MINC-11

Book 1:
Introduction to MINC

November 1978

Book 1 introduces the capabilities and components of MINC and provides instructions for using MINC.

Order Number AA-D571A-TC

MINC-11

VERSION 1.0

To order additional copies of this document, contact the Software Distribution Center, Digital Equipment Corporation, Maynard, Massachusetts 01754

digital equipment corporation · maynard, massachusetts

First Printing, November 1978

The postage-prepaid READER'S COMMENTS form on the last page of this document requests the user's critical evaluation to assist us in preparing future documentation.

4/79-14

CONTENTS

FIGURES

PART 1

INTRODUCING MINC

CHAPTER 1
CAPABILITIES

MINC, the Modular INstrument Computer, is a portable, expandable, and versatile system designed for performing computations, producing graphic displays, monitoring laboratory processes, controlling experiments, and acquiring data.

MINC is a powerful tool for users at any level of computer experience. You do not need prior computer training to use MINC productively. The books accompanying MINC explain and demonstrate how to use MINC's capabilities. If you have used a computer before, you will appreciate MINC's straightforward, logical adaptation to your applications.

PORTABILITY

The MINC system is compact and portable. MINC consists of two physical parts: a terminal and a cart on wheels. The whole system can conveniently be wheeled from lab to office to classroom for different users and different applications.

When the system is being moved, the terminal rests on the low shelf of the MINC cart. When the system is being used, the terminal rests on a table or counter.

MINC does not require any special wiring or environment. MINC can use either 12 amps at 115 volts ac or 6 amps at 230 volts ac. MINC operates under normal room conditions. (That is, within a temperature range from 15° to 32° C and a relative humidity range from 20 to 80 percent.)

EXPANDABILITY

The MINC system can be expanded by the addition of lab modules. The MINC chassis contains eight empty slots into which you can put any valid combination of MINC lab modules. The lab modules connect most analog and digital laboratory instruments to MINC. The set of modules in the chassis determines MINC's capabilities.

Configuring a MINC System

The process of adding the lab modules necessary to perform the data transfer and control operations is called system configuration. The modules make the configuration process simple and reliable. The modules are keyed so that only electrically compatible modules can be placed next to one another in the chassis to form a valid configuration. The top panel of each module has a diagram that represents the function of the module. These diagrams combine to form an overall picture of the system's function as the modules are plugged into the chassis to configure the system.

MINC Configuration Cards

The diagrams on the top panels of the MINC modules are duplicated on a set of cards called MINC configuration cards (see Figure 5). Each module is represented by a card, which carries a diagram showing the function of the module and its permissible relationships with other modules. You can use these configuration cards to design configurations and to confirm viable interaction among the modules selected for a configuration.

The edges of some playing cards contain the letter "N" or "O". If two cards spell the word "NO," then the modules corresponding to the cards cannot be placed side by side in the chassis.

An example of a workable configuration that permits the collection of analog data under control of the clock module is shown by the sequence of configuration cards in Figure 1.

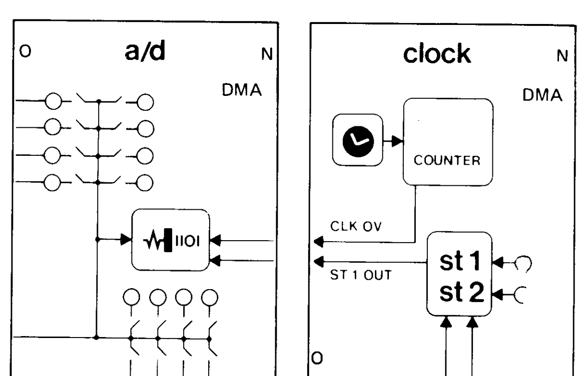


Figure 1. Valid Configuration Example 1

An example of a configuration that permits MINC both to collect analog values and to send data to analog instruments under control of the clock module is given by the sequence of configuration cards in Figure 2.

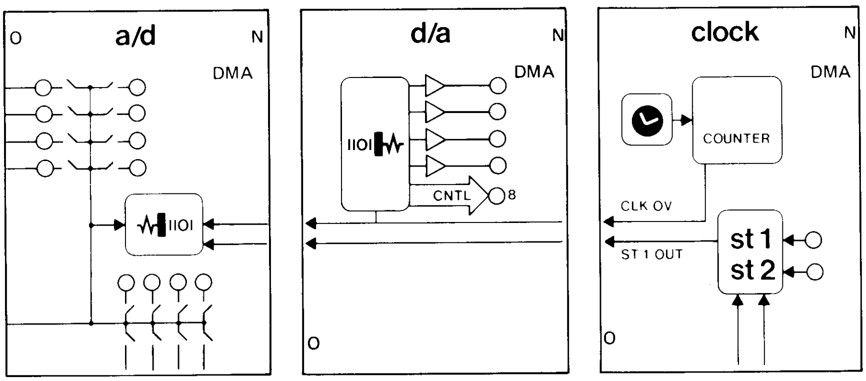


Figure 2. Valid Configuration Example 2

An example of a configuration that allows you both to send data to and to receive data from digital instruments is given by the sequence of configuration cards in Figure 3.

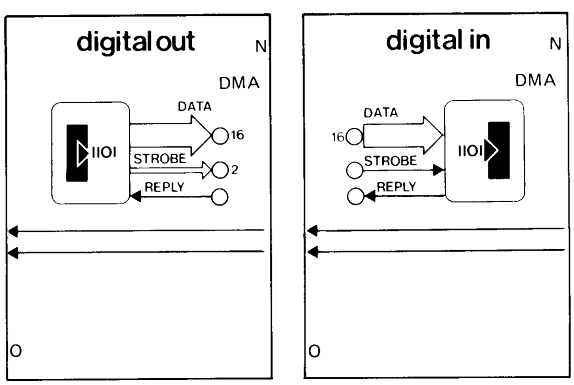


Figure 3. Valid Configuration Example 3

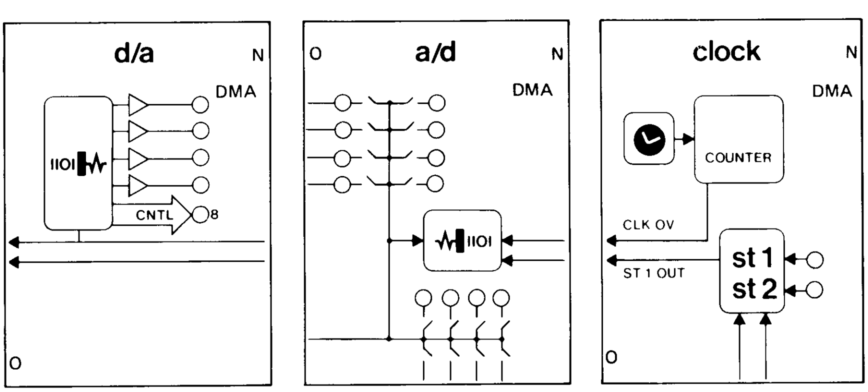


Figure 4. Invalid Configuration Example

An example of an unworkable pairing is shown in Figure 4.

Observe that the letters spell "NO," indicating that a digital-to-analog (D/A) converter cannot be placed to the left of an analog-to-digital (A/D) converter in the chassis.

A set of the MINC configuration cards is presented on pages 8 and 9. The set of cards is reproduced on heavy perforated paper at the back of this book. These cards can be punched out and used to design possible configurations. If that page has already been punched out by another reader, you can copy the pages 8 and 9 to obtain your own set of configuration cards.

VERSATILITY

MINC has a wide variety of uses. The standard system can be used for:

- Computations
- Graphic displays
- Communication with IEEE-standard instruments
- Communication with RS-232 ASCII instruments
- Data collection
- Data storage
- Data reduction
- Data analysis
- Data display

With the addition of lab modules, MINC can perform the following tasks for analog and digital instruments:

- Data transfer
- Instrument control

Using the above capabilities, you can program applications, like:

- Laboratory automation
- Process control

MINC is accompanied by a set of eight books. These books contain tutorial, explanatory, and reference information. The tutorial information teaches the use of the system. The explanatory information describes the concepts necessary for the understanding of a particular feature of MINC and provides structured practice in the use of that feature. The reference information allows you to quickly and conveniently locate information on a specific topic.

MINC BASIC

You communicate with MINC in an interactive high-level programming language called MINC BASIC that uses simple statements and convenient mathematical notation. By means of MINC BASIC, you can:

Perform calculations.

Combine a sequence of operations into a *program,* which you can use over and over.

Conveniently change the program using the editing capability.

Create and maintain large collections of data values.

MINC BASIC is described in the following books:

Book 2: MINC Programming Fundamentals teaches the MINC BASIC language and the use of the MINC system.

Book 3: MINC Programming Reference is an alphabetical reference book for the concepts and commands of MINC BASIC.

Graphics

MINC's terminal can produce graphic displays. You control the presentation of data on the video screen with MINC's graphic routines, which allow you to do the following:

Display graphs as discrete points, points connected by lines, or histograms (bar graphs).

Display two graphs simultaneously.

Move a graph continuously from right to left to observe the time-variant nature of the graphed data (like a strip-chart recorder).

Position a graph any place on the screen and change the size of the graph without affecting its shape.

Add special text features, such as underlining, boldface, or flashing characters.

The MINC system can read the data to be displayed from your laboratory instruments (with the lab module routines) or it can compute new data internally. Thus, data can either be acquired or produced under computer control.

The process by which you display data under computer control is called graphic programming. *Book 4: MINC Graphic Programming* explains the concepts of graphic programming and provides an alphabetical reference for the graphic routines.

Data Transfer and Instrument Control

You control communication between MINC and the external instruments connected to it using the following devices:

- MINC's IEEE bus interface, which connects instruments that conform to IEEE standard 488-1975. The MINC IEEE bus is included in every system.
- MINC's serial ASCII controller, which connects instruments that use serial transmission of ASCII characters in RS-232 protocol. The MINC Serial ASCII controller is included in every system.
- MINC's special lab modules, which connect a wide variety of analog or digital instruments. MINC's lab modules are purchased as options.

Data Transfer MINC's data transfer capability lets you control the transfer of information from instruments to MINC (input) and from MINC to instruments (output).

- You can collect data values and status information from IEEE-compatible instruments or send data values and control information to one or more such instruments using MINC's IEEE bus interface.
- You can collect ASCII character data and status values from ASCII instruments using RS-232 protocol or send character information to these instruments using MINC's serial ASCII interfaces.
- You can sample data values, accumulate data values (signal averaging), and generate histograms of analog input values using MINC's optional analog-to-digital converter.

- You can send data values to analog instruments using MINC's optional digital-to-analog converter.

- You can send or receive digital values using MINC's optional digital input and output modules.

Instrument Control MINC's instrument control capability lets you synchronize data transfer processes with events in MINC or in the lab environment. The sources of control information can be internal to MINC or in the external instrument environment. The following capabilities are provided:

- You can control the transfer of information from one IEEE-488 standard instrument to one or more other such instruments.

- You can schedule events (for example, starting input transfers) to occur at certain times and can coordinate schedules for different events (for example, stopping one transfer and immediately starting another).

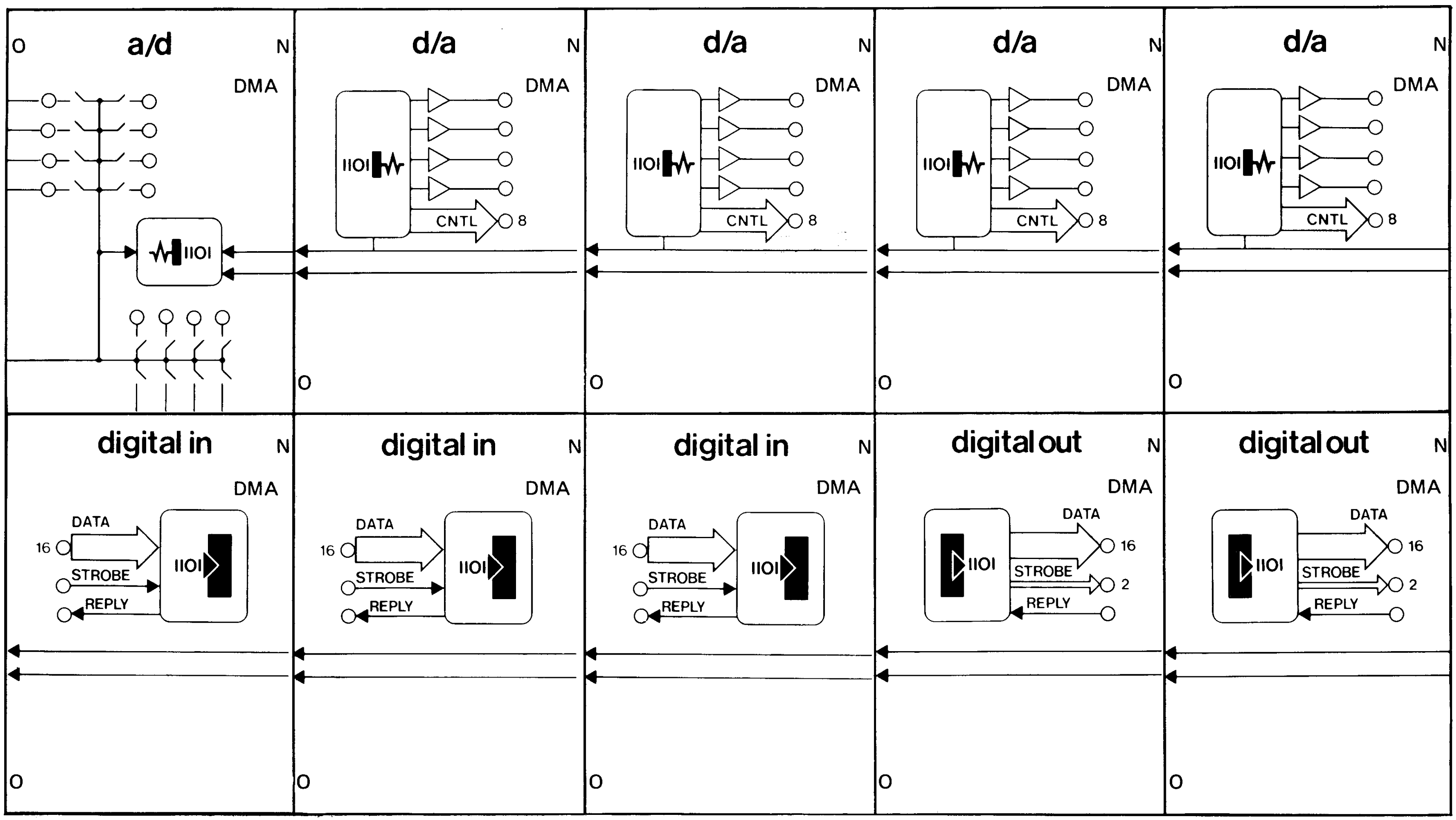
- You can start or stop transfers based on control signals from the instruments themselves. With the appropriate sensors and instruments connected to lab modules, you can monitor and control the environment in which the instrument operates.

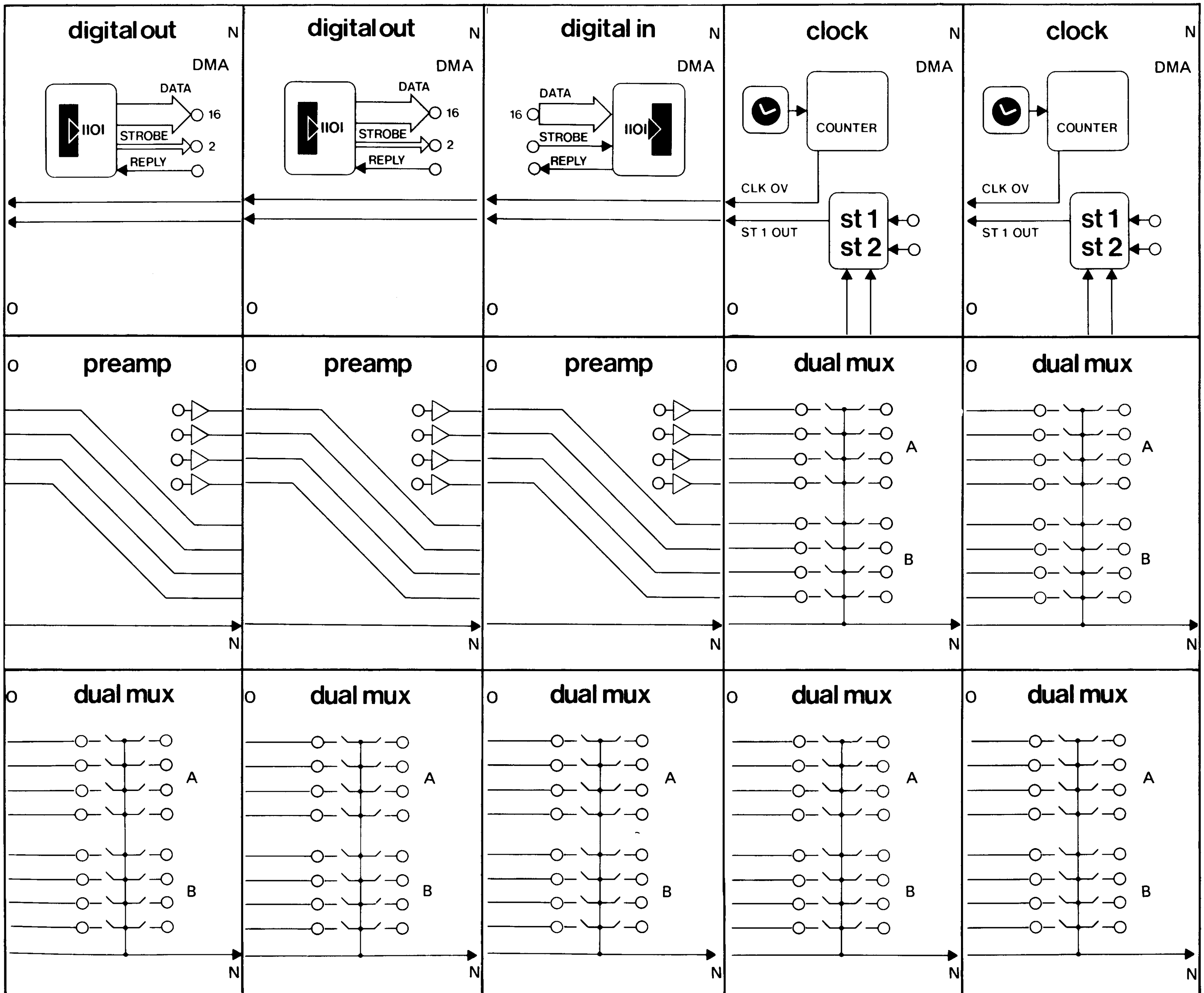
The process by which you transfer data and control information to IEEE-compatible instruments is called IEEE bus programming. *Book 5: IEEE Bus Programming* explains the concepts required for programming IEEE-488 standard instruments and provides an alphabetical reference to the IEEE bus routines.

The process by which you transfer data values and control information between MINC and laboratory instruments is called lab module programming. *Book 6: Lab Module Programming* describes the concepts required for lab module programming and provides an alphabetical reference to the lab module routines.

Book 7: Working with MINC Devices describes the physical and electrical properties of MINC. This book helps you connect MINC to laboratory instruments and identify problems resulting from faulty connections.

Figure 5. MINC Configuration Cards




digital out
N
DMA
DATA
16
IIOI
STROBE
2
REPLY
O
digital out
N
DMA
DATA
16
IIOI
STROBE
2
REPLY
O
digital in
N
DMA
DATA
16
STROBE
REPLY
IIOI
O
clock
N
DMA
COUNTER
CLK OV
ST 1 OUT
st 1
st 2
O
clock
N
DMA
COUNTER
CLK OV
ST 1 OUT
st 1
st 2
O
preamp
O
N
preamp
O
N
preamp
O
N
dual mux
O
A
B
N
dual mux
O
A
B
N
dual mux
O
A
B
N
dual mux
O
A
B
N
dual mux
O
A
B
N
dual mux
O
A
B
N
dual mux
O
A
B
N

CHAPTER 2
COMPONENTS

MINC consists of a terminal and a cart. The following sections describe these components.

