the operator to decide whether to use a fixed breakpoint approximation or a variable breakpoint approximation. Once this decision is made the operator is ready to set the VDFG to deliver a straight line segment approximation of the desired function.

There are two methods of setting the VDFG: the dynamic method and the static method. The dynamic method requires a display scope with a graticule. The DFG is set up with switch 1 in the OP position. The function is set up and observed at the same time, so that adjustments to the pots of the segments can be made until the desired function approximation is obtained. The advantages of this method lie in the ease of set up, the possibility of directly seeing the effects of diode rounding for each breakpoint, the ability to make adjustments to compensate for diode rounding while viewing the function, and the ease of obtaining a close overall approximation of a function. Function values cannot be as accurately set at each input value as in the static method. In the static method, switch 1 is in the CAL position. Breakpoint input voltages are supplied to the DFG from the D3.12 VDFG Calibrator, or from the output of an operational amplifier (see Figure 1.5). The function is set for each input voltage and the output is read from the AD/FIVE digital ratiometer (DRM) until the entire function is set for each input voltage. The advantages of this method are the elimination of the need for a display scope, the ability to achieve accurate function values for given input values via the DRM readout; the disadvantages are the difficulty of using diode rounding to advantage to achieve the best overall approximation of the desired function, and the greater complexity of set up calculations. However, a great degree of accuracy can be obtained through the use of this method, as described in Sections 3.3 - 3.6 of this manual.

3.1 VDFG Set Up: Dynamic Method, Fixed Breakpoint

In the fixed breakpoint mode the DFG is programmed to provide a straight line segment approximation of a desired function with nine equally spaced breakpoints whose nominal X values are 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, and 0.9. The central slope term provides an additional segment, so that there are ten segments in the function approximation.

The first step in the dynamic set up of a DFG is to draw the function of which an approximation is desired. The display scope is then set up with a graticule to enable the function to be set as desired. The X input to the DFG is patched to the DFG, and also is an input to the scope. The DFG output is connected to the scope. The scope should be calibrated to the graticule so that an accurate approximation of the desired function can be achieved. See Figure 3.0.

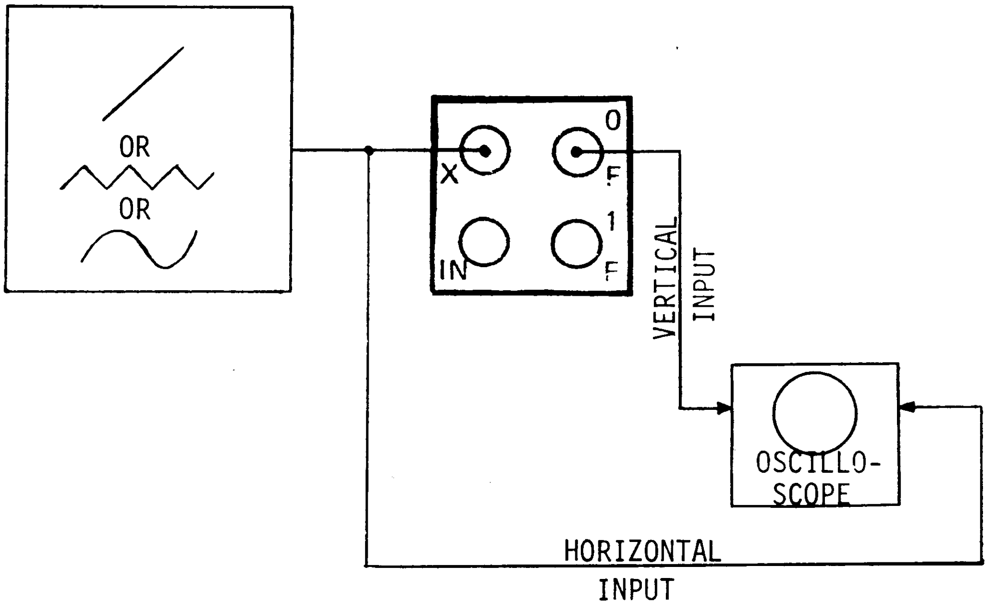


Figure 3.0 Oscilloscope Connections

The input to the DFG may be an integrator integrating a constant from an initial condition of Ø if the function is for $0 \leq X \leq 1.0$, or from an initial condition of -1.0 if the function is for $-1.0 \leq X \leq 1.0$; a triangle wave input or a sine wave input with appropriate amplitudes may also be used.

Switch 1 is set to the OP position. Switch 2 is set in the F position. The DFG is then set to the desired function as follows:

1. Set the DFG gain pot labelled G full counterclockwise for minimum gain. It is desirable to have the gain set as low as possible in order to achieve the best dynamic response. Begin to set the function with the gain at minimum. If at any time you encounter a segment which can not be set with gain at minimum, turn the gain pot clockwise until the desired value is achieved, then return to step 2 and begin to set the function from the beginning. (See also Section 3.4.)

2. Turn the parallax pot labelled P until the desired f(X) is obtained at X = 0.

3. Turn the central slope pot labelled CS until the central slope segment reaches the desired f(X) at X = 0.1.

4. Turn the segment 1 pot labelled 1 until the first segment reaches the desired f(X) at X = 0.2; if adjustment of the segment 1 pot affects the f(X) at X = 0.1 due to diode rounding of the first breakpoint, readjust the central slope pot until the desired result is achieved; then adjust the segment 1 pot again until the desired result is obtained.

5. Continue to set the succeeding segments in order until the desired function is obtained for all X values, paying careful attention to the interaction of segments upon one another at the breakpoints.

The DFG is now set for the desired function approximation, and will not need to be adjusted further until a new function approximation is desired.

3.2 VDFG Set Up: Dynamic Method, Variable Breakpoint

When a fixed breakpoint approximation to the desired function will not suffice, DFG Ø and DFG 1 may be used together to provide a ten segment approximation with nine variable breakpoints whose locations may be determined by the programmer. DFG Ø creates a function which transforms the X values of the desired variable breakpoints into a fixed breakpoint output with the breakpoint 0.1 machine unit apart. This is achieved by switching the DFG network of DFG Ø into the feedback look of the DFG Ø output amplifier, as shown in Figure 1.2. The output of DFG Ø is internally connected to the input of DFG 1. Since DFG Ø transforms the variable breakpoints to fixed breakpoints, the output of DFG 1 yields a final function which has breakpoints at variable X values as determined by the set up function of DFG Ø.

First the graph of the desired function is studied to determine the X values at which breakpoints are desired. The display scope is set up with a graticule. The X input is patched to the DFG input and connected to the scope. The output of DFG Ø labelled ØF is connected to the scope, and the scope is calibrated to the graticule.

Switch 1 is set to the OP position. Switch 2 is set in the F position. DFG Ø is then set up as follows:

1. Set the DFG gain as described in step 1 for the fixed breakpoint set up.

2. Set the parallax pot until the function f(X) is at zero for X = 0. This is necessary because of feedback restrictions for DFG Ø. The desired function must begin at the origin of the axes.

3. The function for DFG Ø must lie in the fourth quadrant due to feedback restrictions. The function values at each X input to DFG Ø are the negative values of the desired final variable breakpoints. The central slope pot is adjusted until the central slope segment reaches the negative value of the first desired breakpoint. For example, if the first desired variable breakpoint is at X = 0.310, the the central slope pot of DFG Ø is adjusted until f(X) at X = 0.1 is -0.310. Following the procedure for the fixed breakpoint set up, each segment is adjusted so that the f(X) for each breakpoint equals the negative value of the desired variable breakpoints as shown below:

X	POT	f(X)
Ø	P	Ø
.1	CS	$-BP_1$
.2	1	$-BP_2$
.3	2	$-BP_3$
.4	3	$-BP_4$
.5	4	$-BP_5$
.6	5	$-BP_6$
.7	6	$-BP_7$
.8	7	$-BP_8$
.9	8	$-BP_9$
1.0	9	-1.0

BP refers to the X values of the desired variable breakpoints.

Note that at X = 1.0 the segment 9 pot is adjusted to yield an f(X) of -1.0. Since DFG Ø provides the input to DFG 1, the output of DFG Ø must be full scale in order to drive DFG 1 full scale. Thus it is necessary that segment 9 reaches -1.0 for DFG Ø.

DFG Ø is now set.