MINC TERMINAL

You communicate with MINC through its terminal. The terminal consists of a video screen and a separate keyboard (Figure 6). You can position the keyboard next to the screen if you are using it alone or you can place the keyboard some distance from the screen unit to allow several people to view the screen simultaneously.

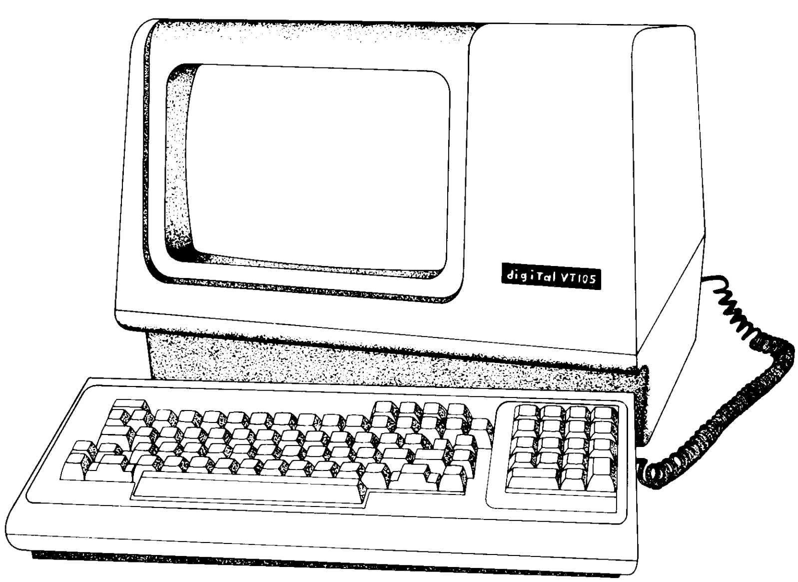


Figure 6. The MINC Terminal

To tell MINC to perform calculations, collect data, or monitor processes, you type commands on the keyboard of the terminal. MINC displays each character you type on the terminal screen.

MINC does not interpret the command you are typing until you signal that the command is complete by pressing the RETURN key. Thus, if you make a typing mistake before you press the RETURN key, you can correct the error. When you press the RETURN key, MINC responds with text on the screen that either acknowledges the command or indicates an error in the command. If the command is complete and correct, MINC performs the actions requested and notifies you when the actions have been completed.

A detailed drawing of the keyboard and some instructions on its use are given in Chapter 3. More detailed instructions can be found in *Book 2: MINC Programming Fundamentals*.

Graphic Routines

You can use the terminal to display graphs and specially formatted text. In the MINC system, the tools for graphic programming are called the *graphic routines*. To display a set of discrete points, you use the GRAPH routine. An example of a graph produced by this routine is shown in Figure 7.

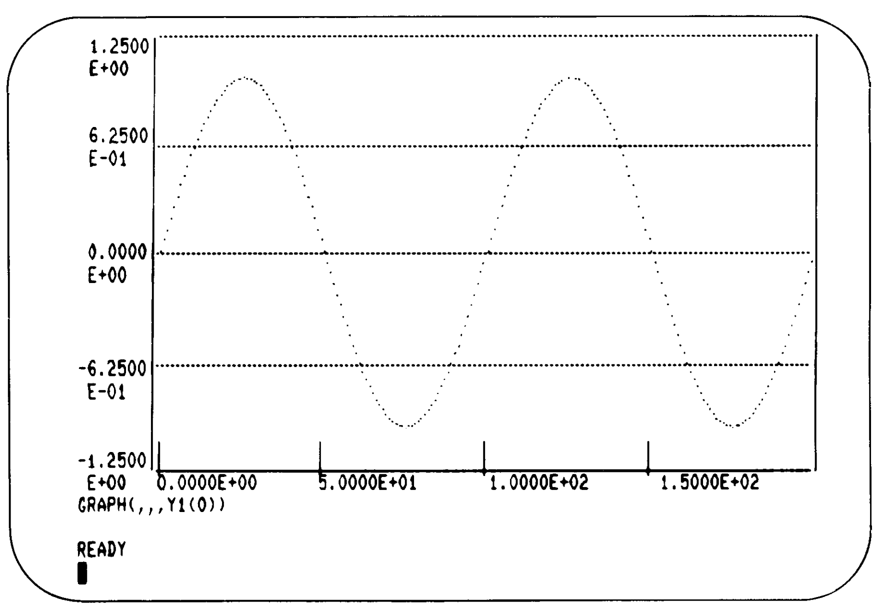


Figure 7. Sample Graph

To display a set of rectangular bars, you use the BARGRAPH routine. An example of a bargraph is shown in Figure 8.

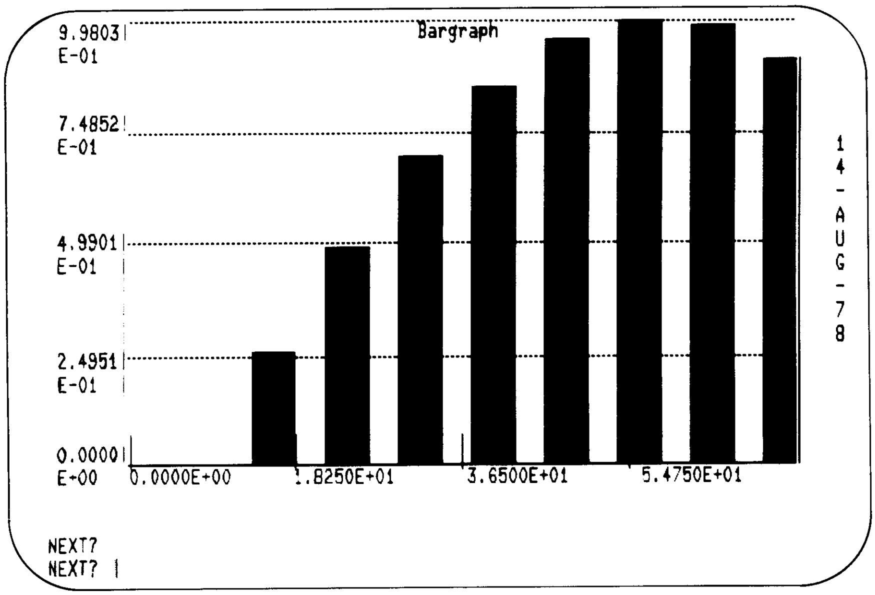


Figure 8. Sample Bargraph

You can display text in a variety of formats using graph routines. Examples of the different formats are shown in Figure 9.

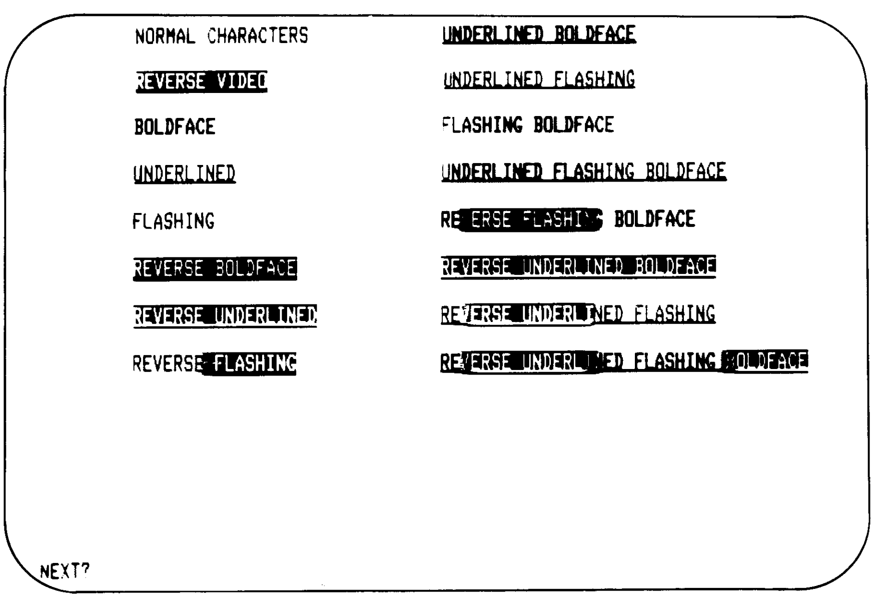


Figure 9. Text Formats

The graphic routines are described in *Book 4: MINC Graphic Programming*. Demonstration programs that use the graphic routines are given in Chapter 6 of this book.

MINC CART

The MINC cart is the structure which supports the diskette drives and the chassis (Figure 10). The MINC cart makes the system portable. When you wish to move MINC from one site to another, you disconnect the terminal cable, place the terminal on the low shelf of the cart, and push the cart to the new site.

The MINC power switch is located on the front of the MINC cart and the circuit breaker is at the bottom of the rear of the cart. The MINC cart also contains the housing for an isolation transformer, which can be purchased as an option.

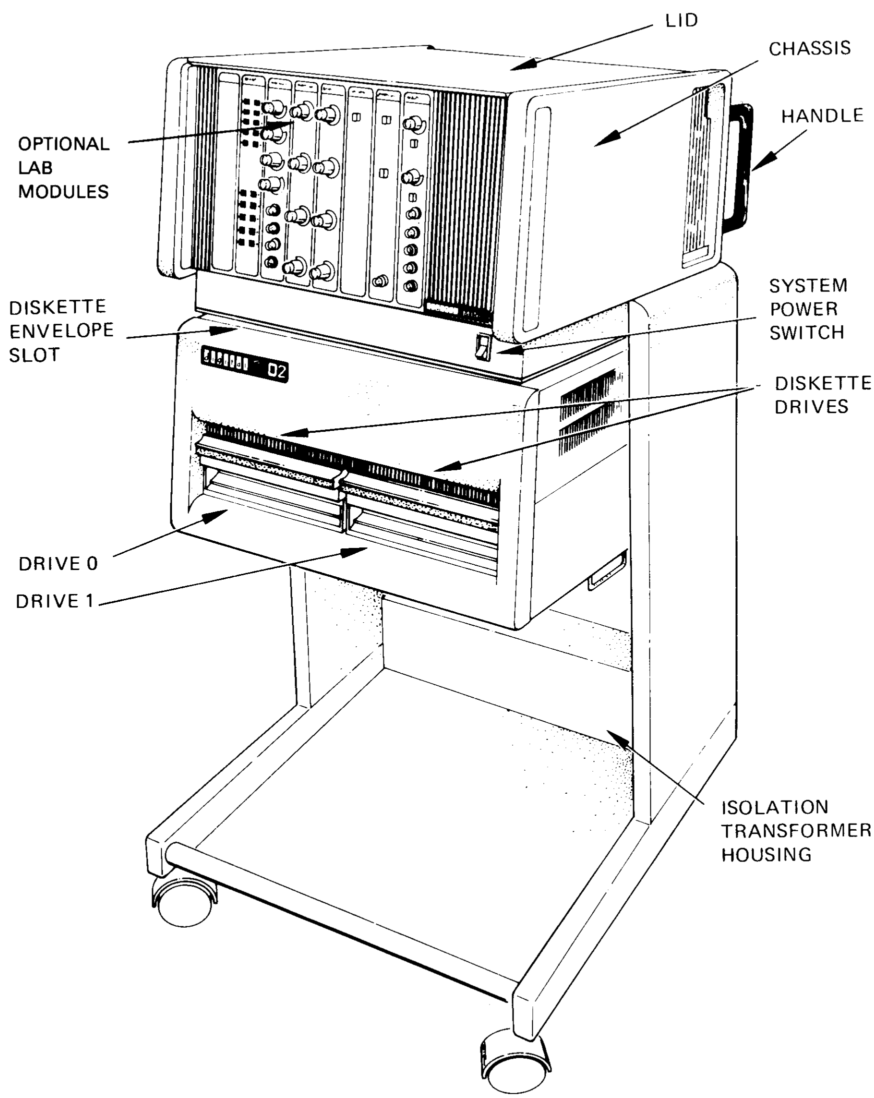


Figure 10. The MINC Cart

Diskette Drives

Minc has two diskette drives, as shown in Figure 10. You start MINC by inserting a system diskette or a demonstration diskette in the left hand drive and setting the system power switch to the ON position. Detailed instructions on starting MINC are given in Chapter 4.

Diskettes are flexible plastic platters with a magnetic coating. Instructions on the care and handling of diskettes are given in Chapter 3.

You save your programs and data on diskettes. Space is available on the system diskette for storage of programs and data. You can also insert another diskette in the right hand drive and save your information on that diskette. When you finish using MINC, remove the diskette that contains your programs and data. If you use MINC for several different applications, you can save the information for each application on a separate diskette. If several people share a MINC system, then each user can maintain a set of diskettes.

MINC Chassis

The MINC chassis contains the central processing unit and memory, diagnostic/loader module, power supply, the IEEE bus interface cable, the serial ASCII controller, and eight empty slots. These empty slots can be filled with any configuration of lab modules. The lab modules available with MINC are:

- Analog-to-digital converter
- Dual multiplexer
- Preamplifier
- Clock module
- Digital-to-analog converter
- Digital input module
- Digital output module

The chassis in Figure 10 contains all seven lab modules plus a blank in the eighth slot. The lab modules used in a configuration determine the capabilities of that configuration.

IEEE Bus

You can transfer data and control information between MINC and instruments manufactured to conform to the IEEE standard 488-1975 general purpose instrument bus. The IEEE bus connector is located on the back of the chassis.

You can connect IEEE-standard instruments to the IEEE bus and to each other. The IEEE protocol allows an instrument to operate as a talker (to send information), or a listener (to receive information). The MINC IEEE bus acts as the controller (to control bus-compatible instruments).

IEEE Bus Routines The MINC system includes a set of routines that control all activity on the IEEE bus. These routines determine which instrument is to talk and which is to listen; they poll the status of instruments; they recognize errors and respond with appropriate error messages; they initiate and terminate data transmission on the IEEE bus.

The IEEE bus routines are described in *Book 5: IEEE Bus Programming*. The SEND and RECEIVE routines are illustrated in the demonstration program for the IEEE bus given in Chapter 7.

Serial ASCII Controller

The serial ASCII controller is located at the back of the chassis. The purpose of the four lines of the controller is as follows:

CONSOLE	The MINC terminal plugs into this line.
SLU2	You can plug an LA35 line printer into Serial Line 2 (SLU2). This line operates at 300 Baud and can be used only for output. The LA35 line can be purchased separately, to provide printed output from MINC.
SLU1	You can connect a laboratory instrument that operates at 1200 Baud to Serial Line Unit 1 (SLU1).
SLU0	You can connect a laboratory instrument that operates at 9600 Baud to Serial Line Unit 0 (SLU0).

You can send data to and receive data from SLU1 and SLU0 with the CIN and COUT lab module routines. These routines are described in *Book 6: Lab Module Programming.* A demonstration program that uses these routines is given in Chapter 7 of this book.

LAB MODULES

The MINC chassis contains eight empty slots into which you can put any valid combination of MINC lab modules. The lab modules connect most analog and digital laboratory instruments to MINC. A MINC lab module is shown in Figure 11.

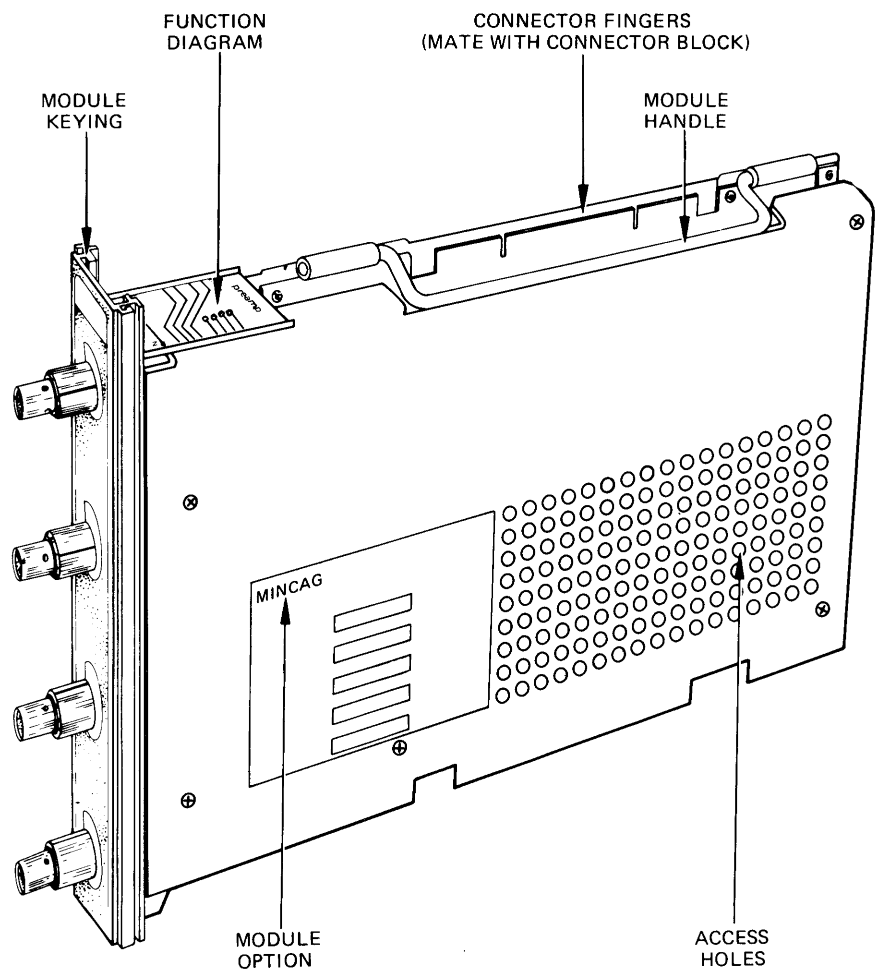


Figure 11. MINC Lab Module

Each MINC lab module has the following features:

It is completely protected within a metal enclosure.

It has a 24-line input/output connector.

It is keyed so that it can be plugged into the chassis only adjacent to electrically compatible modules, thus avoiding accidental damage to any of the modules.

It has a function diagram on its top panel.

As the modules are plugged into the chassis, these diagrams combine to give you an image of the functions performed by the system you have connected.

Modules can be conveniently inserted and removed from the MINC chassis, as shown in Figure 12. Detailed instruction on the installation of modules is given in the *Unpacking Guide.*

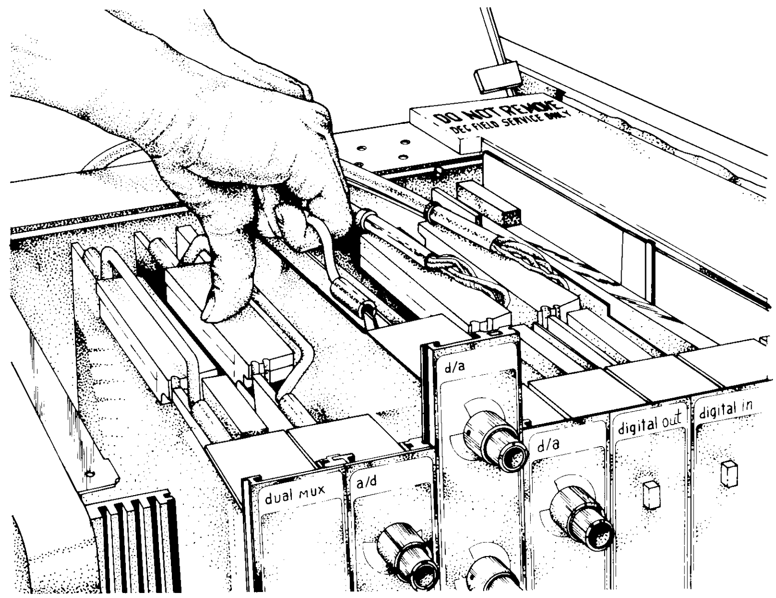


Figure 12. Removing a MINC Lab Module

The lab modules extend MINC's capabilities with the following four basic functions:

Analog-to-digital conversion

Digital-to-analog conversion

Digital input and output

Time keeping and event counting

Analog-to-Digital Conversion

Analog-to-digital (A/D) conversion is necessary when you want to monitor or control processes that involve continuously changing values. If these values are electrical voltage, current, or resistance equivalents, then they can be routed to an A/D converter and translated to the discrete digital form that is the internal representation used by MINC.