Disconnect the output of DFG Ø from the scope. Connect the output of DFG 1 to the scope. Set switch 2 in the V position. You must now remove the X input, invert it, and patch the inverted input to the DFG. DFG 1 is now ready to set as follows:

1. Set the gain to minimum as discussed in the section on fixed breakpoint set up.

2. Set the parallax pot until the desired f(X) for the final variable breakpoint function is obtained at X = 0.

3. Set the central slope pot until the desired f(X) is obtained at the X value of the first variable breakpoint.

4. Set the first segment pot until the desired f(X) is obtained at the X value of the second variable breakpoint. Correct the central slope setting if needed.

5. Set the remaining segments and adjust for diode rounding as necessary, until the segment 9 pot is adjusted for the desired f(X) at X = 1.0.

DFG Ø and DFG 1 are now set in the variable breakpoint mode.

3.3 Using Diode Rounding to Advantage

The rounding of the straight line approximation of a function due to the switch characteristics of diodes as described in Section 2.0 enables the programmer to obtain a better approximation of a desired function than would be the case if the approximation generated by the VDFG actually consisted of perfectly straight line segments.

In order to achieve the best results in programming the DFG it is beneficial to have a clear understanding of the effects of diode rounding, so that this phenomenon may be used to advantage. The best way to achieve this understanding is to program a DFG for maximum slope difference between the segments and to study the results. This is easily done, and can be of great help when the DFG is being programmed to generate a function approximation. The maximum slope difference is programmed as described below:

1. A display scope is set up as described in Section 3.1, and a suitable input is patched into the DFG.

2. Turn the gain pot full clockwise to maximum. This will enable the effects of diode rounding to be most clearly visible.

3. Set the parallax to zero, so that the central slope segment begins at the origin of the axes.

4. Turn the central slope pot full clockwise for maximum positive slope.

5. Turn the segment 1 pot full counterclockwise for maximum negative slope.

6. Turn the segment 2 pot full clockwise for maximum positive slope.

7. Proceed in this way, turning alternate pots counterclockwise and clockwise until all the pots are set. You should now have a function similar to that shown in Figure 3.1.

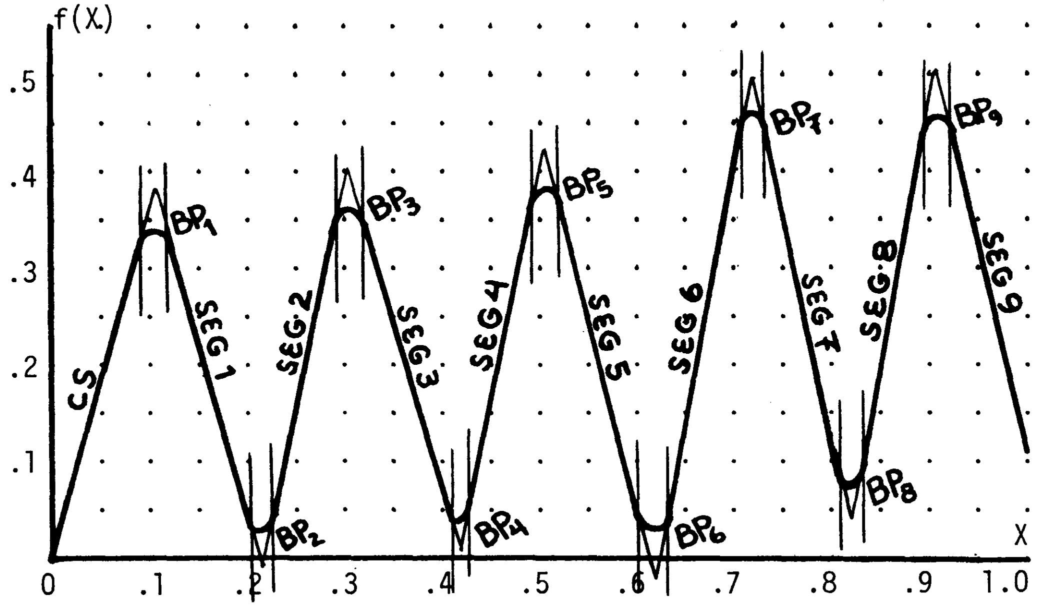


Figure 3.1 Maximum Slope Difference Function

The straight lines in lighter print represent the idealized straight line segments which would be obtained if diode switch characteristics had the graph of Figure 2.2(A); the curve at each breakpoint represents the actual function in the area of diode rounding. The straight vertical lines at each breakpoint represent XA and XB as shown in Figure 2.3(B); XA shows the point at which measurable diode rounding begins, and XB shows the point at which measurable diode rounding disappears. A table of the diode rounding effects for each breakpoint of the DFG for which the maximum slope difference function was programmed should be made as shown below:

BREAKPOINTS:	BP_1	BP_2	BP_3	BP_4	BP_5	BP_6	BP_7	BP_8	BP_9
XA:	.09	.20	.29	.41	.49	.60	.71	.81	.89
XA MINUS .02:	.07	.18	.27	.39	.47	.58	.69	.79	.87
XB:	.12	.22	.32	.43	.52	.64	.73	.83	.92

This table will be somewhat different for each individual DFG unit. Thus the programmer should identify the table with the DFG number for future use.

In the following sections on the static method of DFG set up, the discussion will be based on a DFG which has the diode rounding characteristics shown in Figure 3.1 and the table above. This will serve as an actual programming example to aid the reader in understanding the static method of DFG set up.

With an understanding of diode rounding the operator will be able to achieve an excellent approximation of a desired function. Figure 3.2 illustrates how to obtain a desired value at the nominal breakpoint X value. The curve being approximated is shown in each figure. For purposes of discussion the difference between the desired curve and the straight line approximation is exaggerated. In actual practice a curve of this nature would best be approximated with a variable breakpoint function, but in Figure 3.2 a fixed breakpoint approximation is used in order to clearly illustrate diode rounding. Only the first two segments are shown.

Figure 3.2(A) shows an idealized straight line approximation. (B) shows the central slope set so that the desired value of f(X) represented by the dot is located on the nominal breakpoint value of 0.1. When the segment 1 pot is adjusted for the desired f(X) at X = 0.2, the diode rounding which begins at XA will cause the f(X) set by the central slope pot to change its position to the point shown by the dot. If the operator resets the central slope pot to the desired f(X) as shown in (D), segment 1 will change its position as shown. The segment 1 pot is adjusted for the desired f(X) at X = 0.2 as shown in (E), thus changing the f(X) at X = 0.1. With each pass through the setting procedure the actual f(X) at X = 0.1 comes closer to the desired curve. Three passes through the entire function are recommended to achieve satisfactory accuracy of f(X) values at the nominal breakpoint X values. The third pass through the set up shown in Figure 3.2(F) and (G) provides the desired f(X) at X = 0.1.

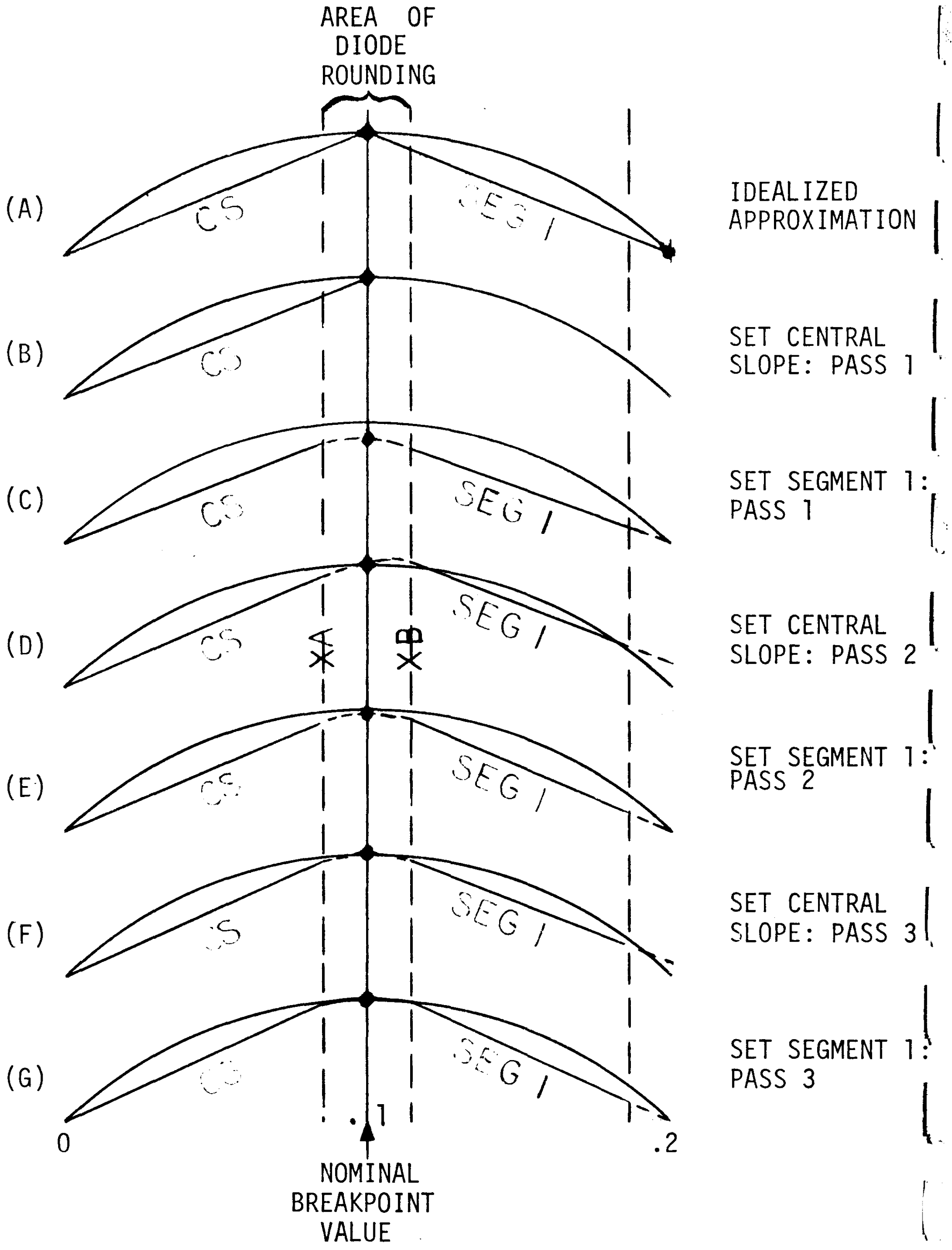


Figure 3.2 Diode Rounding: Setting f(X) at Nominal Breakpoint

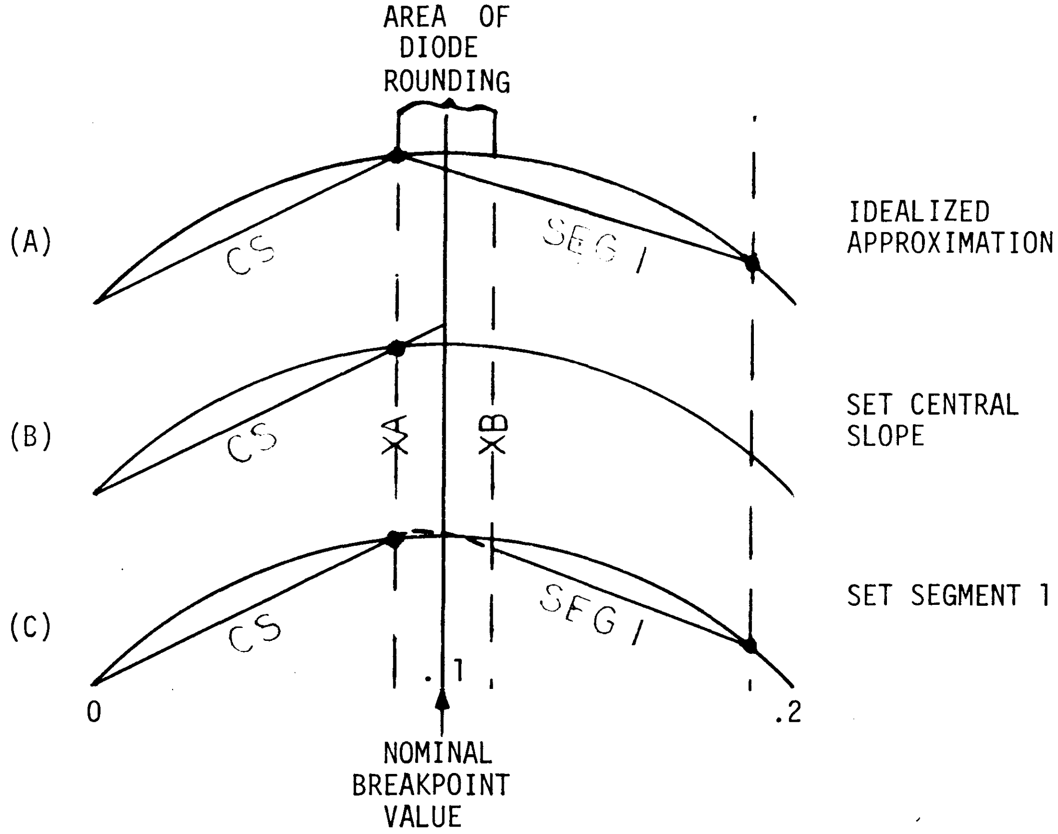


Figure 3.3 Diode Rounding: Setting f(X) Before Nominal Breakpoint

If it is not necessary to have accurate f(X) values at the nominal breakpoints, then accurate f(X) values can be achieved on the first setting of the DFG. Input values to the DFG are chosen before diode rounding begins, so that the f(X) set by one segment will not be affected by the adjustment of the succeeding segment. This is illustrated in Figure 3.3.

Figure 3.3(A) shows the desired curve and an idealized straight line approximation. Figure 3.3(B) shows f(X) being set by the central slope pot with an input X value before XA (i.e., before diode rounding begins). When the segment 1 pot is adjusted for the desired f(X) at the second breakpoint, the f(X) set by the central slope pot (represented by the dot) does not change position, since it is not in the area affected by diode rounding.

It is recommended that an input value be chosen somewhat before XA to be sure of avoiding effects of diode rounding. In the table made for the diode rounding of Figure 3.1 the input values were chosen to be .02 machine units before the XA observed in the maximum slope difference function. The reason for setting up that function now becomes clear: if the programmer knows where diode rounding begins for each breakpoint, he can avoid the effects of diode rounding on the accuracy of his f(X) values by choosing these values before the rounding begins whenever possible. This is the method used in Section 3.5 to set a fixed breakpoint approximation by the static method.

3.4 VDFG Set Up: Static Method, Gain Determination

The best dynamic response is obtained from the VDFG if the gain is set as low as possible to generate the desired function approximation. In order to estimate the necessary gain in the static method of DFG set up it is necessary to perform a number of calculations which are shown in Figure 3.4. These calculations establish the two segments between which the greatest slope change occurs. The gain can then be set to achieve the desired f(X) at this point, and thus the gain will not be set higher than necessary.

To simplify the programmer's task, copies of all the necessary tables with DFG set up information spaces left blank are provided at the back of this manual. Copies of these can be made by the programmer, and all he will have to do to set up a DFG function is to fill in the tables with the appropriate information, thus greatly simplifying his task.

COLUMN 1 INPUT X VALUES	COLUMN 2 BREAKPOINT POTS TO SET f(X) FOR X	COLUMN 3 FUNCTION VALUES FOR X INPUT	COLUMN 4 A	COLUMN 5 B	COLUMN 6 SEGMENT WITH UPPER VALUE AT X
$X_0 = 0$	PARALLAX	f_0	——	——	——
$X_1 = .1$	CENT. SLOPE	f_1	$f_1 - f_0$	$\lvert f_1 - f_0 \rvert$	CENT. SLOPE
$X_2 = .2$	1	f_2	$f_2 - 2f_1 + f_0$	$\lvert A \rvert$	1
$X_3 = .3$	2	f_3	$f_3 - 2f_2 + f_1$	$\lvert A \rvert$	2
$X_4 = .4$	3	f_4	$f_4 - 2f_3 + f_2$	$\lvert A \rvert$	3
$X_5 = .5$	4	f_5	$f_5 - 2f_4 + f_3$	$\lvert A \rvert$	4
$X_6 = .6$	5	f_6	$f_6 - 2f_5 + f_4$	$\lvert A \rvert$	5
$X_7 = .7$	6	f_7	$f_7 - 2f_6 + f_5$	$\lvert A \rvert$	6
$X_8 = .8$	7	f_8	$f_8 - 2f_7 + f_6$	$\lvert A \rvert$	7
$X_9 = .9$	8	f_9	$f_9 - 2f_8 + f_7$	$\lvert A \rvert$	8
$X_{10} = 1.0$	9	f_{10}	$f_{10} - 2f_9 + f_8$	$\lvert A \rvert$	9

NOTE: The B values of Column 5 are absolute values of the corresponding A value which is found in Column 4.