Conversion can be controlled in several ways. The A/D converter may be assisted by one or more of three other modules: the dual multiplexer, the preamplifier, and the clock module.

Analog-to-Digital Converter The MINC Analog-to-Digital

(A/D) converter translates the instantaneous value of a voltage into a digital value that can be used by MINC. The unit can support either 16 single-ended channels or eight single-ended and four differential channels.

A MINC A/D converter is illustrated in Figure 13. The connector block and function diagram are shown enlarged in this figure.

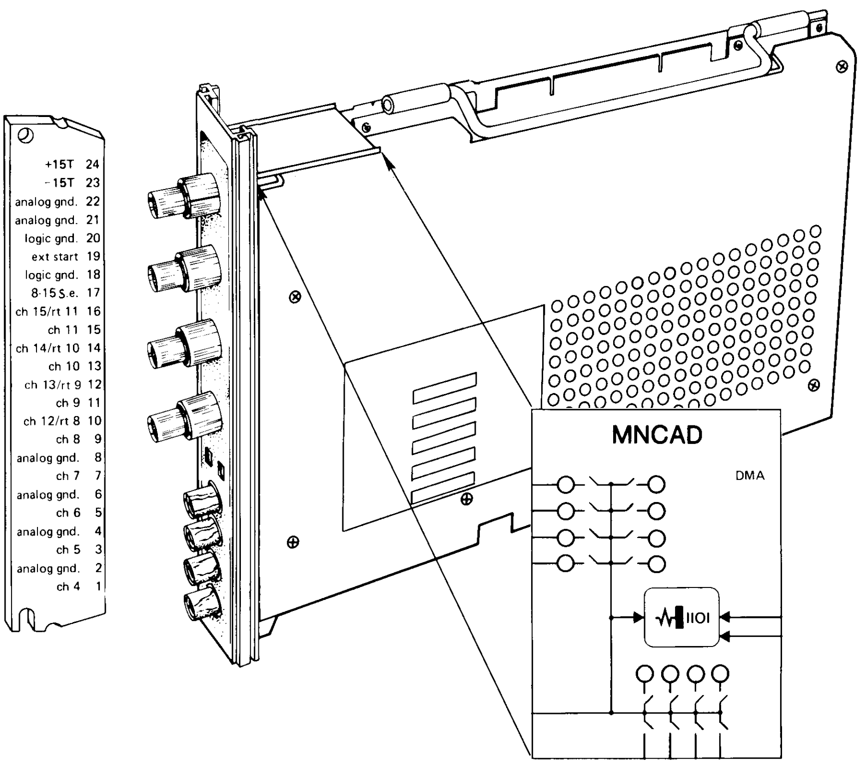


Figure 13. MINC A/D Converter

Dual Multiplexer A dual multiplexer expands the channel-handling ability of the A/D converter by 8, 12, or 16 channels, depending on the selected mode. The A/D converter contains the necessary control logic to deal with up to 64 channels. A MINC multiplexer is illustrated in Figure 14, with connector block and function diagram enlarged.

Preamplifier The MINC preamplifier is designed to accommodate external devices that produce signals whose levels are either too low or too high for direct A/D conversion. It also accommodates resistance or current measurements in addition to the standard voltage measurements. Each preamplifier provides instrument-quality gain control for a group of four input

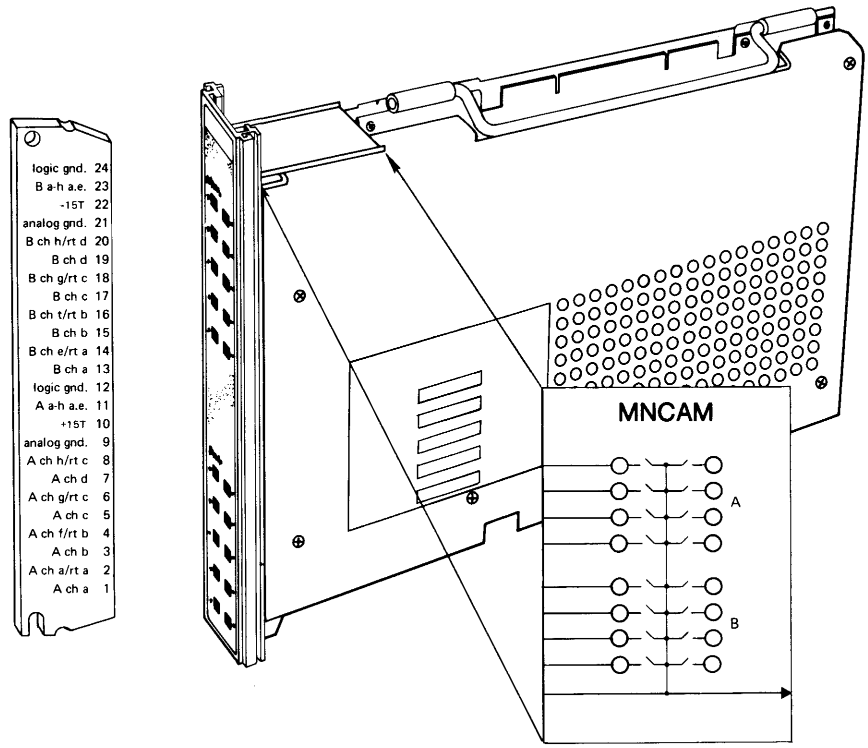


Figure 14. MINC Dual Multiplexer

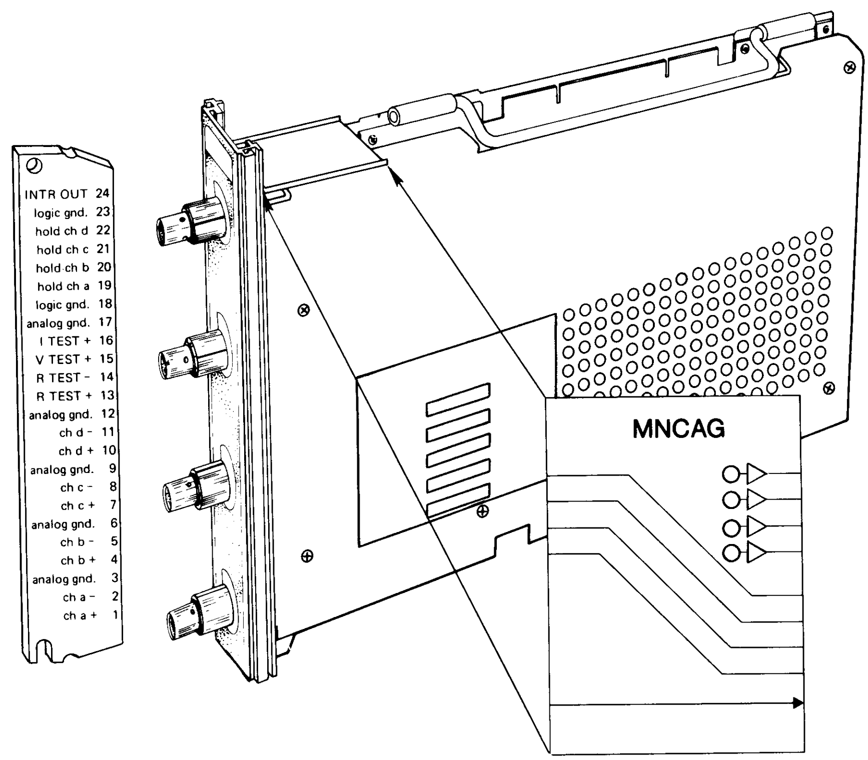


Figure 15. MINC Preamplifier

channels. A MINC preamplifier is illustrated in Figure 15, with connector block and function diagram enlarged.

Clock Module The MINC clock module is a multirate, multimode timing and counting device. You can use the clock module for counting external events or providing periodic signals to control the frequency of A/D conversions as well as other data transfers.

The clock contains two Schmitt triggers. You can set these triggers to fire when preset voltage levels are attained to control the frequency of A/D conversions or other MINC data transfers. A MINC clock module is illustrated in Figure 16, with connector block and function diagram enlarged.

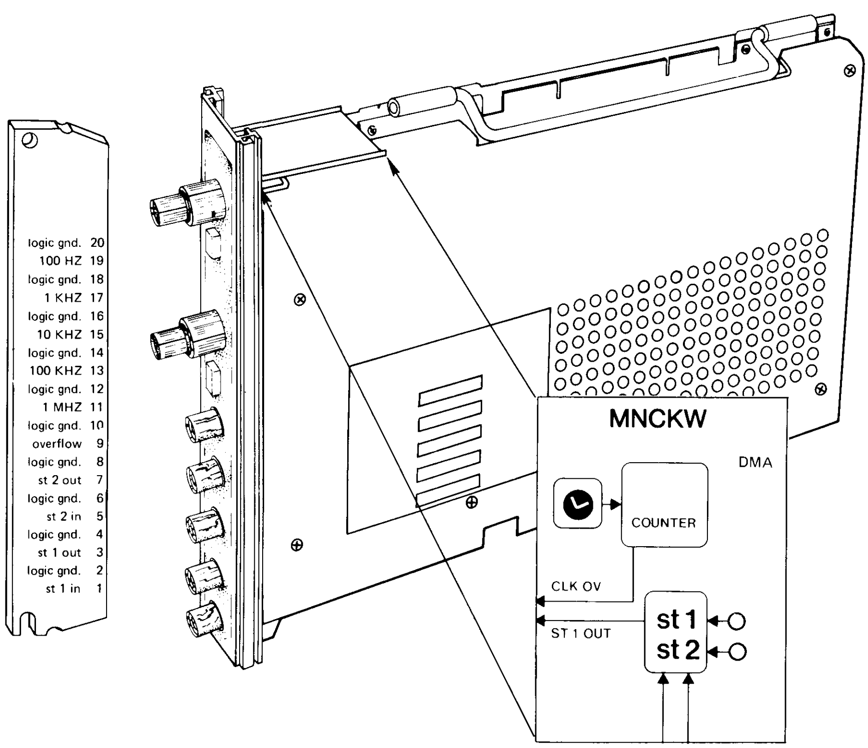


Figure 16. MINC Clock Module

Digital-to-Analog Conversion

Digital-to-analog (D/A) conversion is necessary when you want to connect MINC to external devices that expect analog voltage levels at their inputs.

Digital-to-Analog Converter The MINC digital-to-analog (D/A) converter accepts digital values from your programs and generates corresponding voltages. Each D/A converter contains four analog output channels. Since MINC accommodates up to four D/A converters, you can make use of as many as 16 analog output channels. The voltages you generate on these channels can be used to drive the X and Y axes of oscilloscopes and X-Y plotters or to provide precisely controllable analog values for input to voltage-controlled oscillators. A MINC D/A converter is illustrated in Figure 17, with connector block and function diagram enlarged.

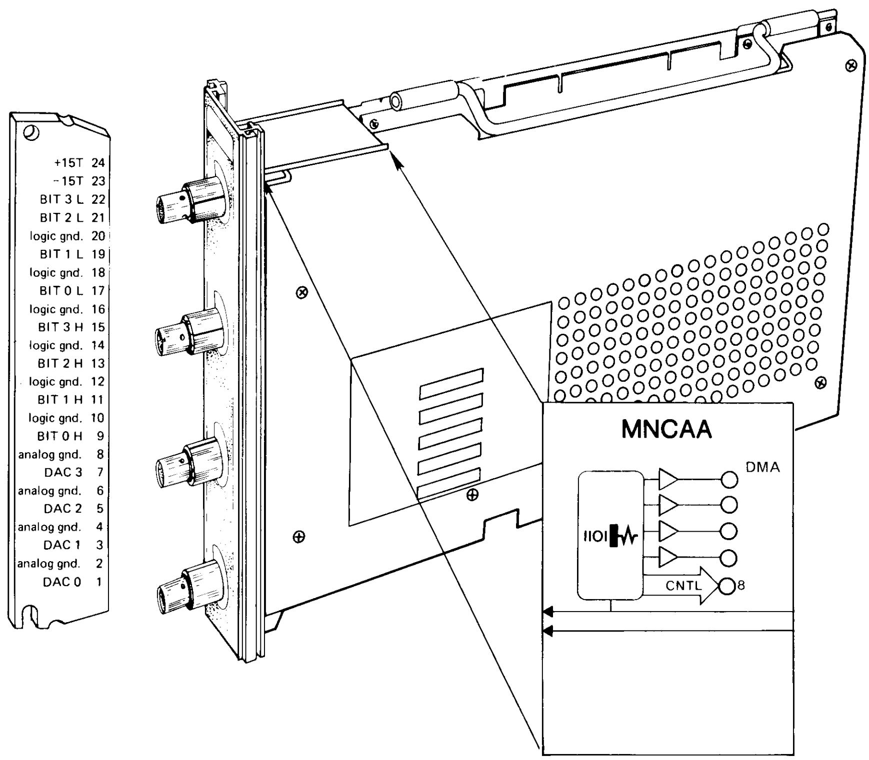


Figure 17. MINC D/A Converter

Digital Input and Output

Digital input to MINC is accomplished by the digital input lab module and digital output from MINC by the digital output lab module.

Digital Input Module The MINC digital input (DI) lab module provides convenient connection to external devices that output digital logic signals. Each DI module accepts up to 16 digital signal lines, either from discrete binary devices or from instruments that transmit 8-bit or 16-bit data words. DI modules are equipped with Schmitt triggers that can translate nonstandard logic waveforms and facilitate reliable transfer. A MINC DI module is illustrated in Figure 18, with connector block and function diagram enlarged.

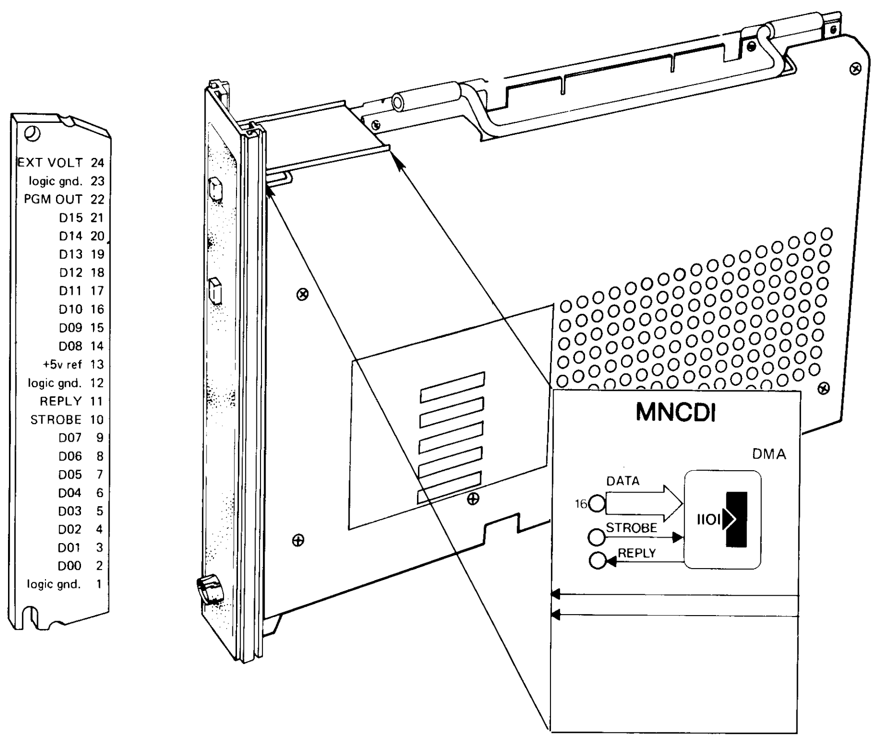


Figure 18. MINC DI Module

Digital Output Module The MINC digital output (DO) lab module sends digital logic levels from the system to as many as 16 binary instruments such as relays, lamps, or light-emitting diodes. You can use the DO module to transmit information to external instruments that require parallel data in 8-bit or 16-bit groups. A MINC DO module is illustrated in Figure 19, with connector block and function diagram enlarged.

Time Keeping and Event Counting

The MINC clock module (Figure 16) can be controlled by your programs in a variety of ways. You can set the clock module to initiate events (such as analog-to-digital or digital-to-analog conversions) after prescribed intervals have elapsed. You can set it to accept external signals via its Schmitt trigger inputs and initiate an event after a predetermined number of these signals has occurred. You can use the clock module in conjunction with the digital input unit to keep an ongoing record of elapsed time that can be accessed by the computer whenever a specified digital input event occurs (time stamping).

Using MINC routines, you can operate the clock at any one of six different rates (from 50 Hz to 1 MHz). Its two external inputs are provided with Schmitt triggers that you can set to respond

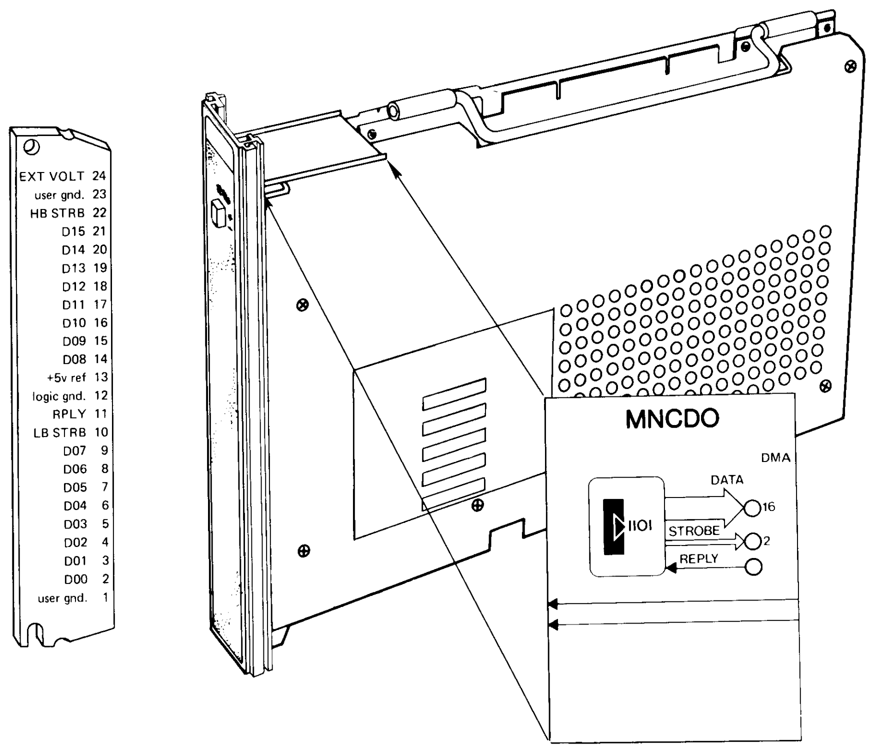


Figure 19. MINC DO Module

either to logic waveforms or to continuously changing analog voltages. For the latter case, controls are provided to permit you to select a given firing point on the positive or negative slope of the waveform. Since the Schmitt triggers are voltage sensitive, they respond reliably even to very slowly changing signals.

Lab Module Routines The MINC system includes a set of routines that enable you to control the logical communication between MINC and the external instruments connected to the lab modules.

The routines fall into three categories: data transfer, control, and support. The data transfer routines acquire data from instruments (analog, digital, or serial) and transmit output information. The control routines schedule transfers and respond to external events, including automatic program response to a time of day or Schmitt trigger input. The support routines provide useful tools for manipulating values, including BCD conversions, and creating masks for digital transfers.