Figure 3.4 Tables for Gain Determination: Fixed Breakpoint

To set the DFG gain, fill in the information required by the tables in Figure 3.4. Note that the gain tables require that the f(X) values be filled in at the nominal breakpoint X values. This is required even if the actual input values will be chosen to occur before diode rounding rather than at the nominal breakpoint values. The reason for this procedure is to insure that the maximum slope change between two segments be determined for equally spaced segments. Once the tables have been filled in, proceed to set the DFG gain as follows:

1. Set switch 1 in the CAL position and switch 2 in the F position.

2. Address the DFG via the AD/FIVE addressing system.

3. The input to the DFG should come either from the D3.12 VDFG Calibrator or from an amplifier output patched into the CAL patchhole as show in Figure 1.5.

4. Set the DFG input to Ø and adjust the parallax pot until the DRM reading is zero.

5. Set the DFG input to 0.100 and adjust the central slope until the DRM reading is zero.

6. At 0.1 machine unit increments for the DFG input, set all the segment pots for zero DRM reading until the entire function has been zeroed.

7. Pick the segment whose Column 5 entry B is the largest. If $B \leq 0.2$, set the DFG gain full counterclockwise for minimum gain and set up the DFG (Sec. 3.5 or 3.6). If B > 2 the function cannot be generated by the E3.17 VDFG.)

8. Find the entry A in Column 4 which corresponds to the largest entry B in Column 5. If entry A is positive, turn the segment pot with the number in Column 2 full clockwise for maximum positive slope. If entry A is negative, turn the segment pot full counterclockwise for maximum negative slope.

9. Set the input voltage X equal to the corresponding value in Column 1. Then set the DFG output to a value of 1.2 times A by adjusting the gain pot until the desired result is obtained.

The DFG gain is now set by the static method.

3.5 VDFG Set Up: Static Method, Fixed Breakpoint

When setting the DFG in the fixed breakpoint mode the X input values will either be at the nominal breakpoints or at values chosen to occur before the diode rounding begins. If the X input values must be at the nominal breakpoints it will be necessary to use the iterative method of setting the function described in Section 3.4 and the discussion of Figure 3.2. It will be necessary to set the entire function, then make another pass through the entire function to accommodate the diode rounding, and then make a third pass if necessary. Three passes will usually suffice to get the desired degree of accuracy. If the input values are chosen before diode rounding begins (see the table in Section 3.3) the function can be set as desired on the first pass.

Fill in the appropriate values in the set up tables shown in Figure 3.5.

COLUMN 1 INPUT X VALUES	COLUMN 2 BREAKPOINT POTS TO SET f(X) FOR X	COLUMN 3 f(X) FOR X	COLUMN 4 SEGMENT WITH UPPER VALUE AT X
Ø	PARALLAX	$f_{Ø}$	———
X_1	CENT. SLOPE	f_1	CENT. SLOPE
X_2	1	f_2	1
X_3	2	f_3	2
X_4	3	f_4	3
X_5	4	f_5	4
X_6	5	f_6	5
X_7	6	f_7	6
X_8	7	f_8	7
X_9	8	f_9	8
1.Ø	9	$f_{1Ø}$	9

Figure 3.5 Tables for DFG Set Up: Fixed Breakpoint

Set the DFG as described below:

1. Set the input to 0.000 using the D3.12 Calibrator or an amplifier output as shown in Figure 1.5. Adjust the parallax pot labelled P for the desired output at X = 0.

2. Set the input to X_1. (This will either be the nominal breakpoint value or the point chosen to occur before diode rounding begins.) Adjust the central slope pot labelled CS for the desired output at X_1.

3. Proceed in this way for X_2 through X = 1.000, adjusting the appropriate pots as shown in the tables for the desired function values until the entire function is set.

If the input values have been chosen before diode rounding begins, the DFG is now set. If the input values are at the nominal breakpoints, it will be necessary to pass through the set up procedure again until the desired accuracy is achieved.

An example of the set up procedure is given below, using a DFG with the diode rounding characteristics of Figure 3.1, and using the XA minus .02 values in the table of Section 3.3 as input values.

Figure 3.6 shows the desired function of which an approximation is to be generated by the E3.17 VDFG.

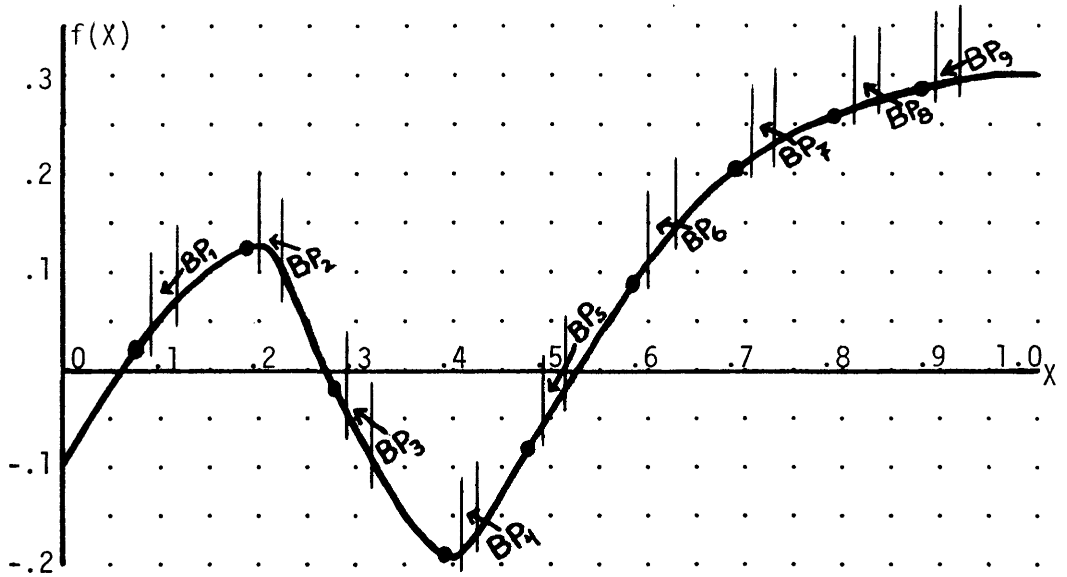


Figure 3.6 Function to be Programmed by the E3.17 VDFG

The dots on the curve of the function represent the XA minus .02 input values.

The breakpoints are labelled as BP_1 - BP_9.

The areas of diode rounding are indicated for each breakpoint by two vertical lines. The first line for each breakpoint represents XA, and the second line for each breakpoint represents XB. Diode rounding begins at XA and ends at XB. These areas of diode rounding were determined for this DFG by programming the maximum slope change function of Figure 3.1 and tabulating the observed XA and XB values in the table of Section 3.3.

Figure 3.7 shows the tables necessary to determine the gain for the desired function and the tables necessary to set up the desired function.

The DFG set up procedure is as follows:

1. The gain is determined with the tables in Figure 3.7.

   After gain determination, the DFG is set for a zero slope function.

TABLES FOR GAIN DETERMINATION: FIXED BREAKPOINT*

COLUMN 1	COLUMN 2	COLUMN 3	COLUMN 4	COLUMN 5	COLUMN 6
INPUT X VALUES	BREAKPOINT POTS TO SET f(X) FOR X	FUNCTION VALUES FOR X INPUT *	A	B	SEGMENT WITH UPPER VALUE AT X
Ø	PARALLAX	-.100	———	———	———
.1	CENT. SLOPE	.050	.150	.150	CENT. SLOPE
.2	1	.125	-.075	.075	1
.3	2	-.065	-.265	.265	2
.4	3	-.195	.060	.060	3
.5	4	-.050	.265	.265	4
.6	5	.110	.015	.015	5
.7	6	.215	-.055	.055	6
.8	7	.260	-.060	.060	7
.9	8	.290	-.015	.015	8
1.Ø	9	.300	-.020	.020	9

*WHEN USING TABLES TO DETERMINE GAIN FOR DFG Ø FOR USE IN THE VARIABLE BREAKPOINT MODE, SET THE NEGATIVE OF THE DESIRED B.P. X VALUES IN COLUMN 3.

TABLES FOR DFG SET UP: FIXED BREAKPOINT

COLUMN 1	COLUMN 2	COLUMN 3	COLUMN 4
INPUT X VALUES	BREAKPOINT POTS TO SET f(X) FOR X	f(X) FOR X	SEGMENT WITH UPPER VALUE AT X
Ø	PARALLAX	-.100	———
.07	CENT. SLOPE	.020	CENT. SLOPE
.18	1	.125	1
.27	2	-.020	2
.39	3	-.190	3
.47	4	-.080	4
.58	5	.095	5
.69	6	.205	6
.79	7	.255	7
.87	8	.285	8
1.Ø	9	.300	9

Figure 3.7
Example Tables: Fixed Breakpoint

2. Segments 2 and 4 (Column 6 of gain determination tables) have the largest Column 5 B entry. Since the gain determination procedure determines the gain necessary to set the maximum slope change between two segments, setting the gain for either segment 2 or segment 4 should achieve the desired result. Segment 2 will be used in this example. Since the Column 4 A entry for segment 2 is negative, segment pot 2 is turned full counterclockwise for maximum negative slope.

3. The input is set to 0.300. The gain pot is then adjusted until the DRM reads -0.3180 (i.e., 1.2 times -0.2650).

The gain has now been set. The DFG is set up as follows:

1. The input is set to 0.000. The parallax pot is adusted until the DRM output reads -0.1000.

2. The input is set to 0.070. The central slope pot is adjusted for a DRM output of 0.0200.

3. The input is set to 0.180. The segment 1 pot is adjusted for a reading of 0.1250.

4. The inputs are set in succession for the rest of the values shown in the DFG set up tables, and the desired outputs are set by the corresponding pots until the entire function has been set.

The DFG has now been set up in the fixed breakpoint mode through the static method.

3.6 VDFG Set Up: Static Method, Variable Breakpoint

In the variable breakpoint mode DFG Ø and DFG 1 are set up separately with switch 1 in the CAL position. When switch 1 is put in the OP position with switch 2 in the V position the two DFGs will together produce a function with variable breakpoints. DFG Ø determines the values of the variable breakpoints and is set up as follows:

1. Determine the gain necessary for DFG Ø using the tables and procedures as outlined in the sections on the fixed breakpoint mode. Set the gain as instructed.

2. Figure 3.8 shows the tables necessary for DFG Ø set up for the variable breakpoint mode. Note that these tables merely simplify the set up procedure and make it virtually foolproof. The use of the blank table sheets included at the back of this manual is greatly recommended. Fill in the appropriate values for the set up tables for DFG Ø.

3. Set up DFG Ø using the procedure as given in section 3.5. Note that the X values are given for the nominal breakpoints. The function generated by DFG Ø has a gradual slope change, and any necessary compensation for diode rounding is made during the set up of DFG 1. Set the desired function values for DFG Ø using Table 4, the input values of Table 1, and the pots shown in Table 2

TABLE 1	TABLE 2	TABLE 3	TABLE 4	TABLE 5
X INPUT VALUE FOR DFG Ø	BREAKPOINT POTS TO SET f(X): DFG Ø	DESIRED BREAKPOINT VALUES	NEGATIVE OF TABLE 3 { f(X) FOR DFG Ø }	SEGMENT WITH UPPER VALUE AT X
Ø	PARALLAX	Ø	(Ø)*	———
.1	CENT. SLOPE	BP_1	$-BP_1$ = Øf_1	CENT. SLOPE
.2	1	BP_2	$-BP_2$ = Øf_2	1
.3	2	BP_3	$-BP_3$ = Øf_3	2
.4	3	BP_4	$-BP_4$ = Øf_4	3
.5	4	BP_5	$-BP_5$ = Øf_5	4
.6	5	BP_6	$-BP_6$ = Øf_6	5
.7	6	BP_7	$-BP_7$ = Øf_7	6
.8	7	BP_8	$-BP_8$ = Øf_8	7
.9	8	BP_9	$-BP_9$ = Øf_9	8
1.Ø	9	(1.Ø)	(-1.Ø)**	9

Figure 3.8 Tables for DFG Ø Set Up: Variable Breakpoint

DFG Ø is now set up. Set DFG 1 as follows:

1. Figure 3.9 gives the tables necessary to set DFG 1 gain. Fill in the tables, determine and set the gain for DFG 1 in the same manner as described for the fixed breakpoint procedure. NOTE: For the nominal breakpoint input values used to set up DFG 1, the f(X) set by the segment pots should be the function value at the corresponding final variable breakpoint. For example, for a set up X input of 0.300, the desired f(X) is the function value of the final variable breakpoint curve at the third breakpoint, BP_3.

2. Figure 3.10 gives the tables necessary for DFG 1 set up. The same note applies as in step 1 above: for the nominal X input set up values the function values of the final variable breakpoint function are given at the corresponding variable breakpoints. The correspondences are tabulated below for clarity.

TABLE 1	TABLE 2	TABLE 3	TABLE 4	TABLE 5	TABLE 6
X INPUT VALUE FOR DFG 1	BREAKPOINT POTS TO SET f(X) AT XBP	f(X) AT XBP	A	B	SEGMENT WITH UPPER VALUE AT X
Ø	PARALLAX	$f_{\emptyset}$	——	——	——
.1	CENT. SLOPE	fBP_1	$fBP_1-f_{\emptyset}$	$\lvert fBP_1-f_{\emptyset}\rvert$	CENT. SLOPE
.2	1	fBP_2	$fBP_2-2fBP_1+f_{\emptyset}$	$\lvert A\rvert$	1
.3	2	fBP_3	$fBP_3-2fBP_2+fBP_1$	$\lvert A\rvert$	2
.4	3	fBP_4	$fBP_4-2fBP_3+fBP_2$	$\lvert A\rvert$	3
.5	4	fBP_5	$fBP_5-2fBP_4+fBP_3$	$\lvert A\rvert$	4
.6	5	fBP_6	$fBP_6-2fBP_5+fBP_4$	$\lvert A\rvert$	5
.7	6	fBP_7	$fBP_7-2fBP_6+fBP_5$	$\lvert A\rvert$	6
.8	7	fBP_8	$fBP_8-2fBP_7+fBP_6$	$\lvert A\rvert$	7
.9	8	fBP_9	$fBP_9-2fBP_8+fBP_7$	$\lvert A\rvert$	8
1.Ø	9	$f_{1\emptyset}$	$f_{1\emptyset}-2fBP_9+fBP_8$	$\lvert A\rvert$	9

Figure 3.9 Tables for Gain Determination: Variable Breakpoint, DFG 1

X INPUT FOR SET UP	FUNCTION VALUE AT VARIABLE BREAKPOINT
Ø	Function value at X = 0.000.
.1	Function value at first var. BP.
.2	Function value at second var. BP.
.3	Function value at third var. BP.
.4	Function value at fourth var. BP.
.5	Function value at fifth var. BP.
.6	Function value at sixth var. BP.
.7	Function value at seventh var. BP.
.8	Function value at eighth var. BP.
.9	Function value at ninth var. BP.
1.Ø	Function value at X = 1.000.

3. Set the input to 0.000 and set the parallax pot for a DRM output of $1f_{\emptyset}$, which is the desired output of DFG 1 at X = Ø.

4. Set the input to 0.100 and set the central slope pot for a DRM reading of $1fBP_1$, which is the desired DFG 1 output at the first BP.

5. Set the input to 0.200 and set the segment 1 pot for a reading of $1fBP_2$, which is the desired DFG 1 output at the second BP.

6. Continue in this manner until the entire function is set.

NOTE: The X input to the DFG must be inverted before switching to OP. In the variable breakpoint mode it is advisable to use the iterative method as a matter of course in setting DFG 1. When the E3.17 switch 1 is set to OP the function should be examined to see if further minor adjustments are necessary to achieve desired accuracy. DFG Ø and DFG 1 are now programmed in the variable breakpoint mode using the static set up method.

TABLE 1	TABLE 2	TABLE 3	TABLE 4
X INPUT VALUE FOR DFG 1	BREAKPOINT POTS TO SET f(X): DFG 1	f(X) AT XBP	SEGMENT WITH UPPER VALUE AT X
Ø	PARALLAX	$1f_{\emptyset}$	——
.1	CENT. SLOPE	$1fBP_1$	CENT. SLOPE
.2	1	$1fBP_2$	1
.3	2	$1fBP_3$	2
.4	3	$1fBP_4$	3
.5	4	$1fBP_5$	4
.6	5	$1fBP_6$	5
.7	6	$1fBP_7$	6
.8	7	$1fBP_8$	7
.9	8	$1fBP_9$	8
1.Ø	9	$1fBP_{1\emptyset}$	9

Figure 3.10 Tables for DFG 1 Set Up: Variable Breakpoint

An example set up is given below for the function shown in Figure 3.11. The variable breakpoints desired are indicated by a vertical line, and the lighter lines following the curve are the idealized variable breakpoint straight line segments of the approximation which is desired.