The lab module routines are described in *Book 6: Lab Module Programming.* The analog input routine (AIN) is illustrated in the demonstration program for the A/D converter given in Chapter 8 of this book. Similarly the analog output routine (AOUT) is illustrated in the demonstration program for the D/A converter; the digital input routine (DIN) in the demonstration program for the digital input module; and the digital output routine (DOUT) in the demonstration program for the digital output module.

DOCUMENTATION

Instructions and guidelines on using the MINC system are contained in a set of eight books. These books contain tutorial, explanatory, and reference information about all aspects of MINC.

Book 1: Introduction to MINC introduces you to the MINC system and its use. You are currently half way through Book 1. At this point, you should be ready to begin using MINC. Part 2 of this book helps you use a set of demonstration programs designed to acquaint you with MINC.

Book 2: MINC Programming Fundamentals is a tutorial manual on the MINC system, which teaches:

How to use MINC as a calculator.

How to write programs.

How to perform operating procedures, such as storing data, copying disks, and so on.

How to edit programs and data.

Book 2 contains many examples, which illustrate the features of MINC and good programming practices. If you read and understand this manual thoroughly, you will have a good introduction to BASIC programming.

Book 3: MINC Programming Reference is the reference manual for the commands, statements, and concepts described in Book 2. The material is presented in alphabetical order for ease of reference.

Book 4: MINC Graphic Programming explains how to use the MINC system for graphic displays. The book contains an explanatory and a reference section.

Part 1, the explanatory section, is an introduction to the concepts of graphic programming. It describes how to implement these concepts using the graphic routines.

Part 2, an alphabetical reference, describes each of the 33 graphic routines.

For each graphic routine, the book contains one or more illustrations. In addition, Book 4 contains many short programming examples and sample statements. By doing each example on the MINC system and comparing your result with the coordinate illustration, you can ensure that you understand MINC graphic programming.

Book 5: MINC IEEE Bus Programming explains how to use the IEEE bus capability of MINC. The book contains an explanatory and a reference section.

Part 1 is an introduction to the protocols involved in the IEEE bus communication and a guide to the use of the routines described in Part 2.

Part 2, an alphabetical reference, describes each of the 23 IEEE bus routines.

Book 6: MINC Lab Module Programming explains the routines that control communication with the lab modules. Like Books 4 and 5, Book 6 contains an explanatory and a reference section.

Part 1, the explanatory section, describes the kinds of transfer and control processing provided by the lab module routines. Further, it discusses the specific concepts you will need to plan programs that use the lab modules and explains many terms used in lab module programming.

Part 2, an alphabetical reference, describes each of the 33 lab module routines.

Book 7: Working with MINC Devices provides reference information regarding the physical and electrical properties of the MINC system. Book 7 helps users who must make electrical connections between MINC lab modules and instruments outside the MINC system. It also helps users who are experiencing trouble with their system and must determine whether their difficulties stem from improper programming, inappropriate connections, or hardware malfunction.

In general, you go to Book 7 when you have a question likely to involve physical rather than programming manipulations — how to make connections, how to set up the system terminal, how to run system diagnostic programs.

Book 8: MINC System Index consists of two parts, an index and an error message compendium.

The index contains an entry for each item indexed in any of the other manuals in the set. You can use the system index to locate information in other manuals.

The error compendium is an alphabetical list of all the error messages that MINC produces. You can look up any error message here and find help on the possible causes of the error and suggestions for recovering from it.

PART 2
USING MINC

CHAPTER 3
GENERAL INSTRUCTIONS

The chapters in this part describe the operating procedures you need to use MINC and present a sequence of programs that you can run to familiarize yourself with MINC and to demonstrate that all the parts are properly connected.

After reading the general instructions in this chapter and following the procedure for starting MINC in Chapter 4, you can read the subsequent chapters in any order. Each chapter presents a set of one or more demonstration programs for a MINC component, as follows:

Chapter	Component
5	MINC BASIC
6	Graphics
7	IEEE Bus and Serial ASCII Controller
8	Lab Modules

DEMONSTRATION PROGRAMS

Demonstration programs are designed to introduce you to MINC. You can use these programs without any previous knowledge of MINC or programming.

Using a Demonstration Program

For each demonstration program, you are given a suggested set of responses. A copy of the terminal screen is reproduced for these responses so that you can verify that you have performed the procedure correctly and that MINC has performed the appropriate computations.

You are then given some possible variations in the use of the demonstrations program. You can experiment with these variations and observe the resulting effects. Or you can choose your own variations. When a demonstration program asks you for a response, it includes, as part of the question, the possible set of responses. These possible responses are given at the end of the question in parentheses. For example, suppose the program asks the following question:

```
How many permutations do you wish to see (1 to 20)?
```

You can respond with any number between 1 and 20. If you respond with a number outside that range, the program repeats the question and waits for another response.

Looking at a Demonstration Program

You are shown a segment of each demonstration program. The segment chosen illustrates the part of the program most relevant to the feature of MINC being demonstrated. This portion is given only to introduce you to MINC programming. You should not expect to understand the program at this point. The other books provided with MINC carefully explain the MINC statements used in these demonstrations.

Listing a Demonstration Program

You can, if you wish, look at the entire program by following the listing instructions given with each demonstration. Demonstration programs contain a good deal of programming to control printing explanatory text, to detect cases in which the user enters an invalid response, to recover from invalid inputs, and so on. This additional programming makes the demonstration program easy to use.

RECOVERY PROCEDURES

If you have trouble using MINC, try the following recovery procedures:

1. Consult Appendix A for help on a particular abnormal behavior.

2. If you cannot find an appropriate correction procedure in Appendix A, back up to the last step that worked correctly and try the sequence again.

3. If the problem persists, consult the "Troubleshooting" section of *Book 7: Working with MINC Devices.*

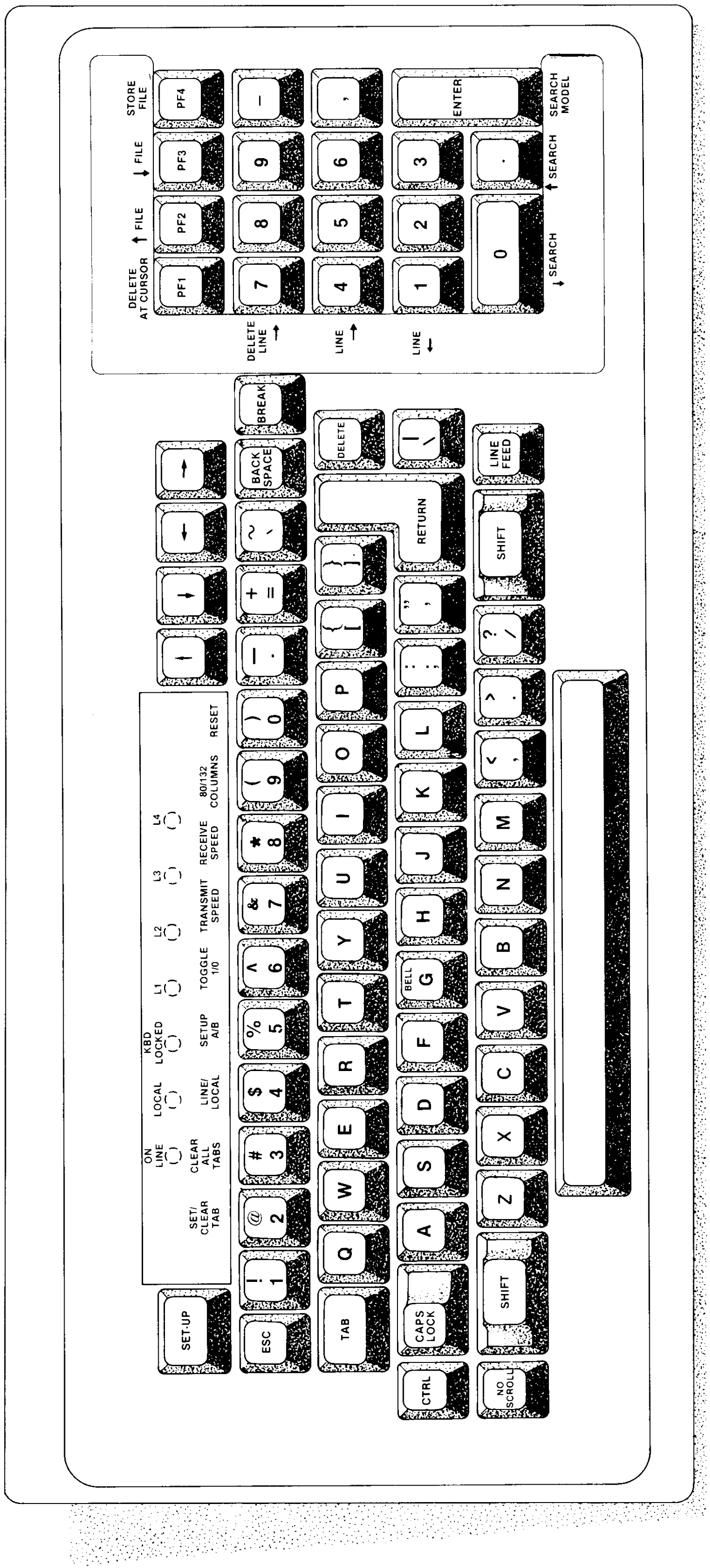


Figure 20. The MINC Terminal Keyboard

TERMINAL KEYBOARD

You can familiarize yourself with the terminal keyboard by examining the keyboard illustrated in Figure 20. Certain keys are of special significance. A brief introduction to the special keys you will need in the following demonstration programs is given here. A more detailed discussion of special keys is found in *Book 2: MINC Programming Fundamentals.*

The DELETE key is used to correct typing mistakes. If you make a typing mistake, press the DELETE key to erase the most recently typed character. If the typing error is not the last character typed, press the DELETE key as many times as necessary to back up to the error and erase it. Then resume typing.

The RETURN key is used to end a message to MINC. When you type a command, you end it by pressing the RETURN key.

The CTRL key is used, in combination with another key, to identify a special function for that key. The abbreviation CTRL stands for "control." To specify a key's control function, hold down the CTRL key and press the designated key. For the procedures in this manual, you need to specify the control function for the C key. The CTRL/C function is used to interrupt MINC. When you want MINC to stop its current activity, you hold down the CTRL key and press the C key twice.

The NO SCROLL key is used to temporarily stop MINC from displaying lines on the terminal screen. You use this key when MINC is displaying many lines of information. These lines are displayed faster than you can read them so you press the NO SCROLL key to stop the display. When you are ready to continue you press the NO SCROLL key again and MINC resumes its display. This procedure can be repeated as many times as you wish.

HANDLING DISKETTES

You can learn the correct way to handle a diskette by examining Figure 21.

Handle the diskette by its edges. Never touch the exposed surfaces of the diskette.

Protecting Your Diskette

To protect your diskettes from accidental destruction, use the following precautions:

- Do not write on the envelope containing the diskette. Write any information on a label before you affix it to the diskette.
- Do not use paper clips on the diskette.

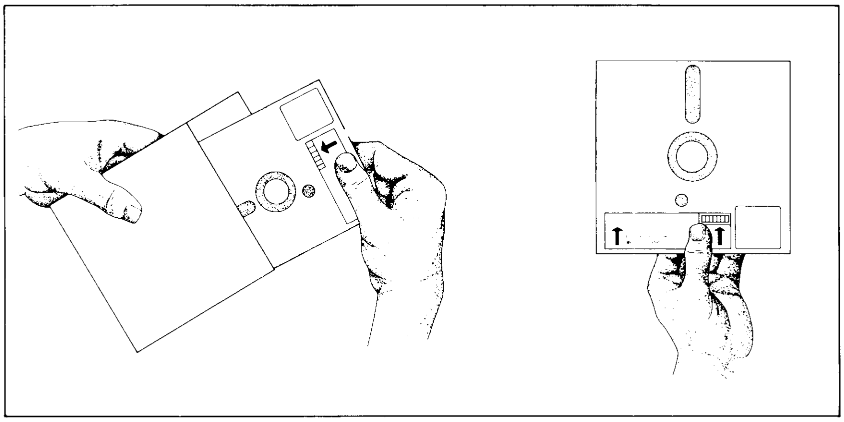


Figure 21. Handling a Diskette

- Do not use writing instruments that leave flakes, such as lead or grease pencils, to label the envelope.
- Do not touch the surface exposed in the diskette slot or index hole.
- Do not bend or clean the diskette in any manner.
- Keep the diskette away from magnets or tools that may have become magnetized. A diskette exposed to a magnetic field may lose information.
- Do not expose the diskette to a heat source or prolonged sunlight.
- Return the diskette to the envelope when not in use to protect it from dust and dirt. Diskettes not being used should be stored in a file box if possible.
- When the diskette is in use, protect the empty envelope from liquids, dust, and metallic materials.
- Do not place heavy items on the diskette.
- Do not store diskettes on top of computer cabinets or in places where dirt can be blown into the diskette interior by fans. Store diskettes in their envelopes in horizontal stacks of ten diskettes or less. If vertical storage is necessary, the diskettes should be supported so that they do not lean or sag, but should not be subjected to compressive forces. Permanent deformation may result from improper storage.

- If a diskette has been exposed to extreme heat or extreme cold, let it come room temperature before you use it. Remove the diskette from its envelope to allow it to adjust to the room temperature more rapidly.

Inserting a Diskette in a Drive

To insert a diskette into a diskette drive, follow these instructions:

1. Remove the diskette from its protective envelope.

2. Release the spring-loaded door on the diskette drive by squeezing the release bar beneath the door handle.

3. Check that the drive is empty. If not, remove the diskette in the drive and return it to its protective envelope.

4. Gently insert the diskette, label side up, oriented as shown in Figure 22. Do not force it into the drive.

5. Press the diskette drive door down until it clicks. The door latches when it is fully closed.

6. Place the protective envelope in the slot provided for envelopes directly above the diskette drive so that it will be convenient when you remove the diskette from the drive.

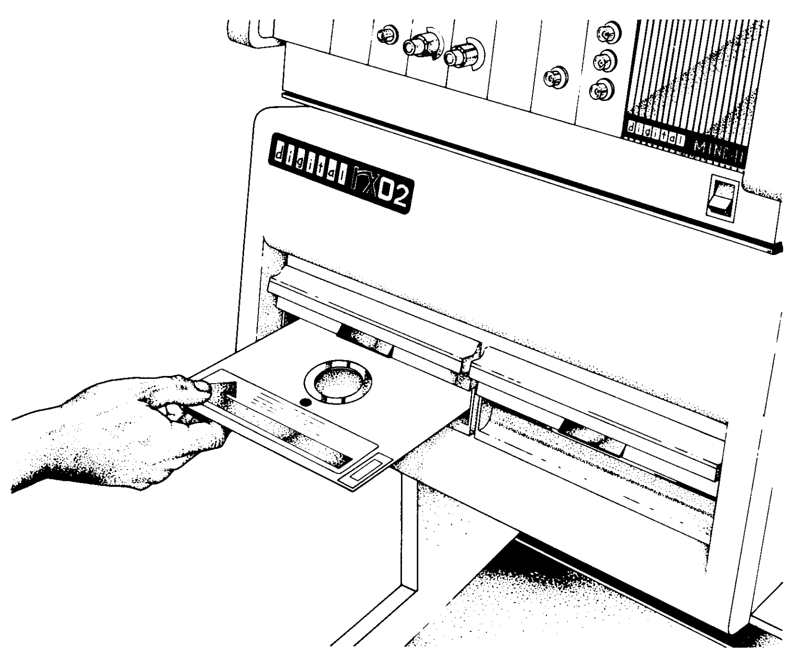


Figure 22. Inserting a Diskette

CHAPTER 4
STARTING MINC

This chapter describes starting MINC, and illustrates some simple calculator operations.

STARTING THE SYSTEM

Before starting MINC, make sure the switches are set properly, as shown in Figure 23.

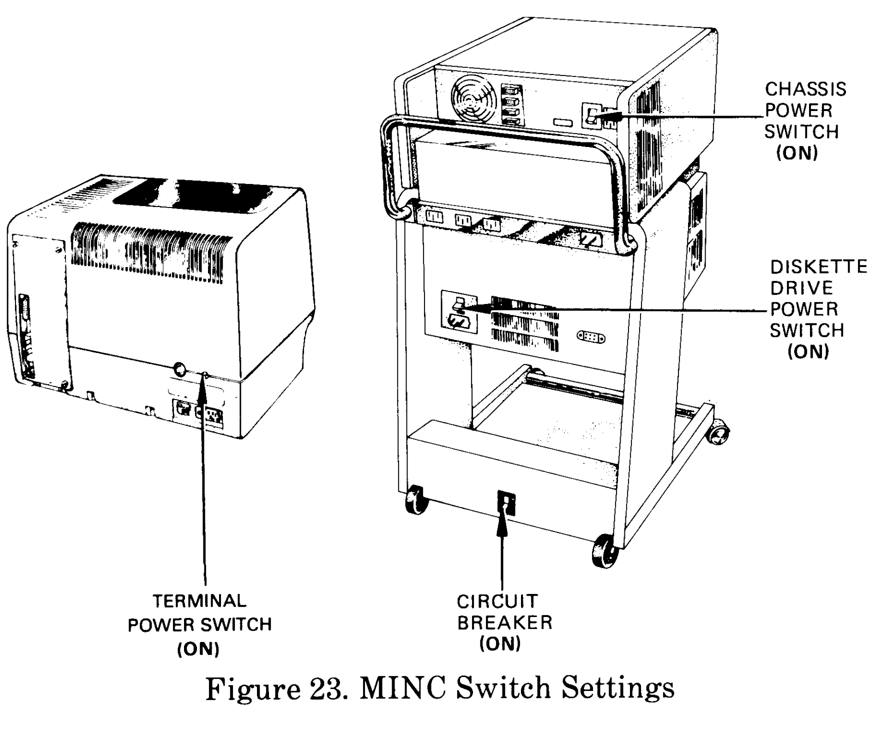


Figure 23. MINC Switch Settings

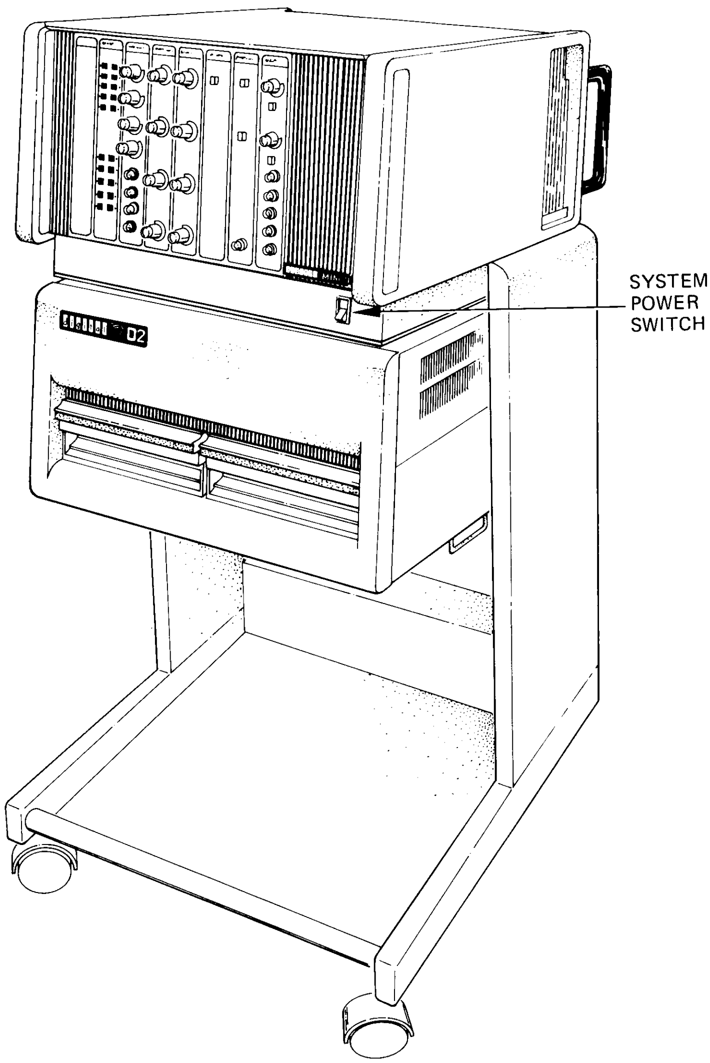


Figure 24. The System Power Switch

3. Insert a blank diskette in the right hand drive (Drive 1).

4. Set the system power switch to the ON (lighted) position.

When you set the system power switch to the ON position, MINC makes some clicking noises. These noises are normal operating sounds. MINC then displays instructions on the terminal screen. These instructions guide you through the process of duplicating a Master Demonstration diskette. MINC works with you to ensure that the copying procedure is carried out properly by informing you of the actions it is performing, asking you if you

For example, to enter the tenth of May in the year 1978, type:

```
10-MAY-78 (RET)
```

If you notice a mistake in typing before pressing the RETURN key, press the DELETE key to delete the last character you typed. If you need to delete more than one character, type the DELETE key as many times as necessary to back up to the incorrect character and delete it. Then type in the correct characters. Type the RETURN key to terminate the line and notify MINC that your response is complete.