1. Figure 3.13 gives the gain determination and set up tables for DFG Ø. The segment with the greatest B entry in column 5 is segment 8. The DFG is programmed for the zero slope function. Since the Column 4 entry for segment 8 is negative, the segment 8 pot is turned full counterclockwise. The input is set to 0.900, and the gain pot is adjusted for a DRM output of -0.1920. The DFG Ø gain is now set. (-0.1920 is 1.2 times -0.1600.)

2. The input is set to 0.000, and the parallax pot is adjusted for a DRM output of 0.0000.

3. The input is set to 0.100 and the central slope pot is set for a DRM reading of -0.1900.

4. The input is set to 0.200 and the segment 1 pot is set for a DRM reading of -0.3000.

5. The procedure is continued until the input is 1.000, and the

segment 9 pot is adjusted for a reading of -1.000.

Check the segment values and adjust if necessary.

DFG Ø is now set up, and with switch 2 in the F position and switch 1 in OP its output would be the curve shown in Figure 3.12.

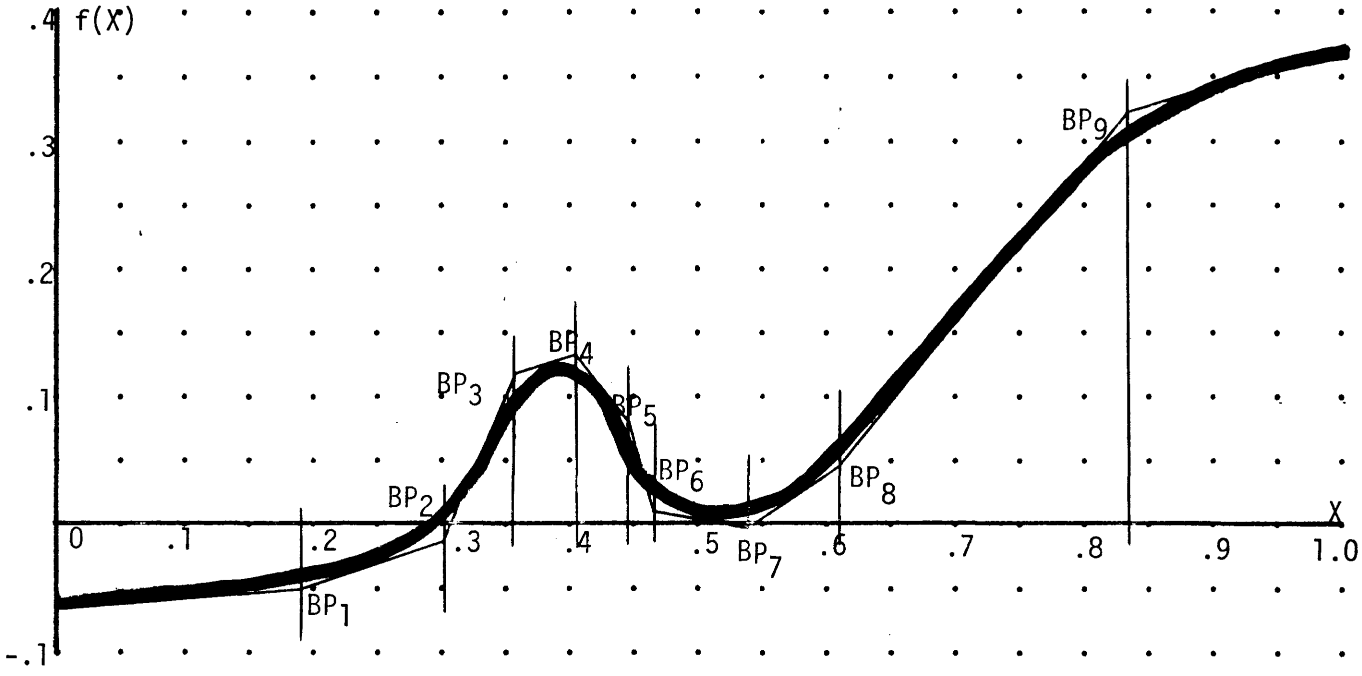


Figure 3.11 Function to be Programmed in the Variable Breakpoint Mode

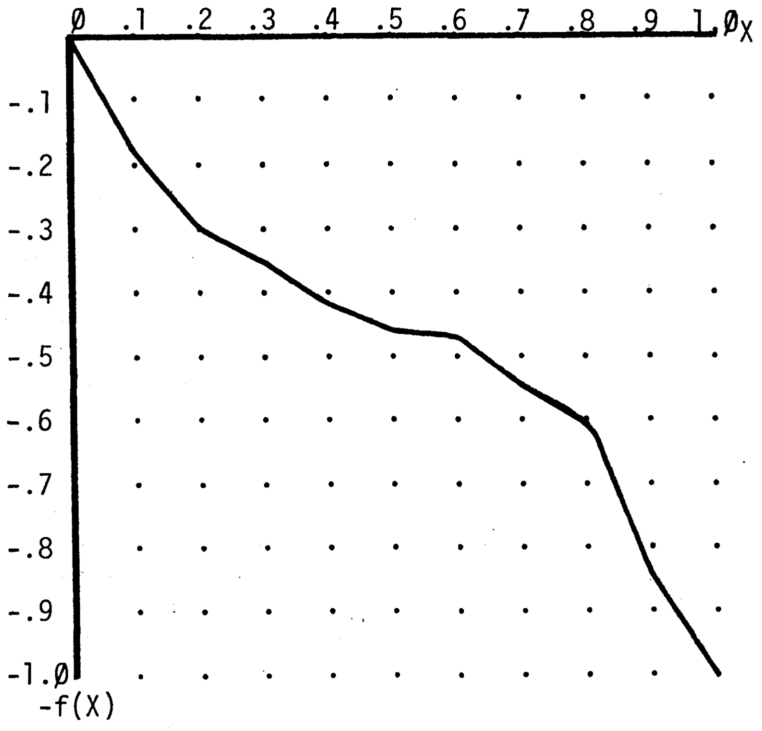


Figure 3.12 DFG Ø Output with Switch 2 in F Position

TABLES FOR GAIN DETERMINATION: FIXED BREAKPOINT

COLUMN 1 INPUT X VALUES	COLUMN 2 BREAKPOINT POTS TO SET f(X) FOR X	COLUMN 3 FUNCTION VALUES FOR X INPUT*	COLUMN 4 A	COLUMN 5 B	COLUMN 6 SEGMENT WITH UPPER VALUE AT X
Ø	PARALLAX	Ø	———	———	———
.1	CENT. SLOPE	-.190	-.190	.190	CENT. SLOPE
.2	1	-.300	.080	.080	1
.3	2	-.360	.050	.050	2
.4	3	-.415	.005	.005	3
.5	4	-.445	.025	.025	4
.6	5	-.465	.010	.010	5
.7	6	-.540	-.055	.055	6
.8	7	-.610	.005	.005	7
.9	8	-.840	-.160	.160	8
1.Ø	9	-1.00	.070	.070	9

*WHEN USING TABLES TO DETERMINE GAIN FOR DFG Ø FOR USE IN THE VARIABLE BREAKPOINT MODE, SET THE NEGATIVE OF THE DESIRED B.P. X VALUES IN COLUMN 3.