Next, MINC asks you to enter the correct time. Type the time in 24-hour format as follows:

```
hh:mm:ss (RET)
```

where:
- hh is the hour
- mm is the minute
- ss is the second
- (RET) indicates the RETURN key.

You can omit the seconds if you wish. For example, to enter 3:35 p.m., type:

```
15:35 (RET)
```

After about 15 seconds, MINC displays the message "READY" on the screen, indicating that it is ready to perform calculations, run programs, or collect data.

DOING SOME SIMPLE CALCULATIONS

Sample Calculation

You can now try some simple calculations. To do this, you type the word "PRINT" followed by a calculation.

For example, to calculate the sum of 2 and 2, type the following:

```
PRINT 2+2 (RET)
```

After you press the RETURN key, MINC performs the calculation and displays the result on your screen as follows:

```
4
```

To calculate the average of the three values 22.12, 35.41, and 76.54, type the following:

```
PRINT (22.12+35.41+76.54)/3 (RET)
```

```
2.44949
2.64575
2.82843
3
3.16228
```

GETTING HELP

You can get help from MINC while you are using it by typing HELP. To find out about the kind of help MINC provides, type:

```
HELP (RET)
```

Or, to get a list of the BASIC commands, type:

```
HELP BASIC (RET)
```

You can also get help from *Book 8: MINC System Index*. Book 8 contains a complete index to all the books describing MINC and an alphabetical guide to the error messages produced by MINC. If you get an unfamiliar error message while using MINC, you can look it up in Book 8.

Next, locate a MINC demonstration diskette. If you have a demonstration diskette, you can proceed directly to the section on "Using the Demonstration Diskette" on page 38. If you have just unpacked MINC or if you don't have a demonstration diskette, you must make a demonstration diskette by copying the Master Demonstration diskette.

MAKING A DEMONSTRATION DISKETTE

To make a demonstration diskette, you need the Master Demonstration diskette and a blank diskette. If you have just unpacked MINC, you will find these diskettes in the software box.

Locating the Necessary Diskettes

The MINC software box contains four different types of diskettes. The following list identifies each type of diskette and indicates how many of each type of diskette are included in the software box.

- One Master Demonstration diskette: A diskette that cannot be used for any purpose except to make new copies of the demonstration diskette.
- One Master System diskette: A diskette that cannot be used for any purpose except to make new copies of the system diskette.
- Three blank diskettes: Two of these diskettes can be used to copy master diskettes or save your programs and data. One of these blank diskettes must be reserved for Field Service use.
- One diagnostic diskette: A diskette that is used to check out the system hardware.

A demonstration diskette is different from a system diskette in that it contains, in addition to the system, a number of demonstration programs. You will need the Master Demonstration diskette and one blank diskette for the following procedures.

Copying the Master Demonstration Diskette

To copy the Master Demonstration diskette, follow these steps:

1. Turn the system power switch (see Figure 24) to the OFF (unlighted) position.
2. Insert the Master Demonstration diskette in the left hand drive (Drive 0) using the instructions given in Chapter 3.

wish to proceed, and waiting until you indicate you are ready to continue.

When the process of copying the Master Demonstration diskette is complete, you have your own demonstration diskette. Remove your demonstration diskette from Drive 1, write some identifying information on a label and apply the label to your diskette. Remove the Master Demonstration diskette from drive 0 and store it for future use. As long as you have the Master Demonstration diskette, you can make new demonstration diskettes when existing ones wear out or are accidentally destroyed.

USING THE DEMONSTRATION DISKETTE

To use the demonstration diskette, follow these steps:

1. Set the system power switch (see Figure 24) to the OFF (unlighted) position.
2. Insert the demonstration diskette in the left-hand drive (Drive 0), using the instructions given in Chapter 3.
3. Set the system power switch to the ON (lighted) position.

When you set the system power switch to the ON position, MINC makes some clicking noises. These noises are normal operating sounds. After 15 seconds, MINC displays an identifying message on the terminal screen and asks you to enter the date.

Entering the Date and Time

Type the date in the following format:

dd-mmm-yy (RET)

where:
dd is the day of the month
mmm is one of the following abbreviations for the month:

JAN	MAY	SEP
FEB	JUN	OCT
MAR	JUL	NOV
APR	AUG	DEC

yy is the last two digits of the year
(RET) indicates the RETURN key.

The RETURN key and its function are described on page 32.

If you make a mistake, you can use the DELETE key, as described previously, to delete a character at a time. When you are satisfied that the line is correct and ready to be interpreted by MINC, press the RETURN key. MINC performs the calculation and displays the following result on your screen.

```
44.69
```

Sample Error Message

MINC examines each command you type for validity. If the command is valid, MINC executes it and displays any result. If the command is not valid, MINC displays an error message that tries to describe in what way the command is invalid. A common typing error occurs when you fail to press the shift key to type a parenthesis. For example, suppose you want to print the square root of 4 and you mistakenly type:

```
PRINT SQR(40 (RET)
```

MINC does not understand the nature of that particular mistake, but it knows that the command is invalid and displays the following message:

```
?MINC-F-Syntax error; cannot translate the statement
```

By examining the command you typed, you can determine the mistake and reenter the command properly, as follows:

```
PRINT SQR(4) (RET)
```

MINC validates the command and responds with the answer.

You need not be concerned about typing something that might damage MINC. MINC works with you to prevent and correct mistakes.

More Calculations

You can calculate more than one result at a time. Suppose you want to know the square roots of the integers from 1 to 10. Type the following command:

```
FOR I = 1 TO 10 \ PRINT SQR(I) \ NEXT I (RET)
```

MINC responds with the ten square roots as follows:

```
1
1.41421
1.73205
2
2.23607
```

CHAPTER 5
BASIC PROGRAMS

This chapter considers two typical BASIC programs. The first program calculates the sine and cosine for a specified angle and the second program randomly permutes a set of test items.

PROGRAM TO CALCULATE SINES AND COSINES

The first demonstration program, TRIG, finds the sine and cosine of an angle. This program is very simple, but it illustrates some fundamental techniques.

Running the Program

To run the program, type the following command:

```
RUN TRIG
```

When you press the RETURN key to end the line, MINC receives the command and runs the demonstration program. The program identifies itself and asks you to enter an angle in degrees.

Type "60" for the angle. The resulting display is shown in Figure 25.

If you are having trouble or if the program does not behave as described, try the procedures on page 30.

You can repeat this demonstration to try some more angles or you can conclude the demonstration. To repeat the demonstration, type "Y" in response to the question "AGAIN (Y or N)?" To conclude this demonstration, type "N" in response to that question.

```
    TRIG

This program calculates the sine and cosine of an angle

Please enter the angle in degrees (0 to 360)? 60

ANGLE         SINE          COSINE

 60            .866025       .5

AGAIN (Y or N)? █
```

Figure 25. TRIG Screen Sample

Looking at the Program

Here is the portion of the TRIG program that converts the angle you specify from degrees to radians and prints the results.

```
100 Z=X*PI/180
110 PRINT 'ANGLE', 'SINE', 'COSINE' \ PRINT
120 PRINT X, SIN(Z), COS(Z)
```

The variable X contains the value you entered in response to the program's request, "Please enter angle in degrees (0 to 360)". Line 100 performs the calculation necessary to convert the angle to radians. Line 110 prints a heading and a blank line; line 120 prints the angle and its sine and cosine.

If you want to examine the complete program, type the following command:

```
LIST (RET)
```

Or if you have purchased an LA35 line printer option, you can get a listing of the program by typing the following:

```
COPY TRIG LP: (RET)
```

You can learn more about MINC BASIC by consulting Books 2 and 3. *Book 2: MINC Programming Fundamentals* teaches commands you need to know to run MINC and explains how to put these commands together to perform a task. *Book 3: MINC Programming Reference* gives a detailed technical summary for each command.

RANDOM PERMUTATIONS PROGRAM

The problem of randomly permuting a set of test stimuli occurs frequently in the laboratory environment. The program SHUFFL uses the algorithm recommended by Knuth in *Seminumerical Algorithms* (Vol. 2)* for producing random permutations. First, the program lets you determine the number of permutations to be performed on a set of preset items. Then the program lets you enter your own set of test items and experiment with permutations of these test items.

Running the Program

To run the program, you type the following:

RUN SHUFFL (RET)

When you press the RETURN key to end the line, MINC receives the command, gets the demonstration program SHUFFL from the Demonstration diskette, and runs it. The program identifies itself and asks you to enter the number of permutations to be performed. (The upper limit, 20, was arbitrarily selected to ensure that all the permutations are displayed on the screen at the same time.)

Type "5" in response to MINC's question. The terminal screen displays the five permutations shown in Figure 26.

```
          SHUFFL

Program to randomly permute a set of items
The item set consists of six words describing MINC.

How many permutations do you want to see (1 to 20)? 5

The items in the set are:

capable  easy-to-learn  expandable  friendly  versatile  portable

The permutations are:

easy-to-learn  versatile  portable  friendly  expandable  capable
portable  easy-to-learn  friendly  capable  versatile  expandable
capable  easy-to-learn  portable  versatile  friendly  expandable
capable  friendly  portable  expandable  versatile  easy-to-learn
versatile  easy-to-learn  expandable  portable  friendly  capable

Do you want to try another permutation (Y or N)? █
```

Figure 26. SHUFFL Screen Sample 1

You can choose to see another set of permutations by typing "Y" followed by the RETURN key in response to the question "AGAIN?" Or you can choose to continue to the next step in the demonstration by typing "N" followed by the RETURN key.

Type "Y" to permute the items again and type "20" for the number of permutations. Then type "N" to proceed to the next step, which lets you enter your own set of test items. Enter the first four letters of the Greek alphabet for test items and request three permutations. The resulting display is shown in Figure 27.

```
Do you wish to enter test data (Y or N)? Y

You can enter any character string for each item
Terminate the character string by typing the RETURN key

How many items are to be in the set (1 to 20)? 4
Enter item number  1 ? ALPHA
Enter item number  2 ? BETA
Enter item number  3 ? GAMMA
Enter item number  4 ? DELTA

How many permutations do you want to see (1 to 20)? 3

The items in the set are:

ALPHA  BETA  GAMMA  DELTA

The permutations are:

DELTA  ALPHA  BETA  GAMMA
GAMMA  DELTA  BETA  ALPHA
ALPHA  DELTA  BETA  GAMMA

Do you want to try another permutation (Y or N)?
```

Figure 27. SHUFFL Screen Sample 2

You can try more permutations of the set of items you just entered by typing "Y" in response to the question "Do you wish to try another permutation (Y or N)?"

If you want to enter a different set of test items, type "N" in response to that question and type "Y" in response to the question "Do you wish to enter test items (Y or N)?" You conclude the demonstration by typing "N" in response to that question.

Looking at the Program

Here is the segment of the program that permutes the order of the items within a set by exchanging the items:

```
500 REM SWAP ITEMS TO RANDOMLY PERMUTE ORDER
510 FOR I2 = N TO 1 STEP -1
515 R1 = RND
520 R = INT(R1 * I2 + 1)
```

```
530 T$ = D$(I2)
540 D$(I2) = D$(R)
550 D$(R) = T$
560 NEXT I2
```

The program starts with the last item in the set and exchanges it with another item randomly selected from the set. The program then exchanges the next-to-last item with another randomly selected item and continues in this way until the first item has been exchanged with a randomly selected item.

The program uses the random function RND to generate a random number and then normalizes the random number so that it is within the desired limits.

You can look at the complete program by typing:

```
LIST (RET)
```

This program has more than 24 lines and, therefore, cannot be completely contained on the terminal screen. You can stop the screen display by typing the NO SCROLL key as described on page 32.

If you have the LA35 line printer option, you can obtain a listing by typing:

```
COPY SHUFFL LP:
```

References

*Knuth, D.E. *The Art of Computer Programming: Seminumerical Algorithms.* Reading, Mass.: Addison-Wesley, 1969, Vol. 2.

CHAPTER 6
GRAPHICS PROGRAMS

This chapter illustrates MINC's graphic capability. The demonstration diskette contains three programs involving graphics. The first demonstration program lets you select the number of cycles to be displayed for a sine wave. The second program lets you determine the number of terms for a Fourier series approximation to a graph. The third program lets you choose a portion of a graph for closer inspection.

CYCLES PROGRAM

The first demonstration displays a sine wave with 1 Hz frequency and asks you to select the number of cycles to be displayed.

Running the Program

To run the CYCLES program, type the following:

```
RUN CYCLES
```

When you press the RETURN key to end the line, MINC runs the CYCLES program. The program begins by identifying itself and placing one cycle of a sine wave on the screen. Then the program asks you to choose a number between 0 and 10 for the number of cycles to be plotted.

Type "2" to get a graph of two cycles of a sine wave. The resulting display is shown in Figure 28.

Type "Y" in response to the question "AGAIN (Y or N)?" to repeat the demonstration for another graph.

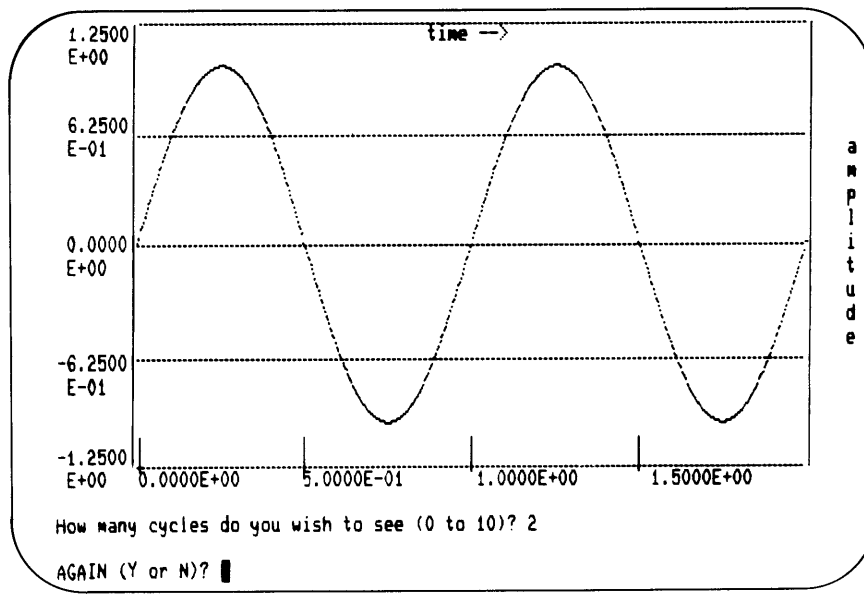


Figure 28. CYCLES Screen Sample 1

Type ".2" to get a graph of 2/10 of a cycle of a sine wave. The resulting display is shown in Figure 29.

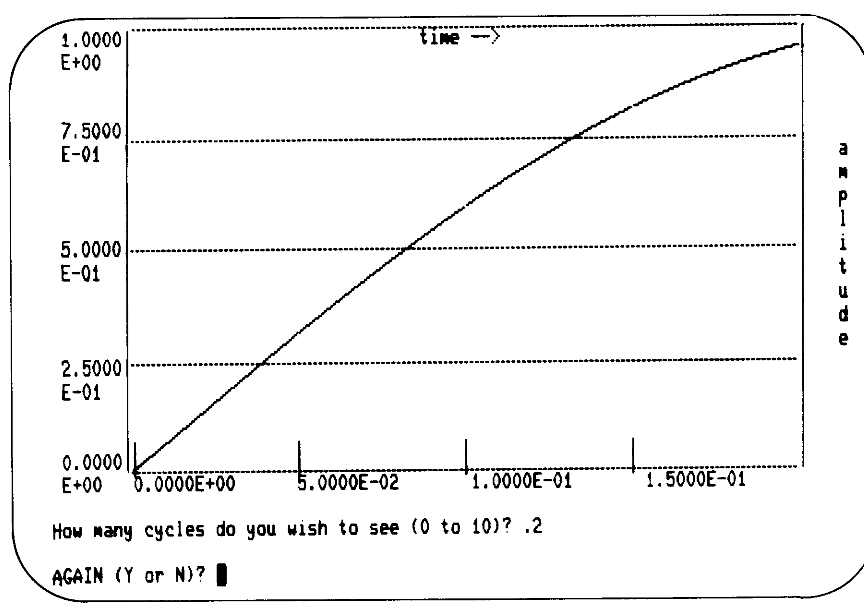


Figure 29. CYCLES Screen Sample 2

If the program does not behave as described, try the procedures given on page 30.

You can repeat the experiment specifying a different number of cycles and observe the resulting display.

When you want to conclude the demonstration, type "N" in response to the question "AGAIN (Y or N)?"

Looking at the Program

Here is the part of the CYCLES program that calculates the points to be displayed and then plots the graph:

```
385 REM Set frequency to 1HZ
390 F=1
395 REM Calculate X and Y points
400 Q = 2*PI*F*N
410 FOR I = 0 to 512
415 X(I) = N*I/512
420 Y(I) = SIN (Q*I/512)
430 NEXT I
440 REM Plot graph and label axes
450 GRAPH (,,X(0),Y(0))
455 LABEL('BOLD', 'time →', 'amplitude')
```

In order to display N cycles of a sine wave, the argument of the SIN function must range from 0 to $2\pi fn$. The loop in the above program segment stores Y values in this range as follows:

```
Y(0) = SIN (2*PI*F*N*0/512) = sin (0)
Y(1) = SIN (2*PI*F*N*1/512) = sin(2πfn/512)
...
Y(N) = SIN (2*PI*F*N*512/512) = sin (2πfn)
```

The GRAPH routine displays the points calculated and the LABEL routine adds labels to the X and Y axes.

You can examine the complete program by typing:

LIST (RET)

If you have the LA35 line printer option, you get a listing of the program by typing:

COPY CYCLES LP: (RET)

You can learn more about MINC's graphic capability by reading *Book 4: MINC Graphic Programming*.

FOURIER SERIES APPROXIMATION TO A SAWTOOTH

The second graphics demonstration illustrates a Fourier series approximation to a sawtooth function by displaying two graphs on the screen: the sawtooth graph and the approximation graph.

The sawtooth function is as follows:

$$f(x) = X \quad -\pi \leqslant X \leqslant \pi$$

The Fourier series approximation for f(x) is:

f(x) = 2 (sinx - 1/2sin2x + 1/3sin3x - ... #)

The program lets you specify the number of terms to be used in the approximation. You can start with one term and observe the improvement in the approximation as more terms are added. The program also lets you specify the number of points to be graphed. The time required to produce the graph depends on the number of points plotted.