TABLES FOR DFG SET UP: VARIABLE BREAKPOINT, DFG Ø

TABLE 1	TABLE 2	TABLE 3	TABLE 4	TABLE 5
X INPUT VALUE FOR DFG Ø	BREAKPOINT POTS TO SET f(X): DFG Ø	DESIRED BREAKPOINT VALUES	NEGATIVE OF TABLE 3 { f(X) FOR DFG Ø }	SEGMENT WITH UPPER VALUE AT X
Ø	PARALLAX	Ø	Ø	———
.1	CENT. SLOPE	.190	-.190	CENT. SLOPE
.2	1	.300	-.300	1
.3	2	.360	-.360	2
.4	3	.415	-.415	3
.5	4	.445	-.445	4
.6	5	.465	-.465	5
.7	6	.540	-.540	6
.8	7	.610	-.610	7
.9	8	.840	-.840	8
1.Ø	9	(1.0)	(-1.0)**	9

Figure 3-3
Example
Tables
For DFG
In Variable
Breakpoint
Mode

TABLES FOR GAIN DETERMINATION: VARIABLE BREAKPOINT, DFG 1

TABLE 1	TABLE 2	TAB.3	TAB.4	TAB.5	TABLE 6
X INPUT VALUE FOR DFG 1	BREAKPOINT POTS TO SET f(X) AT XBP	f(XBP)	A	B	SEGMENT WITH UPPER VALUE AT X
Ø	PARALLAX	-.055	——	——	——
.1	CENT. SLOPE	-.040	-.015	.015	CENT.SLOPE
.2	1	.010	.035	.035	1
.3	2	.190	.130	.130	2
.4	3	.120	-.250	.250	3
.5	4	.060	.010	.010	4
.6	5	.030	.030	.030	5
.7	6	.020	.020	.020	6
.8	7	.060	.050	.050	7
.9	8	.305	.205	.205	8
1.Ø	9	.365	-.180	.180	9

TABLES FOR DFG SET UP: VARIABLE BREAKPOINT, DFG 1

TABLE 1	TABLE 2	TABLE 3	TABLE 4
X INPUT VALUE FOR DFG 1	BREAKPOINT POTS TO SET f(X): DFG 1	f(X) AT XBP	SEGMENT WITH UPPER VALUE AT X
Ø	PARALLAX	-.055	——
.1	CENT. SLOPE	-.040	CENT. SLOPE
.2	1	.010	1
.3	2	.190	2
.4	3	.120	3
.5	4	.060	4
.6	5	.030	5
.7	6	.020	6
.8	7	.060	7
.9	8	.305	8
1.Ø	9	.365	9

Figure 3.14
Example
Tables
For DFG 1
In Variable
Breakpoint
Mode

6. The gain and set up tables for DFG 1 in the variable breakpoint mode are shown in Figure 3.14. The B entry in Table 5 is largest for segment 3. All the DFG 1 segment slopes are set to zero. Since entry A for segment 3 is negative, the segment 3 pot is turned full counterclockwise. An input of 0.400 is set into the DFG. The gain pot is adjusted until the DRM reads -0.3000. The gain for DFG 1 is now set. (-0.3000 is 1.2 times -0.2500.)

7. An input of 0.000 is set into the DFG. The parallax pot is set for a DRM reading of -0.0550.

8. An input of 0.100 is set into the DFG. The central slope pot is set for a reading of -0.0400.

9. An input of 0.200 is set into the DFG. The segment 1 pot is set for a reading of 0.0100.

10. The rest of the segment pots are set in a similar manner until an input of 1.000 is reached and the segment 9 pot is set for a reading of 0.3650.

At this point the operator should return to step 7 and repeat the rest of the set up procedure until the desired accuracy is achieved from the DFG 1 output.

Now the actual X input is inverted and patched into the X input hole.

Switch 1 is set to OP and switch 2 is set to V.

The two DFGs are now operating in the variable breakpoint mode. The final function should be examined to see if any minor changes in any of the segments are necessary.

4.0 VDFG OPERATION IN FOUR QUADRANTS WITH TEN AND TWENTY SEGMENTS

Each DFG in the E3.17 has adjustable segments for $Ø \leq X \leq +1.Ø$. Generally this range will not be sufficient for simulation problems. This section describes methods of achieving ten and twenty segment functions in the fixed and variable breakpoint modes for the input range $-1.Ø \leq X \leq +1.Ø$.

Figure 4.0 shows the patching configuration necessary for a ten segment fixed breakpoint approximation for a full range of input values from -1.Ø to +1.Ø. Two variations are given.

Figure 4.1 shows the means of achieving a full range twenty segment approximation in the fixed breakpoint mode.

Note that two configurations are given. The upper circuit does not have a breakpoint at $X = Ø$, so that the central slopes of the two DFGs act as one segment. The advantage of this configuration lies in the fact that less hardware is needed to achieve the approximation. The true twenty segment approximation with a breakpoint at $X = Ø$ is shown in the lower configuration in Figure 4.1

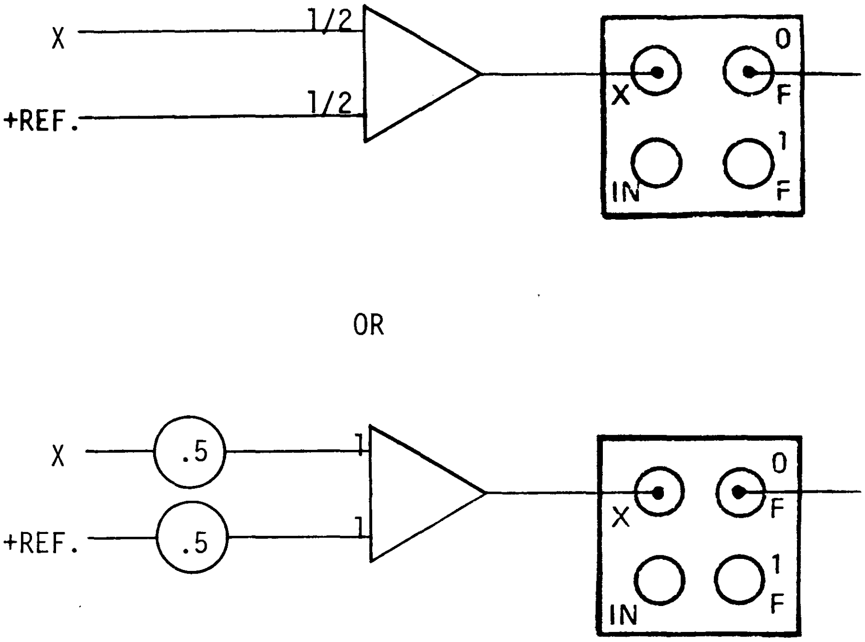
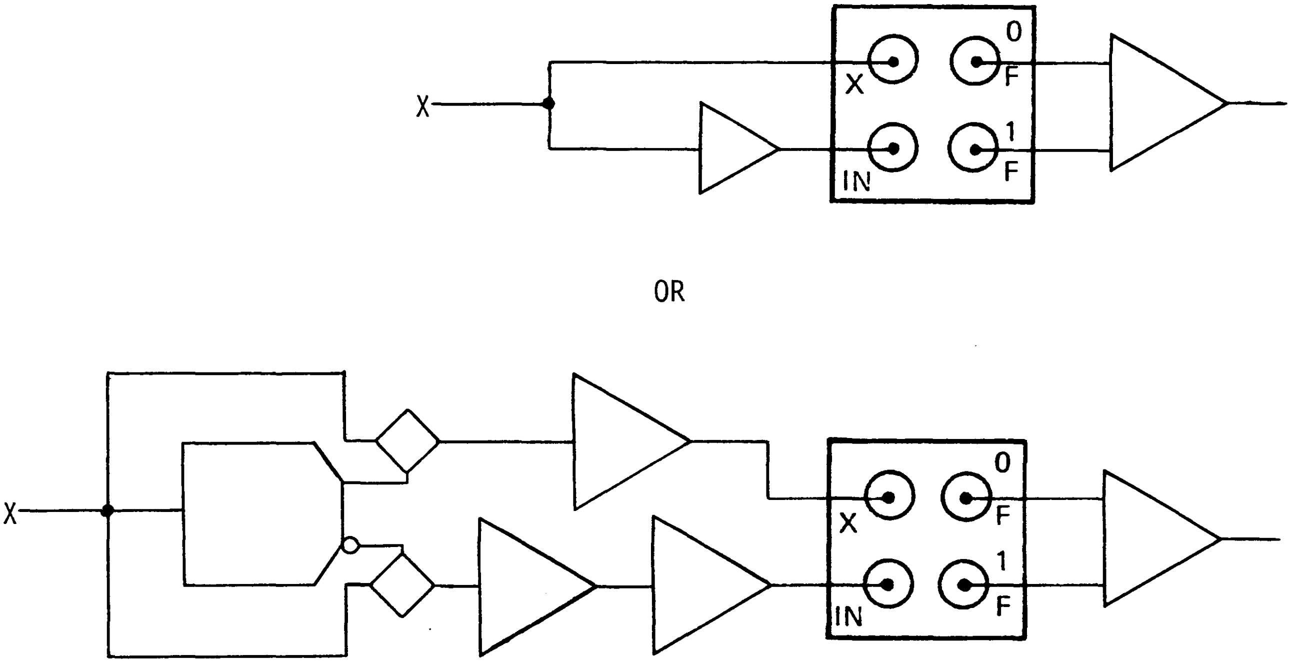


Figure 4.0 Ten Segment Fixed Breakpoint Full Range Configuration



NOTE: THE UPPER CIRCUIT DOES NOT HAVE INDEPENDENT CENTRAL SLOPES, I.E., THERE IS NOT A BREAKPOINT AT X = 0. THE CENTRAL SLOPES OF DFG Ø AND DFG 1 ACT AS ONE SEGMENT. THE LOWER CIRCUIT DOES HAVE A BREAKPOINT AT X = 0, AND THE CENTRAL SLOPES ARE INDEPENDENT.