Running the Program

To run the Fourier approximation program, type the following:

RUN APPROX (RET)

When you press the RETURN key, MINC runs the Fourier approximation program. The program begins by plotting a sawtooth. When the sawtooth graph is complete, the program asks you to enter the number of terms to be used in the Fourier approximation and the number of points to be plotted. The program then makes the sawtooth graph invisible and plots the approximation points. When the approximation points have all been plotted, the program makes the sawtooth graph visible again so that you can compare the approximation to the function it is approximating. The program then lets you choose whether or not to shade the approximation.

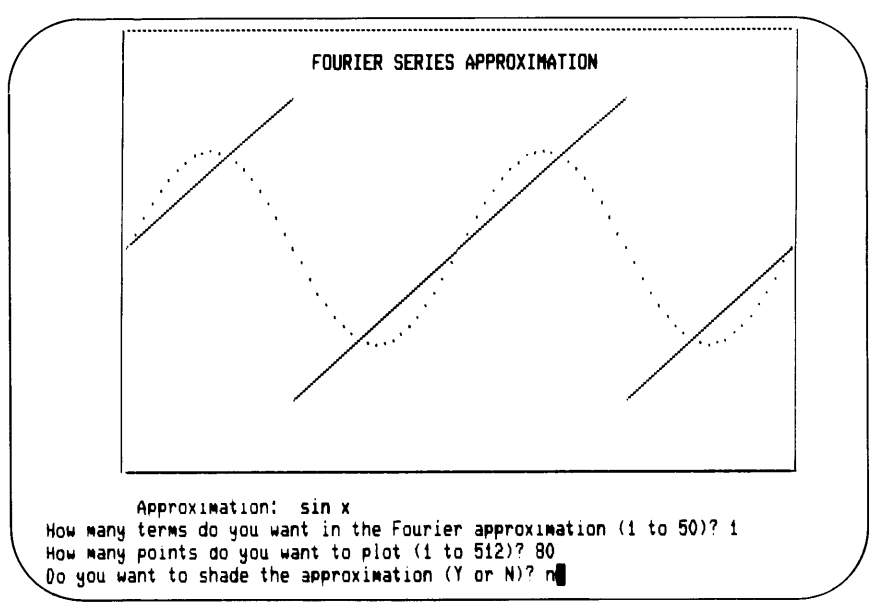


Figure 30. APPROX Screen Sample 1

Type "1" for the number of terms in the approximation and "80" for the number of points to be plotted. Type "N" in response to the question about shading. The resulting display is shown in Figure 30.

A single-term approximation is simply a sine wave and not a good approximation to the sawtooth.

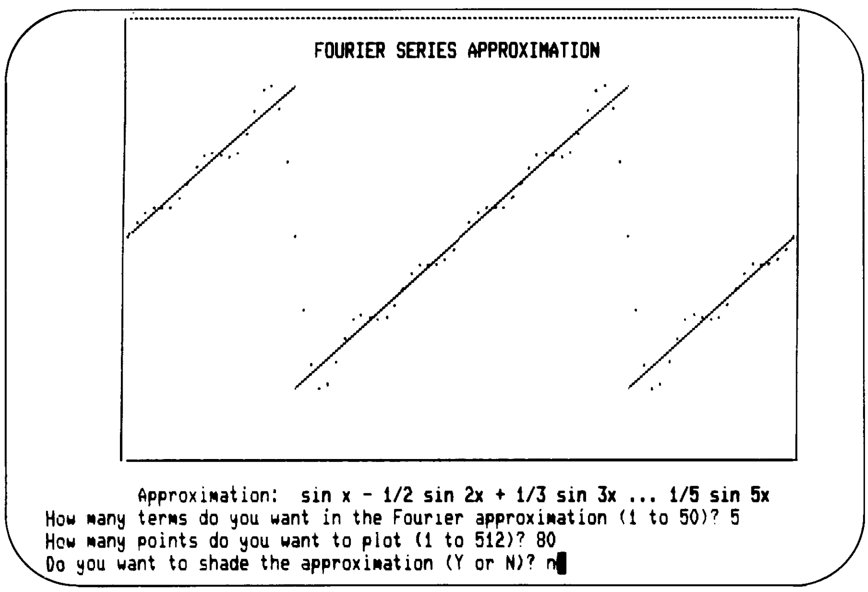


Figure 31. APPROX Screen Sample 2

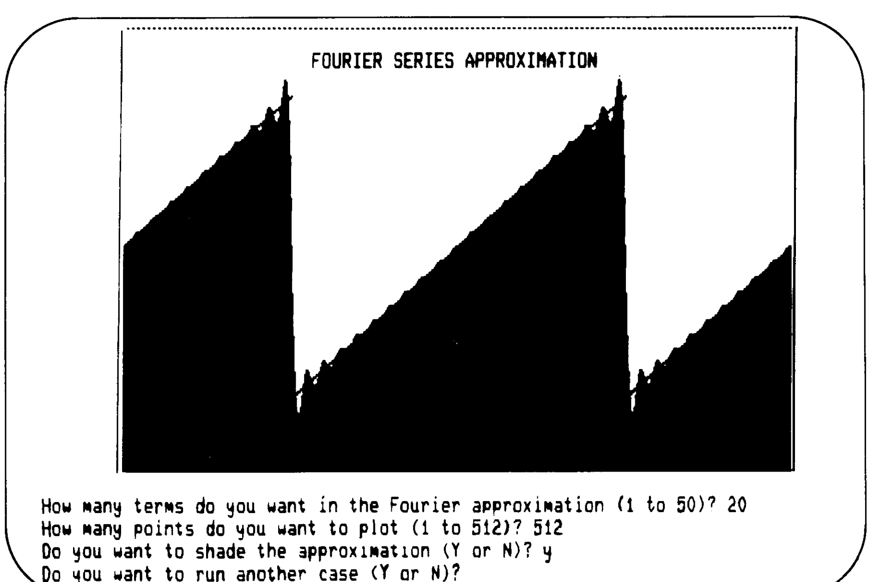


Figure 32. APPROX Screen Sample 3

Type "Y" in response to the question "AGAIN (Y or N)?" to try another approximation. Type "5" for the number of terms and "80" for the number of points. Type "N" in response to the question about shading. The resulting display is shown in Figure 31.

A five-term approximation more closely approximates the sawtooth.

Type "20" for the number of terms and "512" for the number of points and type "Y" in response to the question about shading. The resulting display is shown in Figure 32.

Shading is effective when a curve rises and falls rapidly; it depicts these fluctuations more clearly than points alone can.

Looking at the Program

Here is the portion of the APPROX program that computes the approximation and displays it point by point:

```
370 FOR J=0 TO N
380 X=4*PI*(J/N)-2*PI
390 S1=+2
400 Y=0
410 FOR K=1 TO I
420 Y=Y+S1*SIN(K*X)/K
430 S1=-S1
440 NEXT K
450 POINT (,X,Y,1)
460 NEXT J
```

The value of the variable I determines the number of terms in the approximation. The value of the variable N determines the number of points to be plotted.

If you specify an approximation with only one term, the inner loop controlled by the value I (lines 410 to 440) is executed once. If you ask for two terms, that loop is executed twice. That is, the approximation value, A, is:

A =(+2) * SIN(X) + (-2) * SIN(2X)/2

For three terms, the approximation value is:

A = (+2) * SIN(X) + (-2) * SIN(2X)/2 + (+2) * SIN(3X)/3

After each approximation point is calculated, it is plotted by the POINT routine.

WINDOW PROGRAM

The third demonstration program displays two separate graphs. The program displays the graph of a damped sine wave in the upper half of the screen and asks you to select a portion of that graph to be plotted in the lower region. In this way, you can look closely at a portion of a graph while still retaining the ability to examine the larger view of the graph.

Running the Program

To run the WINDOW program, type the following:

RUN WINDOW (RET)

When you press the RETURN key, MINC runs the program. The program displays the graph in the upper half of the screen and asks you to select the portion of the graph you wish to see enlarged, by entering the lower and upper end of the ranges.

Type "10" for the lower end of range and "20" for the upper end. The resulting display is shown in Figure 33.

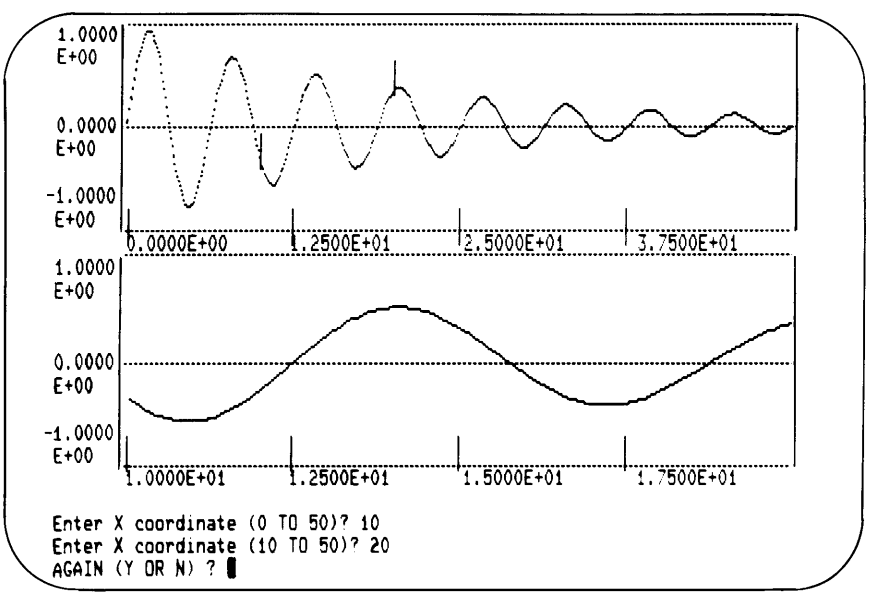


Figure 33. WINDOW Screen Sample

Looking at the Program

The WINDOW program uses graphic routines to display messages. In this way, the program can control the text that is on the screen and the format in which the text is presented.

Here is the portion of the WINDOW program that displays the four lines that start with the line "ENTER WINDOW LIMITS".

```
600 ERASE_TEXT("TEXT",12) \ HTEXT(,12,(40-LEN(Q1$))/2,Q1$)
610 ERASE_TEXT("TEXT",14) \ HTEXT(F2$,14,1,Q2$)
620 ERASE_TEXT("TEXT",15) \ HTEXT(F3$,15,1,Q3$)
630 ERASE_TEXT("TEXT",16) \ HTEXT(F4$,16,1,Q4$)
640 ERASE_TEXT("TEXT",17) \ HTEXT(F5$,17,1,Q5$)
```

This portion of the program displays the strings stored in Q1$ through Q5$ according to the instructions given in F1$ through F5$.

CHAPTER 7
IEEE BUS AND SERIAL ASCII CONTROLLER

This chapter demonstrates the two ways in which MINC's standard equipment interfaces with external instruments. The first demonstration illustrates the IEEE bus capability and the second demonstration illustrates the serial ASCII controller.

IEEE BUS DEMONSTRATION

If you have an IEEE-compatible instrument, you can connect it to the MINC IEEE bus and use the following program to send or receive messages.

Connection Instructions

To connect your IEEE-compatible instrument, follow these instructions:

1. Plug your instrument into MINC's IEEE bus connector.

2. Check the user's guide for your instrument to find its address. The program asks for this address.

Running the Program

To run the program, type the following:

```
RUN BUSDEM (RET)
```

When you press the RETURN key, MINC runs the program. The program identifies itself and requests that you enter the address of the IEEE-compatible instrument.

Suppose your instrument is an incremental digital plotter with address 1 that requires the following format to print a string:

```
P string
```

Type "S" followed by the RETURN key to send data and type "P Are you there??" followed by the RETURN key to plot the characters "Are you there??" The resulting display is shown in Figure 34.

```
IEEE BUS DEMONSTRATION PROGRAM

This program lets you send data to and receive data from an IEEE-
compatible instrument.  Connect your instrument to MINC using the
cable provided.  Check your instrument's user's guide to find the
instrument's address.

What is the address of your instrument (0  to 30)? 1

Do you wish to send or receive data (S or R)? s
What characters do you wish to send to the instrument? P Are you there??

AGAIN (Y or N)? N

READY
█
```

Figure 34. BUSDEM Screen Sample

If the demonstration program does not behave in the way described, then try the recovery procedures given on page 30.

Looking at the Program

Here is the portion of the BUSDEM program that sends data to an IEEE-compatible instrument connected to MINC's IEEE bus:

```
290 PRINT "What characters do you wish to send to the instrument?";
300 LINPUT A$
310 SEND (A$,I%)
```

The SEND routine sends the characters you just entered to the instrument with the address specified by I%.

Here is the portion of the BUSDEM program that receives characters from the IEEE-compatible instrument and displays them on the screen:

```
330 RECEIVE (A$,I%)
340 PRINT 'The instrument sent the following characters:'
350 PRINT\PRINT A$
```

The RECEIVE routine accepts a sequence of characters from the instrument with the address specified by I% and saves the character string in A$. If you want to look at the complete program, then type the following command:

LIST (RET)

After you press the RETURN key, MINC displays the program on the terminal screen.

You can learn more about IEEE bus programming by reading *Book 6: MINC IEEE Bus Programming.*

SERIAL ASCII CONTROLLER PROGRAM

If you have purchased the LA-35 line printer option, you can plug it into the connector labelled SLU2 and obtain a listing output of any of the programs on the demonstration diskette. For example, to get a listing of the program SHUFFL, you type the following command:

COPY SHUFFL LP: (RET)

In response to that command, MINC prints the program SHUFFL on the LA-35 line printer.

The serial ASCII demonstration program makes use of the wrap-around connectors that connect the transmission line to the reception line for Serial Line Unit 0 and Serial Line Unit 1. These wrap-around connectors are in place in SLU0 and SLU1 when MINC is unpacked. The program requests a channel number and an ASCII string and then sends the string out over the specified channel. The wrap-around connector then directs the string back to the same channel.

Running the Program

To run the program, type the following:

RUN SLUDEM (RET)

When you press the RETURN key, MINC runs the program. The program identifies itself and asks you to select a channel.

Choose channel 0 and send the string “greetings”. The resulting display is shown in Figure 35.

```
SERIAL ASCII DEMONSTRATION PROGRAM

A serial line unit is a device for sending and receiving ASCII codes.

You can send characters on one serial line unit and receive them
back on the same unit using the wrap-around connector.

This program requests a channel number and an ASCII message.
The string is sent out over the specified channel.  If a response
is received within 3 seconds of the transmission, the response
is printed.

What channel do you wish to use (0 or 1)? 0
What message do you want to send? greetings
Response received was: greetings

AGAIN (Y or N)? n

READY
```

Figure 35. SLUDEM Screen Sample

You can send as many messages as you like. To send another message, type "Y" followed by the RETURN key in response to the question "AGAIN (Y or N)?" To conclude the demonstration, type "N" followed by the RETURN key.

If your program does not behave as described, consult the recovery procedures given on page 30.

Looking at the Program

Here is the portion of the SLUDEM program that sends and receives the data:

```
162 CIN(,A$,1,C,,1)
164 COUT(,A$,,C)
166 CIN(,R$,,C,3)
```

The first CIN routine starts the channel specified by C listening. The COUT routine sends the string contained in A$ out to the channel specified by C. The second CIN routine waits 3 seconds for a response. If it receives a response on the channel specified by C during that time, it saves the response in R$.

If you want to look at the complete program, then type the following command:

LIST (RET)

After you press the RETURN key, MINC displays the program on the terminal screen.

CHAPTER 8
LAB MODULE PROGRAMMING

The lab module demonstrations are designed to be as simple to follow and to implement as possible.

Each demonstration program uses only one lab module. No matter which set of modules you purchased, there is a demonstration program for you to use.

Some of the demonstration programs require a function generator or a volt-ohmmeter to serve as a simulator for a laboratory device. These instruments have been chosen as device simulators because they are common pieces of laboratory equipment and because they can simulate the activity of a wide variety of laboratory devices. Moreover, they are essential to routine connection confirmation procedures and are defined as requirements by *Book 7: Working with MINC Devices*. The only requirements for the volt-ohmmeter and function generator are as follows:

- The volt-ohmmeter must be able to measure 5- or 10-volt dc levels.

- The function generator must be able to generate signals down into the 1/2 Hz range and to generate output levels of 5 to 10 volts peak-to-peak.

The demonstration programs appear in the order shown in the following list. For each demonstration program, the external instrument necessary is also given.

Module	*External Device*
A/D Module	Function Generator
D/A Module	Volt-ohmmeter
Digital Input	Function Generator
Digital Output	Volt-ohmmeter
Clock	(none)

Choose the demonstration you need. Follow the general instructions and then turn to the section describing the individual demonstration.

GENERAL INSTRUCTIONS

To perform the demonstrations for the A/D, D/A, DI, and DO modules you must connect the module to an external instrument. To do this proceed as follows:

1. Set the system power switch to the OFF position.

2. Open the chassis lid and locate the lab module required by the demonstration.

3. Remove its connector block by lifting upward with a slight rocking motion, as shown in Figure 36.

4. Locate the two test wires (stripped and tinned wire at

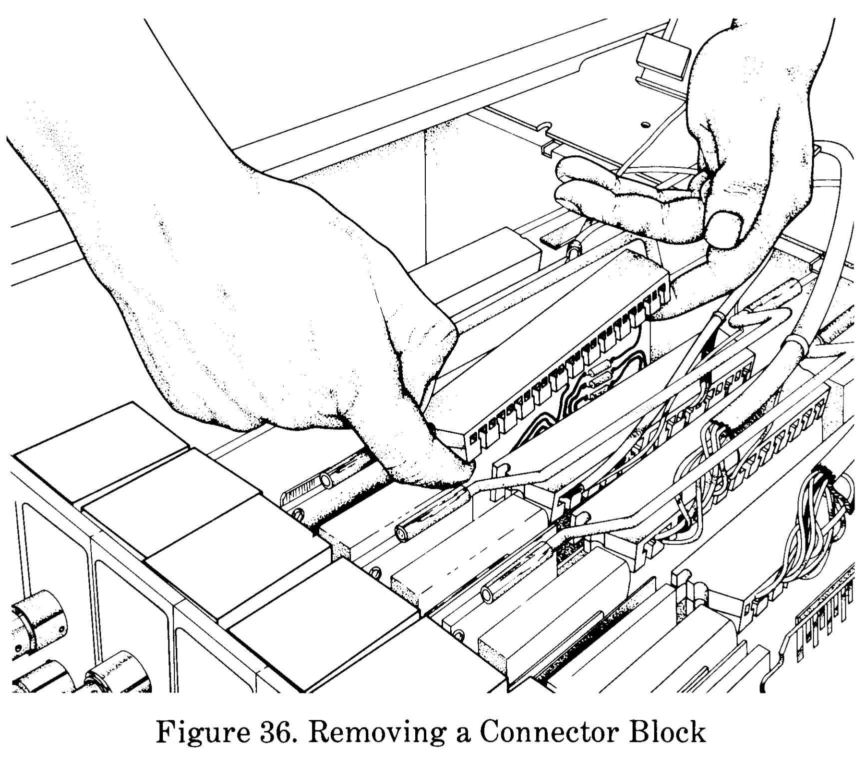


Figure 36. Removing a Connector Block

one end and probe clips at the other end). These test wires are originally packed in the bottom drawer of the interface kit. You will be using test wires to connect a lab module to an external instrument in the following demonstrations.

The specific connection instructions are given for each lab module.

ANALOG-TO-DIGITAL DEMONSTRATION

This demonstration illustrates the collection of analog input, its conversion to digital values, and its simultaneous display on the terminal screen. An example of such an application is plethysmographic monitoring, in which the medical researcher measures the ongoing changes from a blood volume measuring device attached to a subject's finger. For the purposes of this demonstration, the function generator provides the input.

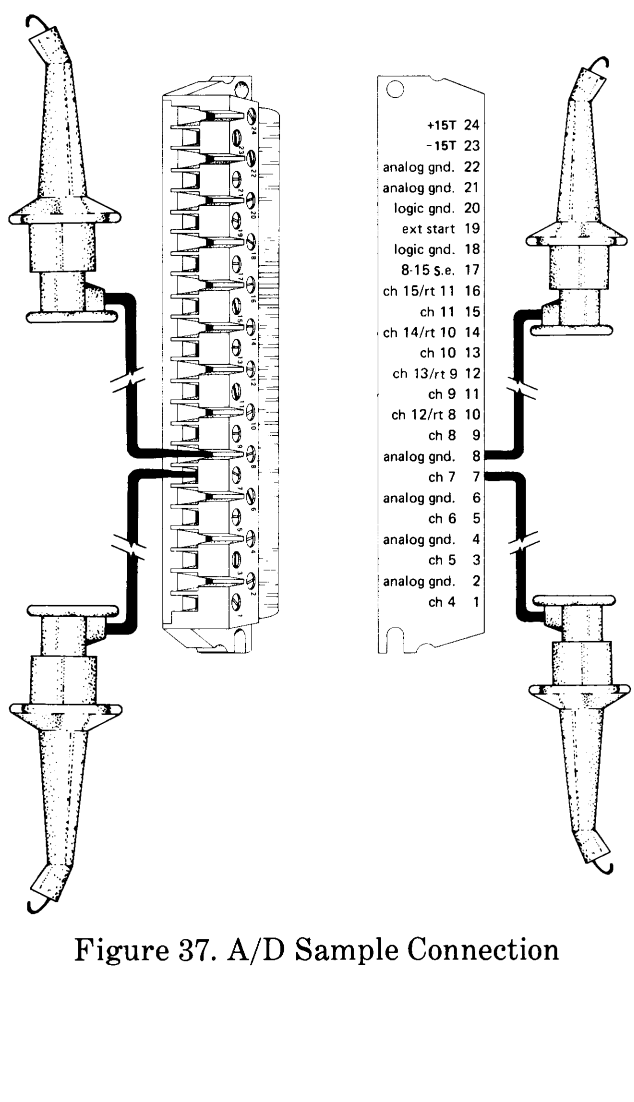


Figure 37. A/D Sample Connection

Connection Instructions

To connect the function generator to the A/D converter, proceed as follows:

1. Set the system power switch to the OFF position.

2. Locate the A/D converter; remove the connector block and locate the test wires, as described in "General Instructions."

3. Connect one of the test wires to a ground terminal on the connector block and the other wire to the connector block terminal for channel 7 (Figure 37).

4. Connect the clip attached to the ground wire to the negative or common lead of the function generator. Connect the remaining clip to the signal lead of the function generator (Figure 38).

5. Replace the A/D connector block, pushing firmly down until the block is fully seated.

6. Set the system power switch to the ON position and enter the date and time, as described on page 38.

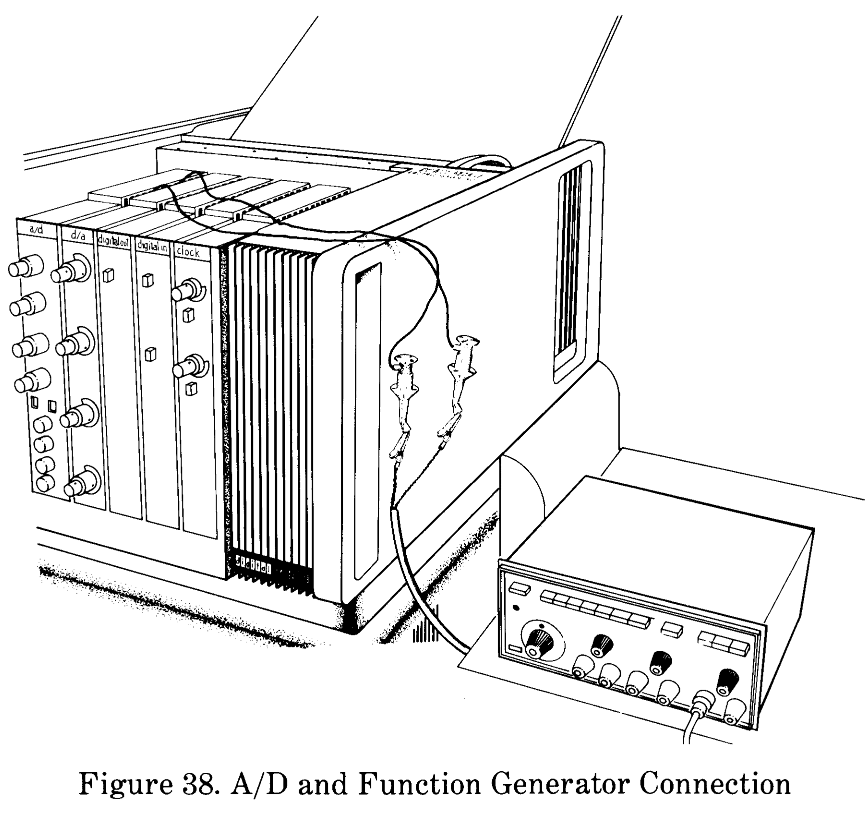


Figure 38. A/D and Function Generator Connection

7. Set the function generator mode to sine wave. Set the frequency to approximately 1/2 Hz and set the level to 2 volts peak-to-peak.

Running the Program

When you are satisfied that the A/D module is properly connected to the signal generator according to the instructions given above, run the demonstration program by typing the following command on your terminal keyboard:

RUN ADDEM (RET)

The program identifies itself and asks you to specify the channel to which you connected the function generator. The program then collects data for 120 seconds producing a strip chart representation of the input values on the terminal screen.

Since you connected to channel 7, type "7" in response to the program's request for channel number. The resulting display is shown in Figure 39.

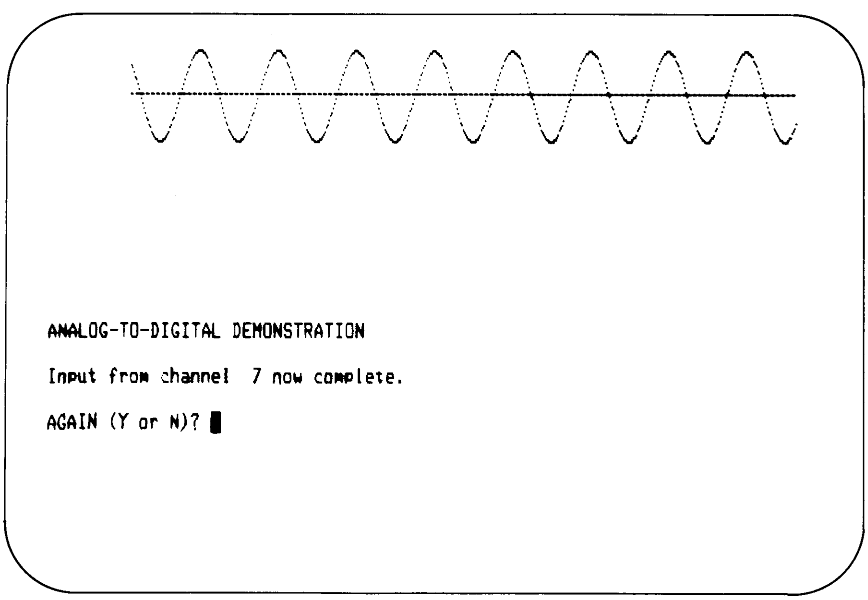


Figure 39. ADDEM Screen Sample

The program displays the analog values input from the function generator.

If the demonstration does not behave as described, check your connections and try the procedures suggested on page 30.

You can experiment with the function generator, changing the frequency by small increments, and changing the amplitude of the generator output. For each change in the signal, you can observe the effect on the terminal screen.

You can experiment with the other waveforms produced by your function generator. You can repeat the experiment with other channels by changing the wiring to the connector block and responding with that channel number.

To repeat the demonstration, type "Y" in response to the question "AGAIN (Y or N)?" To conclude the demonstration, type "N".

Looking at the Program

Here is the portion of the ADDEM program that collects the analog input:

```
210 REM Take points for 120 seconds. Display points on screen.
220 REM
240 I=1
250 SCHEDULE ("Interval",120,290)
255 REM Repeat the following until interval complete.
260 AIN('DISPLAY',M%(),,,C)
270 IF I=0 THEN GOTO 291
280 GOTO 260
290 I=0\RETURN
```

This portion of the program sets a flag, I, to one and then calls the SCHEDULE routine to specify a time interval of 30 seconds. After 120 seconds, the SCHEDULE routine sets flag I to zero. As long as the flag is one, indicating that the time interval is not complete, the program calls the AIN routine to read an analog value from the channel specified by C, convert the analog value to digital form, store it in M%, and display it on the screen. When the flag is zero, indicating that the time interval is complete, the program proceeds to the next logical step.

If you want to look at the complete demonstration program, type the command:

```
LIST (RET)
```

After you press the RETURN key, MINC displays the program on the terminal screen.

DIGITAL-TO- ANALOG DEMONSTRATION

This demonstration illustrates how to send output signals of the sort needed to drive oscilloscopes, analog X-Y plotters, and similar instruments. For the purposes of this demonstration, the volt-ohmmeter substitutes for one channel of the X-Y plotter.

Connection Instructions

To connect the volt-ohmmeter to the D/A converter, proceed as follows:

1. Set the system power switch to the OFF position.

2. Select the D/A converter that occupies the rightmost position in the chassis and remove the connector block, as described in "General Instructions."

3. Set the volt-ohmmeter to DC volts, select a range that will give nearly full-scale deflection for a 5-volt input.

4. Connect one of the test wires to a ground terminal on the connector block and connect the other wire to the terminal for channel 1 (DAC 1), as indicated in Figure 40.

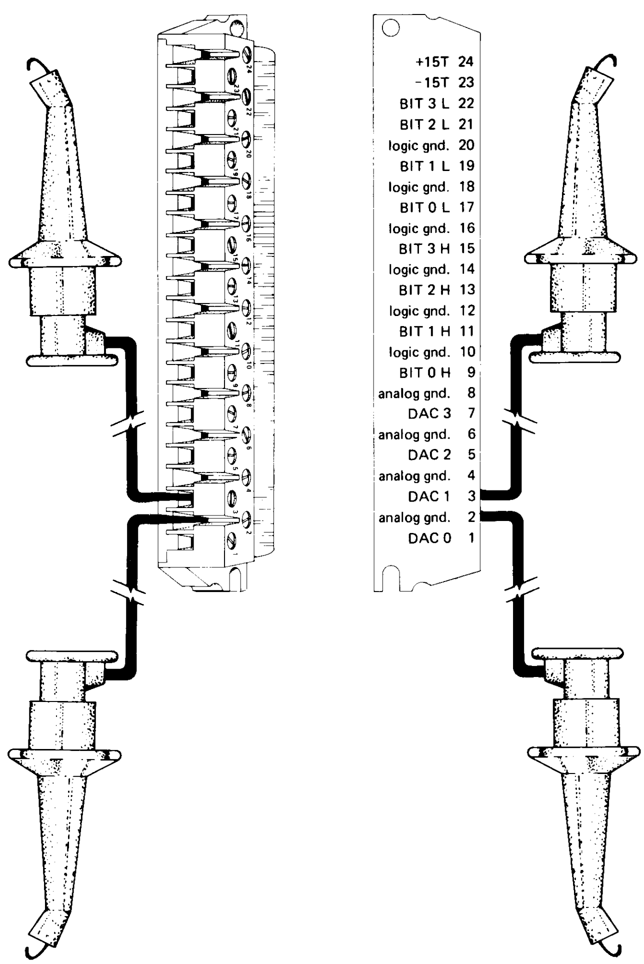


Figure 40. D/A Sample Connection

5. Connect the clip attached to the ground wire to the negative lead of the volt-ohmmeter. Connect the remaining clip to the positive lead (Figure 41).

6. Replace the D/A converter connector block, pushing firmly down until the block is fully seated.

7. Set the system power switch to the ON position and enter the date and time as described on page 38.

8. Set the selected D/A converter channel to unipolar mode, 5 volts.

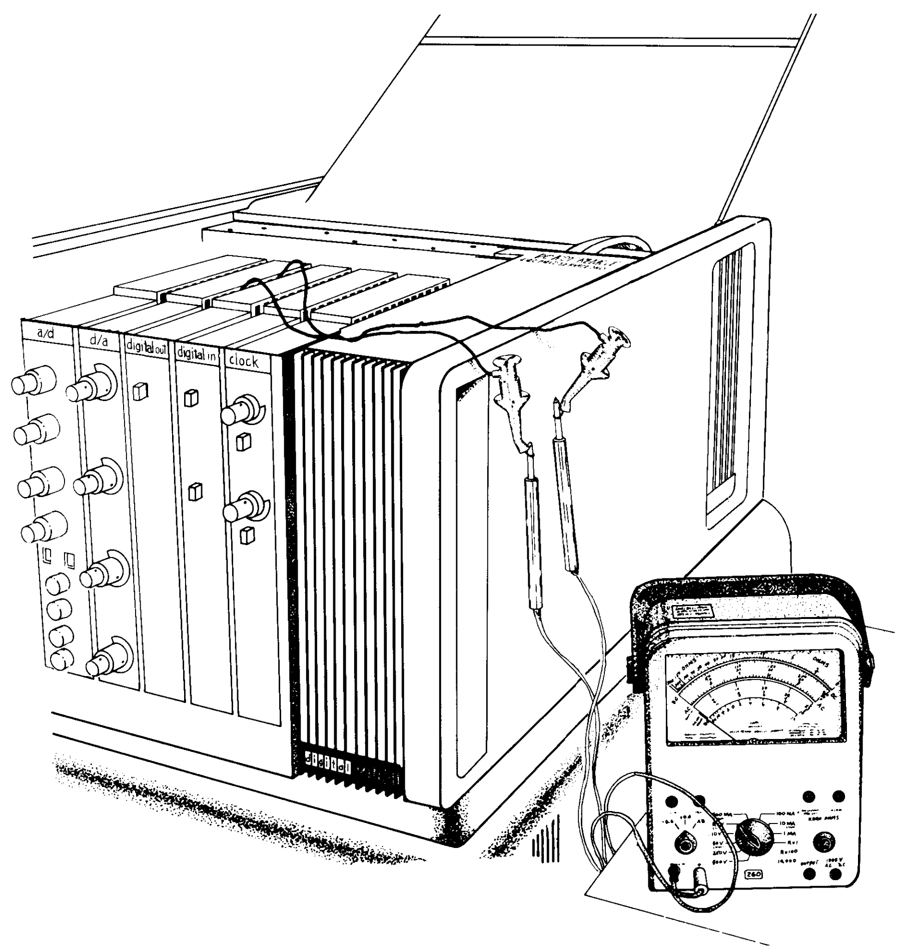


Figure 41. D/A and Volt-ohmmeter Connection

Running the Program

When you are satisfied that the D/A converter is properly connected to the volt-ohmmeter, run the demonstration program by typing the following command on the terminal:

RUN DADEM (RET)

The program identifies itself and asks some questions. When the program then asks you to choose a channel number, type "1". The resulting display is shown in figure 42.

```
DIGITAL-TO-ANALOG DEMONSTRATION PROGRAM

The digital-to-analog demonstration program  generates a voltage
at the D/A output causing the volt-ohmmeter to oscillate starting
with a small amplitude and slowly increasing until a maximum
amplitude is reached.

Connect the volt-ohmmeter to one of the four channels of the D/A
converter.  Set the volt-ohmmeter to DC volts and select a range
that will give  full-scale deflection for a 5 volt input.

How many seconds do you wish to run (1 to 30)? 25
How many D/A converters do you have in your configuration? 1
What channel did you connect the volt-ohmmeter to (0 to  3)? 1

AGAIN (Y or N)? n

READY
█
```

Figure 42. DADEM Screen Sample

The program generates a varying voltage at the D/A output, causing the volt-ohmmeter to oscillate, starting with a small amplitude and slowly increasing the amplitude until a maximum amplitude is reached. Then the amplitude declines until the minimum amplitude is reached. This process continues for the duration you specified.

If the demonstration program does not behave as described, check your connections and then try the procedures on page 30.

You can repeat the demonstration with other channels and other D/A converters by changing the wiring and responding with the appropriate channel number.

To repeat the demonstration, type "Y" in response to the question "AGAIN?" To conclude the demonstration, type "N".

Looking at the Program

Here is the portion of the DADEM program that performs the analog output:

```
180 REM Schedule an interrupt T seconds from now
200 S=0
210 SCHEDULE ('Interval',T,420)
```

```
270 REM Repeat the following code until interval completes.
290 IF S=1 THEN 500
300 AOUT(,V,,,C)
310 GOSUB 330
320 GOTO 290
```

This segment of the program uses the SCHEDULE routine to control the amount of time spent in sending data to the volt-ohmmeter. The program calls the AOUT routine to convert the value of V to analog form and send it to the output channel specified by C. The subroutine at line 500 computes the value V so that it varies in the way described above.

If you want to look at the complete program, type the command:

```
LIST (RET)
```

After you press the RETURN key, MINC displays the program on the terminal screen.

DIGITAL INPUT DEMONSTRATION

One of the simplest and most commonly-used monitoring procedures consists of counting external events such as organism responses, mechanical operations, and outputs from complex instruments. These events frequently need to be accumulated over controlled periods of time. This demonstration uses a digital input unit as an interface device and a function generator as a convenient external event producer.

Connection Instructions

To connect the function generator to the digital input unit, proceed as follows:

1. Set the system power switch to the OFF position.
2. Set the function generator output level to about 5V peak-to-peak (2.5 volt maximum positive excursion); set the mode to square wave; set the frequency to 5 or 10 Hz. **CAUTION:** Peak level must not exceed -16 or +20V.
3. Select the digital input unit that occupies the rightmost position in the chassis and remove its connector block, as described on page 60.
4. Connect one of the test wires to a ground terminal on the connector block and the other wire to input line 2 of the digital input module (Figure 43).

5. Connect the clip attached to the ground wire to the negative or common terminal of the function generator; connect the remaining wire to the signal terminal of the generator (Figure 44).