Figure 4.1 Twenty Segment Fixed Breakpoint Full Range Configuration

Figure 4.2 shows the ten segment full range configuration for the variable breakpoint mode. Two versions are given.

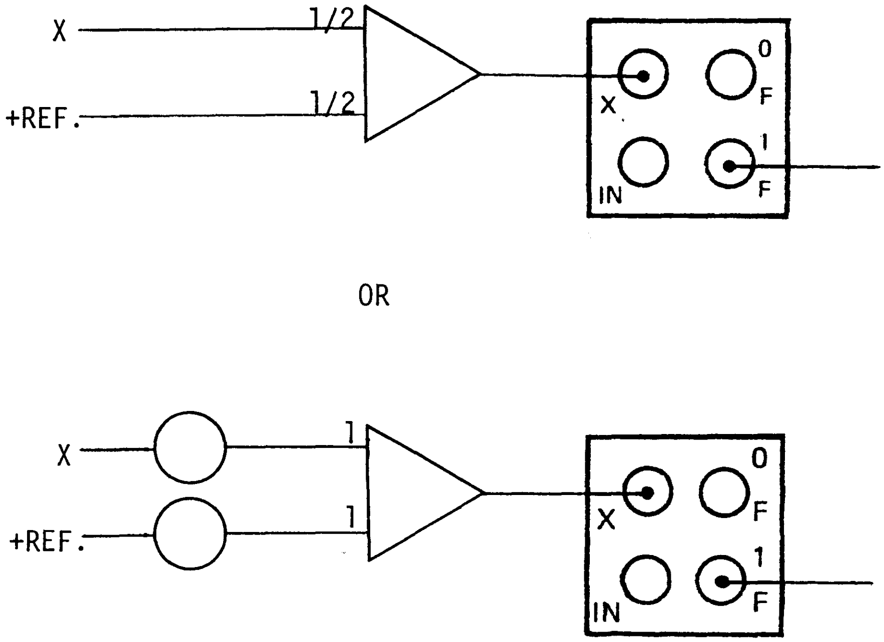


Figure 4.2 Ten Segment Variable Breakpoint Full Range Configuration

Figure 4.3 shows the twenty segment full range configuration for the variable breakpoint mode. There is a breakpoint at X = Ø.

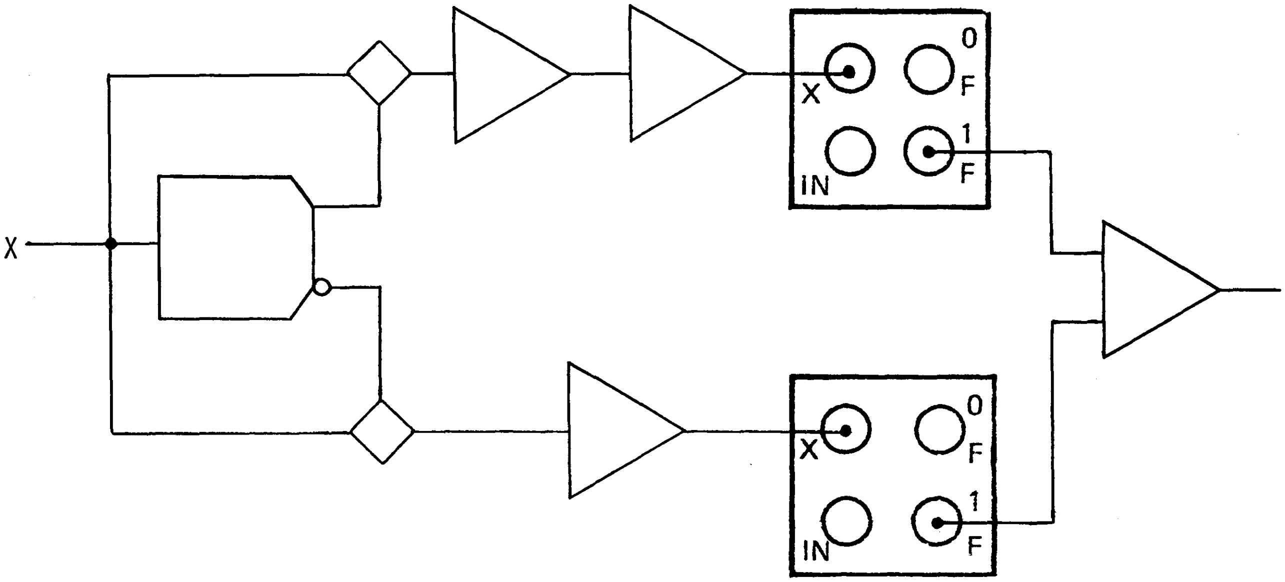


Figure 4.3 Twenty Segment Variable Breakpoint Full Range Configuration

BLANK TABLES FOR DFG GAIN DETERMINATION AND SET UP PROCEDURES

TABLES FOR GAIN DETERMINATION: FIXED BREAKPOINT*

COLUMN 1	COLUMN 2	COLUMN 3	COLUMN 4	COLUMN 5	COLUMN 6
INPUT X VALUES	BREAKPOINT POTS TO SET f(X) FOR X	FUNCTION VALUES FOR X INPUT *	A	B	SEGMENT WITH UPPER VALUE AT X
Ø	PARALLAX		———	———	———
.1	CENT. SLOPE				CENT. SLOPE
.2	1				1
.3	2				2
.4	3				3
.5	4				4
.6	5				5
.7	6				6
.8	7				7
.9	8				8
1.Ø	9				9

*WHEN USING TABLES TO DETERMINE GAIN FOR DFG Ø FOR USE IN THE VARIABLE BREAKPOINT MODE, SET THE NEGATIVE OF THE DESIRED B.P. X VALUES IN COLUMN 3.

TABLES FOR DFG SET UP: FIXED BREAKPOINT

COLUMN 1	COLUMN 2	COLUMN 3	COLUMN 4
INPUT X VALUES	BREAKPOINT POTS TO SET f(X) FOR X	f(X) FOR X	SEGMENT WITH UPPER VALUE AT X
Ø	PARALLAX		———
	CENT. SLOPE		CENT. SLOPE
	1		1
	2		2
	3		3
	4		4
	5		5
	6		6
	7		7
	8		8
1.Ø	9		9

TABLES FOR DFG SET UP: VARIABLE BREAKPOINT, DFG Ø

TABLE 1	TABLE 2	TABLE 3	TABLE 4	TABLE 5
X INPUT VALUE FOR DFG Ø	BREAKPOINT POTS TO SET f(X): DFG Ø	DESIRED BREAKPOINT VALUES	NEGATIVE OF TABLE 3 { f(X) FOR DFG Ø }	SEGMENT WITH UPPER VALUE AT X
Ø	PARALLAX	Ø	Ø	———
.1	CENT. SLOPE			CENT. SLOPE
.2	1			1
.3	2			2
.4	3			3
.5	4			4
.6	5			5
.7	6			6
.8	7			7
.9	8			8
1.Ø	9	(1.Ø)	-1.Ø	9

TABLES FOR GAIN DETERMINATION: VARIABLE BREAKPOINT, DFG 1

TABLE 1	TABLE 2	TAB.3	TAB. 4	TAB.5	TABLE 6
X INPUT VALUES FOR DFG 1	BREAKPOINT POTS TO SET f(X) AT XBP	f(XBP)	A	B	SEGMENT WITH UPPER VALUE AT X
Ø	PARALLAX		—	—	—
.1	CENT. SLOPE				CENT.SLOPE
.2	1				1
.3	2				2
.4	3				3
.5	4				4
.6	5				5
.7	6				6
.8	7				7
.9	8				8
1.Ø	9				9

TABLES FOR DFG SET UP: VARIABLE BREAKPOINT, DFG 1

TABLE 1	TABLE 2	TABLE 3	TABLE 4
X INPUT VALUES FOR DFG 1	BREAKPOINT POTS TO SET f(X): DFG 1	f(X) AT XBP	SEGMENT WITH UPPER VALUE AT X
Ø	PARALLAX		—
.1	CENT. SLOPE		CENT. SLOPE
.2	1		1
.3	2		2
.4	3		3
.5	4		4
.6	5		5
.7	6		6
.8	7		7
.9	8		8
1.Ø	9		9