6. Replace the digital input connector block, with wires attached, pushing firmly down until the block is fully seated.

7. Set the system power switch to the ON position and enter the date and time, as described on page 38.

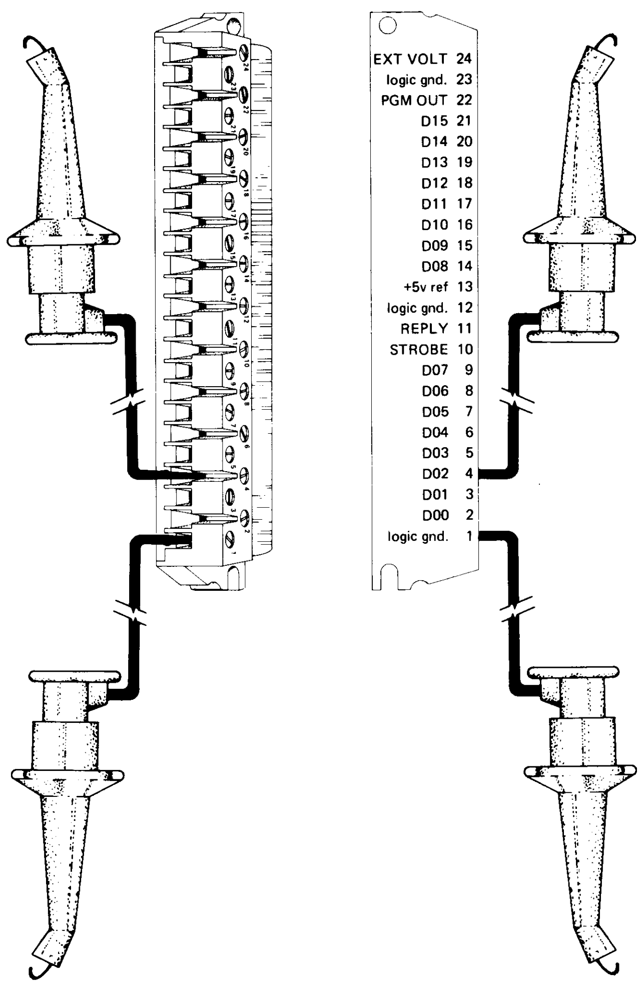


Figure 43. Digital Input Sample Connection

Running the Program

When you are satisfied that the digital input module is properly connected to the function generator, run the demonstration program by typing the following command on your terminal:

RUN DIDEM (RET)

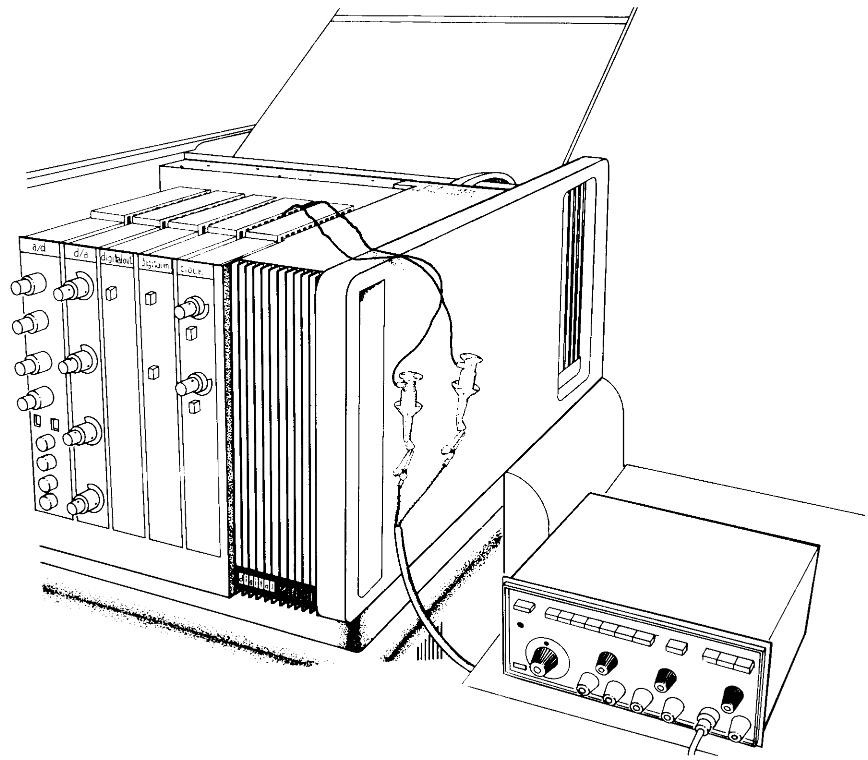


Figure 44. Digital Input and Function Generator Connection

MINC asks you to specify the digital input module (if you have more than one in the chassis) and the number of seconds you want to spend collecting data. The program then runs for the specified amount of time. MINC examines all the input lines of the specified digital input unit and determines, by the incoming data, the channel that you connected. At the end of the duration you specified, MINC displays the total number of samples taken, and the status of the specified line.

Choose a duration of 25 seconds. The resulting display is shown in Figure 45.

Because you set the mode of the function generator to square wave, the function generator puts out a signal that spends half time in a set state (above the threshold) and half time in a clear state (below the threshold). Thus, the number of samples shown in the set state should be approximately the same as the number of samples taken in the clear state.

If the demonstration program does not behave as described check the connections and then try the procedures suggested on page 30.

```
DIGITAL INPUT MODULE DEMONSTRATION PROGRAM

The digital input demonstration program looks at all the channels
of the digital input module that you specify for the amount of time
you specify and then reports the number of samples taken on each
channel

How many seconds do you wish to run (5 to 30)? 25
How many digital input modules do you have in the chassis? 1

A total of  726 samples were taken.

Line no.       No. set          No. clear    Percent set.

     2           368               358         50.69

AGAIN (Y or N)?
```

Figure 45. DIDEM Screen Sample

You can repeat the demonstration with other channels and other digital input modules. To repeat the demonstration, type "Y" in response to the question "AGAIN (Y or N)". To conclude the demonstration, type "N".

Looking at the Program

Here is the portion of the program that performs the digital input.

```
210 S=0
220 SCHEDULE('INTERVAL',T,2000)
240 REM---Loop until the signal S is nonzero
255 IF S>0 THEN 400
260 DIN(,M,,U)
265 K=K+1
280 REM---Scan for active bits
300 SCAN_BIT(J,M)
310 IF J<0 THEN 255
320 C(J)=C(J)+1
330 GOTO 300
```

This segment of the program uses the SCHEDULE routine to control the amount of time spent looking for digital input. The DIN routine reads the digital input unit specified by U. The routine SCAN_BIT determines the line on which the input is received. Then the count C(J) associated with that line is incremented.

If you want to look at the complete demonstration program, type the following command:

LIST (RET)

After you press the RETURN key, MINC displays the program on the terminal screen.

DIGITAL OUTPUT DEMONSTRATION

This demonstration illustrates how you can use a digital output module to control the frequency and duration of an OFF/ON condition — for example, a stimulus in a psychology experiment. For the purposes of this demonstration, the volt-ohmmeter acts as the stimulus-monitoring instrument.

Connection Instructions

To connect the volt-ohmmeter to the digital output module, follow the steps given below:

1. Set the system power switch to the OFF position.

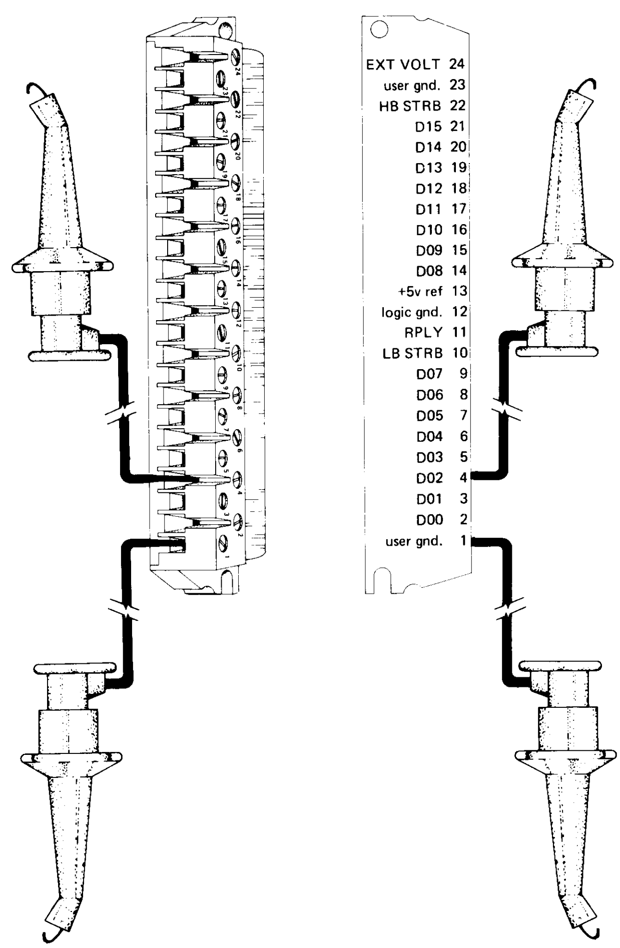


Figure 46. Digital Output Sample Connection

2. Select the digital output module that occupies the rightmost position in the chassis and remove the connector block, as described on page 62.

3. Set the volt-ohmmeter to DC volts. Select a range that will give nearly full-scale deflection for a 3.5- to 4-volt input.

4. Connect one of the test wires to a ground terminal on the digital output connector block, and connect the other wire to output line 2 (Figure 46).

5. Connect the clip attached to the ground wire to the common lead of the volt-ohmmeter. Connect the remaining clip to the positive lead (Figure 47).

6. Replace the digital output connector block, pushing firmly down until the block is fully seated.

7. Set the system power switch to the ON position and enter the date and time, as described on page 38.

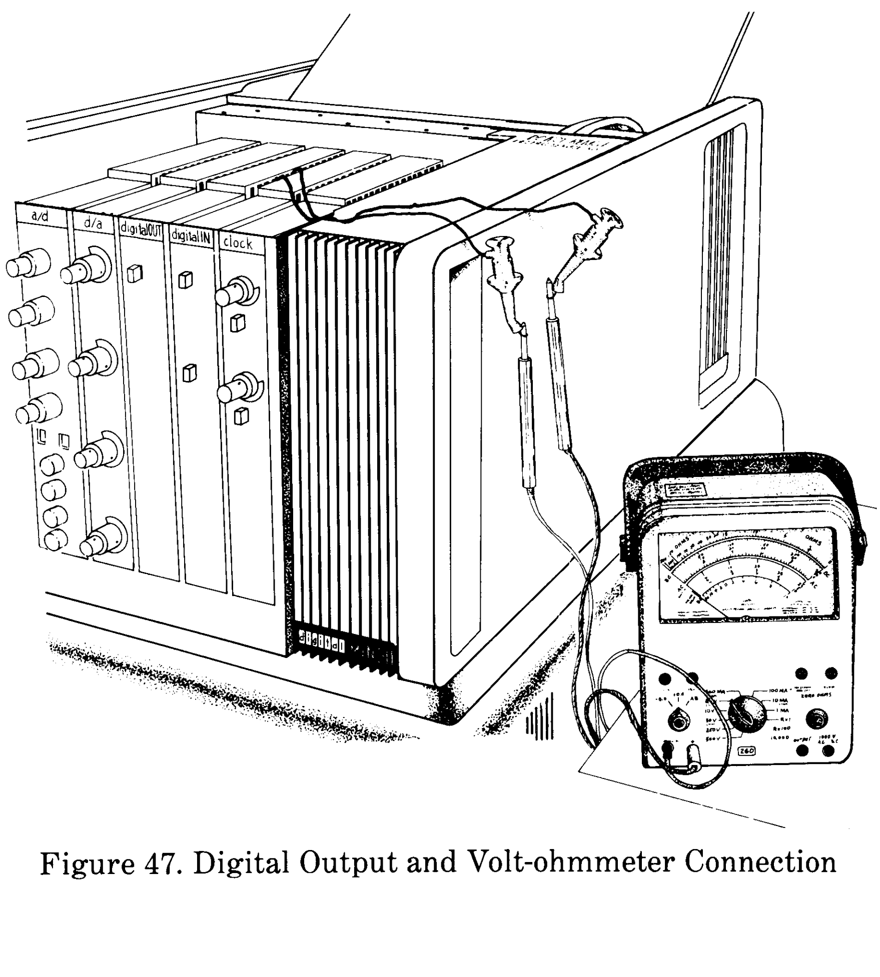


Figure 47. Digital Output and Volt-ohmmeter Connection

Running the Program

Now the digital output module is connected to the volt-ohmmeter, and you can call the demonstration program by typing the following command on the terminal:

RUN DODEM (RET)

The program identifies itself and asks some questions.

Choose a 20-second test duration, output line 2, a set period of 1 second, and a clear period of 5 seconds. The resulting display is shown in Figure 48.

```
DIGITAL OUTPUT MODULE DEMONSTRATION PROGRAM

The digital output demonstration program moves the needle of a volt-
ohmmeter to a 0 volt deflection (approximately) for a set period and
to a 4 volt deflection for a clear period

You specify the digital output unit number and  the line number for
that unit to which you connected the volt ohmmeter.  Then, you specify
the amount of time you want the demonstration to run and the amount
of time you want to spend in the set state and the amount of time
you want to spend in the clear state.

Watch the volt-ohmmeter to see the two states.

How many seconds do you want to run (5 to 30)? 20
How many digital output units do you have in the chassis (1 to 4)? 1
What line did you connect the volt-ohmmeter to (0 to 15)? 2
How many seconds do you want in the set period? 1
How many seconds do you want in the clear period? 5

AGAIN (Y or N)?
```

Figure 48. DODEM Screen Sample

The program generates "stimuli" on the volt-ohmmeter with the set period and clear period time you specified in answering the questions asked by the demonstration program. Note that clear period is approximately 4 volts and the set period is approximately 0 volts.

If the demonstration program does not behave as described, try the procedures suggested on page 60.

You can repeat the demonstration specifying a different time duration or different set and clear periods, or you can change the connections and run the demonstration program for different output lines or different digital output modules. To repeat the demonstration, type "Y" in response to the question "AGAIN (Y or N)?" To conclude the demonstration, type "N".

Looking at the Program

Here is the portion of the DODEM program that is concerned with digital output.

```
90 REM Specify and schedule interval.
100 J=0
110 SCHEDULE('Interval',T,2000)
120 REM Repeat the following until interval complete.
230 IF J=1 THEN 400
250 REM First, set line for P1 seconds.
260 REM Then, clear line for P2 seconds.
300 SET_LINE(L,1,U)
310 PAUSE(P1)
320 SET_LINE(L,0,U)
330 PAUSE(P2)
340 GOTO 230
2000 J=1 \ RETURN
```

This segment of the program uses the SCHEDULE routine to control the amount of time spent sending data through the digital output module to the volt-ohmmeter. The SET_LINE routine sets the line indicated by L for the digital output module specified by U to the set state. The PAUSE routine pauses for the number of seconds you requested to be in the set period. The next SET_LINE statement (line 320) clears the line and the PAUSE statement following it pauses for the clear period you requested. The process of setting and clearing the lines continues until the interval is complete.

If you want to look at the complete demonstration program, type the command:

LIST (RET)

After you press the RETURN key, MINC displays the program on the terminal screen.

CLOCK MODULE DEMONSTRATION

The need to measure the time it takes a person to answer a question often occurs in the psychology laboratory. The clock module demonstration program asks you to perform a sequence of simple arithmetic calculations. The program uses the clock module to determine the time it takes you to respond. Your answer is checked for correctness and your response time recorded. At the end of the sequence of calculations, your score is printed.

Running the Program

Before running the program, pull out the knobs on the clock module labeled ST1 and ST2 and turn these knobs clockwise to the last position.

To run the program, type the following:

RUN CLKDEM (RET)

When you press the RETURN key, MINC runs the program. It identifies itself and asks you how many calculations you want to try.

Choose five calculations. Then respond to each calculation presented by typing the answer followed by the RETURN key. Your response time is calculated after you press the RETURN key. The resulting display is shown in Figure 49.The program chooses a different set of problems each time you run it, so the problems, the answers you give, and the response time will be different from those shown in the figure.

```
 +  means addition
 -  means subtraction
 *  means multiplication

Your answer is validated and your response time computed.

How many problems do you want to try (1 to 20)? 5

 7 - 5 = ? 2
                    CORRECT!  Your response time -   1.47 seconds
 8 * 4 = ? 32
                    CORRECT!  Your response time -   2.33 seconds
 9 + 8 = ? 17
                    CORRECT!  Your response time -   2.36 seconds
 2 + 3 = ? 5
                    CORRECT!  Your response time -   1.35 seconds
 6 + 3 = ? 9
                    CORRECT!  Your response time -   1.75 seconds

You had  5 correct and  0 wrong answers.
Average correct response time was  1.85 seconds

Again (Y or N)? █
```

Figure 49. CLKDEM Screen Sample

If the time given for your response does not seem reasonable, then try the recovery procedures given on page 30.

If you want to try another sequence of calculations, type "Y" followed by the RETURN key in response to the question "AGAIN?" If you want to conclude the demonstration, type "N".

Looking at the Program

Here is the portion of the CLKDEM program that determines your response time.

```
360 START_TIME('KHZ') \ REM -- START CLOCK
370 INPUT A \ REM -------------- GET ANSWER FROM USER
380 GET_TIME(T) \ REM ---------- GET RESPONSE TIME
```

The routine START_TIME starts an elapsed time clock running at 1 KHz. The INPUT statement waits for your response and stores your response for later evaluation by the program. The GET_TIME routine reads the time elapsed between starting the clock and receiving the answer.

To list the complete program, type the following command:

LIST (RET)

In response to this command, MINC displays the program on the terminal screen.

CHAPTER 9
CONCLUDING REMARKS

This concludes the introduction to MINC.

If you have just unpacked MINC, there are several additional things you should do, namely:

- Make a copy of the Master System Diskette. To do this, you follow the instructions given for copying the Master Demonstration diskette except that you insert the Master System diskette in the left-hand drive.

- Turn to Book 7 and follow the directions for Diagnostic Procedures.

If you are new to programming, turn to Book 2 for simple, step-by-step instructions on how to proceed. If you are an experienced programmer, you will probably still want to skim Book 2. It contains a detailed introduction to the MINC editor and valuable insights into the effective use of MINC.

If you plan to use the graphic capability, read the explanatory part of Book 4. If you plan to connect IEEE-compatible instruments to MINC, read the explanatory part of Book 5. If you plan to use the Serial ASCII controller or the lab modules, read the explanatory part of Book 6.

APPENDIX A
HANDLING UNUSUAL CONDITIONS

This appendix describes some unusual conditions that might occur when you first begin using MINC. Typically, these conditions occur if MINC is not properly connected or if some normal operating procedure is overlooked.

This appendix attempts to give advice about what to do when a particular condition occurs.

Condition	*Procedure*
You turn on the system power switch and the system does not identify itself and ask for the date.	Make sure that a demonstration or system diskette is in drive 0 and that the drive door is properly closed. Check that all the switches are set properly, as described on page 35. Check that the power cord is plugged into an electrical outlet. Check that all the cables are connected correctly. (Refer to Book 7 to determine proper connections.)
You get one of the following messages when you turn on the system power switch: BOOT-F-I/O error MON-F-System read failure halt	Check that a demonstration or system diskette is in drive 0 and that the drive door is closed correctly.

You are using MINC and an "@" character followed by a sequence of digits appears on the terminal screen.	Hold down the shift key and press the P key. If MINC responds with the message "READY", you can continue your work. If not, you must restart MINC, by turning off the system power switch, waiting 5 seconds, turning on the system power switch, and entering the date and time. Then you must start at the beginning of a demonstration by typing "RUN" followed by the program name.
You type characters on the keyboard and they do not appear on the terminal screen.	Press the NO SCROLL key and try typing some more characters. If these characters appear on the screen, you can continue your work. If not, check that the terminal keyboard is plugged into the terminal. If you find that the plug is loose, plug it in securely and you should be able to continue your work. If the above procedures don't help, you must restart MINC. If you have trouble restarting MINC, consult the first two error condition recovery procedures.
The blinking command "SET UP" appears in the upper left-hand corner of the terminal screen.	Press the SETUP key (located in the upper left-hand corner of the terminal keyboard). The screen should return to normal and you should be able to continue your work. If not, you must restart MINC. If the condition persists, refer to "SETUP Directions" in Book 7.

If none of the conditions given here seem to apply to your problem, try restarting MINC by following the instructions for starting MINC given in Chapter 4.

If the problem persists, consult Error Recovery in Book 3 or "Troubleshooting" in Book 7.

READER'S COMMENTS

NOTE: This form is for document comments only. DIGITAL will use comments submitted on this form at the company's discretion. If you require a written reply and are eligible to receive one under Software Performance Report (SPR) service, submit your comments on an SPR form.

Did you find this manual understandable, usable, and well-organized? Please make suggestions for improvement.

Did you find errors in this manual? If so, specify the error and the page number.

Please indicate the type of user/reader that you most nearly represent.

- ☐ Assembly language programmer
- ☐ Higher-level language programmer
- ☐ Occasional programmer (experienced)
- ☐ User with little programming experience
- ☐ Student programmer
- ☐ Other (please specify)__________

Name __________ Date __________

Organization __________

Street __________ Telephone __________

City __________ State __________ Zip Code or Country __________

Fold Here

Do Not Tear - Fold Here and Staple

FIRST CLASS
PERMIT NO. 152
MARLBOROUGH,MA
01752

BUSINESS REPLY MAIL
NO POSTAGE STAMP NECESSARY IF MAILED IN THE UNITED STATES

Postage will be paid by:

digital

Software Publications
200 Forest Street MR1-2/E37
Marlborough, Massachusetts 01752